18th International Conference on Information Fusion
Washington, DC - July 6-9, 2015

Gaussian–Mixture based Ensemble Kalman Filter

Felix Govaers, Wolfgang Koch
Fraunhofer FKIE
Wachtberg, Germany
Email: {felix.govaers,wolfgang.koch}@fkie.fraunhofer.de

Peter Willett
University of Connecticut
Storrs, USA
Email: peter.willett@uconn.edu

***Abstract*—The Ensemble Kalman Filter (EnKF) is a Kalman based particle filter which was introduced to solve large scale data assimilation problems where the state space is of very large dimensionality. It also achieves good results when applied to a target tracking problem, however, due to its Gaussian assumption for the prior density, the performance can be improved by introducing Gaussian mixtures. In this paper, a new derivation of the EnKF is presented which is based on a duality between Gaussian products and particle densities. A relaxation of the Gaussian assumption is then achieved by introducing a particle clustering into Gaussian Mixtures by means of the Expectation Maximization (EM) algorithm and to apply the EnKF on the clusters. The soft assignment of the EM allows all Gaussian components to contribute to each of the particles. It is shown that the EM-EnKF performs better than a standard particle filter while having less computation time.**

I. Introduction

The theory of target tracking is a growing family of algorithms to calculate the probability density function (pdf) of the state of a system or an object based on noise corrupted sensor observations. An estimate of the state is then obtained by taking the mean of the pdf and a corresponding covariance matrix which is the expected estimation error squared additionally provides a measure of accuracy for this estimate. Bayesian estimation is the framework of recursive filtering methodologies, which allow us to process a current measurement by means of a *prior* or *initial* density and a measurement likelihood function which statistically describes the performance of the involved sensor. Thus, a tracking algorithm is an iterative updating scheme for calculating conditional probability density functions $p(\mathbf{x}_k|\mathcal{Z}^k)$ that represent all available knowledge on the object states $\mathbf{x}_k$ at some time t_k, which typically is chosen as the present time. The densities are explicitly conditioned by the sensor data time series $\mathcal{Z}^k$. The iterative scheme consists of two processing steps per update cycle, referred to as *prediction* and *filtering*. The propagation of the probability densities involved is given by basic update equations (see [1] for instance).

Prediction. The prediction density $p(\mathbf{x}_k|\mathcal{Z}^{k-1})$ is obtained by combining the evolution model $p(\mathbf{x}_k|\mathbf{x}_{k-1})$ with the previous filtering density $p(\mathbf{x}_{k-1}|\mathcal{Z}^{k-1})$:

$$p(\mathbf{x}_{k-1}|\mathcal{Z}^{k-1}) \xrightarrow[\text{constraints}]{\text{evolution model}} p(\mathbf{x}_k|\mathcal{Z}^{k-1})$$

$$p(\mathbf{x}_k|\mathcal{Z}^{k-1}) = \int \mathrm{d}\mathbf{x}_{k-1} \underbrace{p(\mathbf{x}_k|\mathbf{x}_{k-1})}_{\text{evolution model}} \underbrace{p(\mathbf{x}_{k-1}|Z^{k-1})}_{\text{previous filtering}}. \tag{1}$$

Filtering. The filtering density $p(\mathbf{x}_k|\mathcal{Z}^k)$ is obtained by combining the sensor model $p(\mathbf{z}_k|\mathbf{x}_k)$, also called the "likelihood function", with the prediction density $p(\mathbf{x}_k|\mathcal{Z}^{k-1})$ according to:

$$p(\mathbf{x}_k|\mathcal{Z}^{k-1}) \xrightarrow[\text{sensor model}]{\text{current sensor data}} p(\mathbf{x}_k|\mathcal{Z}^k)$$

$$p(\mathbf{x}_k|\mathcal{Z}^k) = \frac{p(\mathbf{z}_k|\mathbf{x}_k)\, p(\mathbf{x}_k|\mathcal{Z}^{k-1})}{\int \mathrm{d}\mathbf{x}_k \underbrace{p(\mathbf{z}_k|\mathbf{x}_k)}_{\text{sensor model}} \underbrace{p(\mathbf{x}_k|\mathcal{Z}^{k-1})}_{\text{prediction}}}. \tag{2}$$

According to this paradigm, an *object track* represents all relevant knowledge on a time varying object state of interest, including its history and measures that describe the quality of this knowledge. As a technical term, 'track' is therefore either a synonym for either the collection of densities $p(\mathbf{x}_l|Z^l)$, $l = 1, \ldots, k, \ldots$, or of suitably chosen parameters characterizing them, such as estimates related to appropriate risk functions and the corresponding estimation error covariance matrices.

An analytical solution to a recursive computation of these densities is given for instance by the Kalman filter in the case of linear Gaussian models. For non–linear scenarios, only approximate solutions are feasible. The first-order Taylor approximation is called the *extended Kalman filter* (EKF) which has low computation cost, since it is still an analytic solution. The performance of the linearization can be improved by means of deterministic samples chosen at the local neighborhood of the current estimate. This algorithm is known as the *unscented Kalman filter* (UKF). The term *Particle filter* (PF) subsumes all kinds of numerical solutions. Here, knowledge about the state typically is represented by a set of state samples, which implies that the density is given by a Dirac–mixture. Since then the noise terms are simulated by means of appropriately sampled random numbers, these methods are also known as *sequential Monte–Carlo* (SMC) methods.

In the literature, a variety of particle filter algorithms can be found [2]. Still, the basic *sampling importance resampling* (SIR) particle filter [3] is often used due to its robustness. The main drawback of the SIR–PF is that it can suffer from

impoverishment of the particle weights. For numerical reasons, resampling has to be used in order to avoid the particles to degenerate. More recently, new algorithms have been proposed which are based on a log–homotopy transition between the prior and the posterior. For instance, the Daum–Huang filters (see [4] and the references within) model this transition phase in terms of a physical flow which is determined by a "force" induced by the measurement. This leads to a stochastic differential equation (SDE) which then can be solved numerically by introducing a discretized pseudo time evolving from prior to posterior. However, the computation time for the SDE is too high for standard target tracking scenarios, and as a consequence these algorithms are often outperformed by a SIR–PF as shown in [5]. A different homotopy approach is provided by the *progressive filter* which was presented by Hanebeck in [6]. In the progressive filter, a incremental inclusion of the likelihood function is achieved by a partition of the exponent going from zero to one. This prevents particle impoverishment by means of frequent resampling and an appropriately chosen step size. To improve the computation speed, a fast resampling method based on a Gaussian approximation has been proposed. However, this Gaussian assumption might lead to additional approximation errors. A Kullback–Leibler divergence based approach to obtain the posterior particles is proposed in [7]. The resulting algorithm is similar to the *Ensemble Kalman filter* (EnKF) based filter proposed in [8], however, the additional noise term in the Kalman based update is different. The EnKF adds some zero–mean Gaussian distributed noise to the measurement for each sample and applies Kalman filter update equations for each particle. As a consequence, a consistent filter results which is consitent and performs well in non–linear scenarios. The EnKF also has been extended to Gaussian mixtures in [9], [10].

In this paper a novel derivation of the EnKF by means of a Gaussian product representation is presented. Then, the Gaussian assumption for the prior is relaxed and the product representation is applied to Gaussian mixtures. The Gaussian mixture clustering is achieved by the *Expectation Maximization* (EM) algorithm [11].

Structure: This paper is structured as follows. In Section II, the statistical models are provided and a formulation of the estimation problem is given. Then, in Section III, the relationship between a Gaussian distributed particle distribution and a Gaussian product representation is introduced. This relationship is used in Section IV to compute the posterior particle states in the linear as well as in the non–linear case. An evaluation of the algorithm is provided in Section VI and the paper is concluded in the final section.

II. Formulation of the Problem

Throughout this paper, the following models are assumed. The state transition from time t_{k-1} to time t_k given by the equation[1]

$$\mathbf{x}_k = \mathbf{f}_{k|k-1}(\mathbf{x}_{k-1}) + \mathbf{w}_k \tag{3}$$

where $\mathbf{f}_{k|k-1}(\mathbf{x}_{k-1})$ describes the deterministic evolution of the parameters in time and $\mathbf{w}_k$ is some Gaussian zero–mean random noise with covariance matrix $\mathbf{Q}_{k|k-1}$. Whenever $\mathbf{f}_{k|k-1}$ is a linear function in the state, we can write it as a matrix and will use a capital letter notation:

$$\mathbf{x}_k = \mathbf{F}_{k|k-1}\mathbf{x}_{k-1} + \mathbf{w}_k. \tag{4}$$

Note that this transition is fully described by the Markov kernel

$$p(\mathbf{x}_k|\mathbf{x}_{k-1}) = \mathcal{N}\big(\mathbf{x}_k;\, \mathbf{f}_{k|k-1}(\mathbf{x}_{k-1}),\, \mathbf{Q}_{k|k-1}\big). \tag{5}$$

The sensing process at time t_k is modeled by the measurement equation[1]

$$\mathbf{z}_k = \mathbf{h}_k(\mathbf{x}_k) + \mathbf{v}_k. \tag{6}$$

The measurement function $\mathbf{h}_k(\mathbf{x}_k)$ is the mapping from state space into the measurement space. The Gaussian zero–mean random noise $\mathbf{v}_k$ describes the measurement error of the sensor. Its covariance is given by the matrix $\mathbf{R}_k$. As a consequence, the measurement is described as random variable itself whereby its distribution is given by the probability density function

$$p(\mathbf{z}_k|\mathbf{x}_k) = \mathcal{N}\big(\mathbf{z}_k;\, \mathbf{h}_k(\mathbf{x}_k),\, \mathbf{R}_k\big). \tag{7}$$

Again, whenever the measurement function is linear in the state, the matrix notation is used: $\mathbf{h}_k(\cdot) = \mathbf{H}_k$. Since in many applications, either the transition model or the measurement model can be chosen to be linear, for the remainder the former is assumed to be given by the linear Gaussian density:

$$p(\mathbf{x}_k|\mathbf{x}_{k-1}) = \mathcal{N}\big(\mathbf{x}_k;\, \mathbf{F}_{k|k-1}\mathbf{x}_{k-1},\, \mathbf{Q}_{k|k-1}\big). \tag{8}$$

The goal of a Bayesian filter is to compute recursively the posterior density $p(\mathbf{x}_k|\mathcal{Z}^k)$, which provides an estimate of the state by taking the mean, and where $\mathcal{Z}^k$ is the time series of sensor data:

$$\mathcal{Z}^k = \{\mathbf{z}_1, \ldots, \mathbf{z}_k\}. \tag{9}$$

The recursion is set up by an initial density at time t_1 which is based in the first observation $\mathbf{z}_1$. Then, for each instant of time t_k, the current result is obtained by a prediction–filtering cycle applied to the posterior at time t_{k-1} as described in (1) and (2).

III. On the Gaussian Product Particle Duality

In this section, the focus is on the transition of a particle representation of a pdf at time t_{k-1} to time t_k. Therefore, let the representation of the density at time t_{k-1} be given by a Dirac mixture of N components:

$$p(\mathbf{x}_{k-1}|\mathcal{Z}^{k-1}) = \frac{1}{N}\sum_{i=1}^{N} \delta(\mathbf{x}_{k-1} - \mathbf{x}^i_{k-1|k-1}). \tag{10}$$

[1]The assumption of additive noise is not the most general model, however, it is general enough for most practical applications.

According to the laws of marginalization and due to the Markov property, the prior at time t_k which refers to the density at the current time without consideration of the measurement $\mathbf{z}_k$ is given by the integral

$$p(\mathbf{x}_k|\mathcal{Z}^{k-1}) = \int \mathrm{d}\mathbf{x}_{k-1}\, p(\mathbf{x}_k|\mathbf{x}_{k-1}) \cdot \frac{1}{N}\sum_{i=1}^{N} \delta(\mathbf{x}_{k-1} - \mathbf{x}^i_{k-1|k-1}) \tag{11}$$

which results in a Gaussian mixture by means of the model introduced in (8):

$$p(\mathbf{x}_k|\mathcal{Z}^{k-1}) = \frac{1}{N}\sum_{i=1}^{N} \mathcal{N}\big(\mathbf{x}_k;\, \mathbf{F}_{k|k-1}\mathbf{x}^i_{k-1|k-1},\, \mathbf{Q}_{k|k-1}\big). \tag{12}$$

An approximation of (12) by means of a particle set is obtained by taking samples from the Gaussian mixture components and it is well known that taking

$$p(\mathbf{x}_k|\mathcal{Z}^{k-1}) = \frac{1}{N}\sum_{i=1}^{N} \delta(\mathbf{x}_k - \mathbf{x}^i_{k|k-1}) \tag{13}$$

with

$$\mathbf{x}^i_{k|k-1} = \mathbf{F}_{k|k-1}\mathbf{x}^i_{k-1|k-1} + \mathbf{w}^i_{k|k-1}, \tag{14}$$

$$\mathbf{w}^i_{k|k-1} \sim \mathcal{N}\big(\mathbf{O},\, \mathbf{Q}_{k|k-1}\big) \tag{15}$$

satisfies this approximation. However, these samples might also be computed by using a first and second moment approximation of the sampled posterior density at time t_{k-1}:

$$p(\mathbf{x}_{k-1}|\mathcal{Z}^{k-1}) = \frac{1}{N}\sum_{i=1}^{N} \delta(\mathbf{x}_{k-1} - \mathbf{x}^i_{k-1|k-1}) \tag{16}$$

$$\approx \mathcal{N}\big(\mathbf{x}_{k-1};\, \mathbf{x}^{(N)}_{k-1|k-1},\, \mathbf{P}^{(N)}_{k-1|k-1}\big) \tag{17}$$

where the mean is given by

$$\mathbf{x}^{(N)}_{k-1|k-1} = \frac{1}{N}\sum_{i=1}^{N} \mathbf{x}^i_{k-1|k-1}, \tag{18}$$

and the covariance can be computed by

$$\mathbf{P}^{(N)}_{k-1|k-1} = \frac{1}{N-1}\sum_{i=1}^{N} \big\{(\mathbf{x}^{(N)}_{k-1|k-1} - \mathbf{x}^i_{k-1|k-1})(\mathbf{x}^{(N)}_{k-1|k-1} - \mathbf{x}^i_{k-1|k-1})^\top\big\}. \tag{19}$$

According to the product formula in the Appendix, the approximation is proportional to a product of Gaussians with the following parameters

$$p(\mathbf{x}_{k-1}|\mathcal{Z}^{k-1}) \propto \prod_{i=1}^{N} \mathcal{N}\big(\mathbf{x}_{k-1};\, \mathbf{x}^i_{k-1|k-1},\, N\mathbf{P}^{(N)}_{k-1|k-1}\big) \tag{20}$$

Taking N samples from the linear Gaussian Markov kernel yields a similar representation of the transition density:

$$p(\mathbf{x}_k|\mathbf{x}_{k-1}) \approx \prod_{i=1}^{N} \big\{\mathcal{N}\big(\mathbf{x}_k;\, \mathbf{F}_{k|k-1}\mathbf{x}_{k-1} + \mathbf{w}^i_{k|k-1},\, N\mathbf{Q}_{k|k-1}\big)\big\}. \tag{21}$$

As a consequence, the prior density is obtained by an application of the product formula which now yields exactly the parameters given in (14) and (15).

The benefit of this Gaussian product representation for the sampling process is as follows. If it is required to compute an approximation of the posterior which includes the processing of a current measurement $\mathbf{z}_k$, the transition from time t_{k-1} to time t_k is obtained up to a normalization constant factor by

$$p(\mathbf{x}_k|\mathcal{Z}^{k}) \propto \int \mathrm{d}\mathbf{x}_{k-1}\, p(\mathbf{x}_k|\mathbf{x}_{k-1})\, p(\mathbf{z}_k|\mathbf{x}_k) \cdot \frac{1}{N}\sum_{i=1}^{N} \delta(\mathbf{x}_{k-1} - \mathbf{x}^i_{k-1|k-1}). \tag{22}$$

We can now take samples from both, the Markov transition model as well as the measurement model where the latter is given by

$$p(\mathbf{z}_k|\mathbf{x}_k) = \prod_{i=1}^{N} \mathcal{N}\big(\mathbf{z}_k + \mathbf{u}^i_k;\, \mathbf{h}_k(\mathbf{x}_k),\, N\mathbf{R}_k\big), \tag{23}$$

and

$$\mathbf{u}^i_k \sim \mathcal{N}\big(\mathbf{O},\, \mathbf{R}_k\big). \tag{24}$$

Using the sampled Gaussian products from (21) and (23) yields an N–factorized version of the posterior of the Bayesian posterior:

$$p(\mathbf{x}_k|\mathcal{Z}^{k}) \propto \int \mathrm{d}\mathbf{x}_{k-1} \prod_{i=1}^{N} \Big\{\mathcal{N}\big(\mathbf{x}_k;\, \mathbf{F}_{k|k-1}\mathbf{x}_{k-1} + \mathbf{w}^i_{k|k-1},\, N\mathbf{Q}_{k|k-1}\big) \cdot \mathcal{N}\big(\mathbf{z}_k + \mathbf{u}^i_k;\, \mathbf{h}_k(\mathbf{x}_k),\, N\mathbf{R}_k\big) \cdot \mathcal{N}\big(\mathbf{x}_{k-1};\, \mathbf{x}^i_{k-1|k-1},\, N\mathbf{P}^N_{k-1|k-1}\big)\Big\}, \tag{25}$$

IV. Ensemble Kalman Filter

Based on the considerations in the previous section, it is useful as a "sanity check" to consider for a moment that a linear scenario is given, that is that the likelihood has a linear dependency on the state:

$$p(\mathbf{z}_k|\mathbf{x}_k) = \mathcal{N}\big(\mathbf{z}_k;\, \mathbf{H}_k\mathbf{x}_k,\, \mathbf{R}_k\big). \tag{26}$$

As now Kalman filter conditions are satisfied, it is possible to apply the product formula from the Appendix on (25):

$$p(\mathbf{x}_k|\mathcal{Z}^k) \propto \int \mathrm{d}\mathbf{x}_{k-1} \prod_{i=1}^{N} \Big\{ \mathcal{N}\big(\mathbf{x}_k;\, \mathbf{x}^i_{k|k-1},\, N\mathbf{P}^{(N)}_{k|k-1}\big), \cdot \mathcal{N}\big(\mathbf{z}_k + \mathbf{u}^i_k;\, \mathbf{H}_k\mathbf{x}_k,\, N\mathbf{R}_k\big) \cdot \mathcal{N}\big(\mathbf{x}_{k-1};\, \bar{\mathbf{x}}^i_{k-1|k-1},\, N\bar{\mathbf{P}}^{(N)}_{k-1|k-1}\big) \Big\}, \quad (27)$$

$$= \prod_{i=1}^{N} \Big\{ \mathcal{N}\big(\mathbf{x}_k;\, \mathbf{x}^i_{k|k-1},\, N\mathbf{P}^{(N)}_{k|k-1}\big) \mathcal{N}\big(\mathbf{z}_k + \mathbf{u}^i_k;\, \mathbf{H}_k\mathbf{x}_k,\, N\mathbf{R}_k\big) \Big\}, \quad (28)$$

where

$$\mathbf{x}^i_{k|k-1} = \mathbf{F}_{k|k-1}\mathbf{x}^i_{k-1|k-1} + \mathbf{w}^i_{k|k-1}, \quad (29)$$

$$\mathbf{P}^{(N)}_{k|k-1} = \mathbf{F}_{k|k-1}\mathbf{P}^{(N)}_{k-1|k-1}\mathbf{F}^\top_{k|k-1} + \mathbf{Q}_{k|k-1}, \quad (30)$$

$$\bar{\mathbf{x}}^i_{k-1|k-1} = \mathbf{x}^i_{k-1|k-1} + \bar{\mathbf{W}}(\mathbf{x}_k - \mathbf{F}_{k|k-1}\mathbf{x}^i_{k-1|k-1} - \mathbf{w}^i_{k|k-1}), \quad (31)$$

$$\bar{\mathbf{P}}^{(N)}_{k-1|k-1} = \mathbf{P}^{(N)}_{k-1|k-1} - \bar{\mathbf{W}}\mathbf{P}^{(N)}_{k|k-1}\bar{\mathbf{W}}^\top \quad (32)$$

Another application of the product formula on each of the factors yields

$$p(\mathbf{x}_k|\mathcal{Z}^k) \propto \prod_{i=1}^{N} \mathcal{N}\big(\mathbf{x}_k;\, \mathbf{x}^i_{k|k},\, N\mathbf{P}^N_{k|k}\big), \quad (33)$$

where

$$\mathbf{x}^i_{k|k} = \mathbf{x}^i_{k|k-1} + \mathbf{W}_{k|k-1}(\mathbf{z}_k - \mathbf{H}_k\mathbf{x}^i_{k|k-1} + \mathbf{u}^i_k), \quad (34)$$

$$\mathbf{W}_{k|k-1} = \mathbf{P}^{(N)}_{k|k-1}\mathbf{H}^\top_k\mathbf{S}^{-1}_k, \quad (35)$$

$$\mathbf{S}_k = \mathbf{H}_k\mathbf{P}^{(N)}_{k|k-1}\mathbf{H}^\top_k + \mathbf{R}_k. \quad (36)$$

By undoing the transformation from a particle representation of a Gaussian density into the Gaussian product in the previous section, the state samples for the posterior particles are directly obtained:

$$p(\mathbf{x}_k|\mathcal{Z}^k) = \frac{1}{N}\sum_{i=1}^{N} \delta(\mathbf{x}_k - \mathbf{x}^i_{k|k}). \quad (37)$$

As provided in the Appendix and also derived in [8], it is a straightforward calculation to show that first and second moments of this particle cloud matches with the exact solution given by the Kalman filter mean and covariance, respectively. This procedure can be interpreted as a flow of particles from time t_{k-1} to time t_k as schematically displayed in Figure 1.

For the non–linear case let the likelihood be given by the Gaussian density

$$p(\mathbf{z}_k|\mathbf{x}_k) = \mathcal{N}\big(\mathbf{x}_k;\, \mathbf{h}_k(\mathbf{x}_k),\, \mathbf{R}_k\big). \quad (38)$$

For the computation of the posterior particle samples, the *best linear unbiased estimator* (BLUE) [12] as in the linear case can be used, although it is well–known that the result is a sub–optimal approximation of the true filtered density. It should be noted that the term 'linear' refers to the estimation scheme, that is, that the estimated parameter is given by a linear combination of the predicted state and the innovation. As in the *Unscented Kalman filter* (UKF), an analytic derivative of the measurement function can be avoided.

According *fundamental equations of linear estimation* [12] the optimal gain matrix $\mathbf{W}_{k|k-1}$ is given by

$$\mathbf{W}_{k|k-1} = \mathbf{P}_{xz}\mathbf{P}^{-1}_{zz}, \quad (39)$$

where

$$\mathbf{P}_{xz} = \mathrm{cov}\left[\mathbf{x}_{k|k-1}, \mathbf{z}_k|\mathcal{Z}^{k-1}\right], \quad (40)$$

$$\mathbf{P}_{zz} = \mathrm{cov}\left[\mathbf{z}_k|\mathcal{Z}^{k-1}\right]. \quad (41)$$

Basically, there are two different methods to compute these matrices which in the end yields two different algorithms [13].

a) Sampled Cross–Covariance: Both required covariances can be computed by using the particle representation of the prior density $p(\mathbf{x}_k|\mathcal{Z}^{k-1})$:

$$\mathbf{P}^{(N)}_{xz} = \tfrac{1}{N-1}\textstyle\sum_{i=1}^{N}(\mathbf{x}^i_{k-1|k-1} - \mathbf{x}^{(N)}_{k|k-1})(\mathbf{z}_k - \mathbf{h}_k(\mathbf{x}^i_{k|k-1}))^\top \quad (42)$$

$$\mathbf{P}^{(N)}_{zz} = \tfrac{1}{N-1}\textstyle\sum_{i=1}^{N}(\mathbf{z}_k - \mathbf{h}_k(\mathbf{x}^i_{k|k-1}))(\mathbf{z}_k - \mathbf{h}_k(\mathbf{x}^i_{k|k-1}))^\top \quad (43)$$

This yields for the ith sample of the posterior distribution the following update scheme:

$$\mathbf{x}^i_{k|k} = \mathbf{x}^i_{k|k-1} + \mathbf{W}^{(N)}_{k|k-1}(\mathbf{z}_k - \mathbf{h}_k(\mathbf{x}^i_{k|k-1}) + \mathbf{u}^i_k), \quad (44)$$

where

$$\mathbf{W}^{(N)}_{k|k-1} = \mathbf{P}^{(N)}_{xz}(\mathbf{P}^{(N)}_{zz})^{-1}. \quad (45)$$

We will refer to this algorithm as the "Sampled EnKF".

b) Analytic First Order Cross–Covariance: In the linear case, the cross–covariance of the prior estimate and the measurement is given by

$$\mathbf{P}_{xz} = \mathrm{cov}\left[\mathbf{x}_{k|k-1}, \mathbf{z}_k|\mathcal{Z}^{k-1}\right] \quad (46)$$

$$= \mathrm{cov}\left[\mathbf{x}_{k|k-1}, \mathbf{H}_k\mathbf{x}_k + \mathbf{v}_k|\mathcal{Z}^{k-1}\right] \quad (47)$$

$$= \mathbf{P}_{k|k-1}\mathbf{H}^\top_k \quad (48)$$

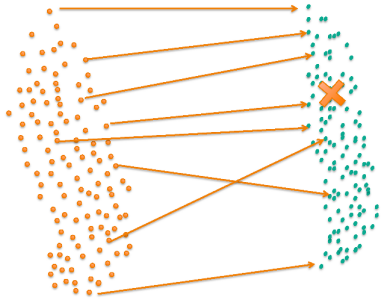

Fig. 1. Schematic figure of the transition of particles from time t_{k-1} (orange) to t_k (green). Each particles moves according to the dynamics model and is then subject to a Kalman filter where the measurement (cross) is artificially disturbed.

since the measurement noise $\mathbf{v}_k$ is independent of the prior estimation error. In the non–linear case, the measurement function $\mathbf{h}_k(\cdot)$ can be replaced by a first order Taylor approximation taken at the ith particle:

$$\mathbf{H}_k^i = D\mathbf{h}\Big|_{\mathbf{x}=\mathbf{x}_{k|k-1}^i}. \tag{49}$$

This approximation yields the following gain matrix $\mathbf{W}_{k|k-1}^i$:

$$\mathbf{W}_{k|k-1}^i = \mathbf{P}_{k|k-1}^{(N)} \mathbf{H}_k^{i\top} (\mathbf{H}_k^i \mathbf{P}_{k|k-1}^{(N)} \mathbf{H}_k^{i\top} + \mathbf{R}_k)^{-1}. \tag{50}$$

Thus, this approach is similar to an EKF for each particle, however, it should be noted that the Jacobian is taken at N potentially distinct points which yields a lower approximation error than a single linearization. However, this algorithm has higher computation time compared to the first approach for the very same reason.

V. Gaussian Mixture Ensemble Kalman Filter

From the considerations above, it becomes clear that the EnKF is based on the assumption of a Gaussian distributed prior density $p(\mathbf{x}_k|\mathcal{Z}^{k-1})$. Whenever the likelihood of previous measurements has been non–linear, this assumption is not given anymore. We therefore propose to approximate the prior by a Gaussian mixture:

$$p(\mathbf{x}_k|\mathcal{Z}^{k-1}) = \sum_{j=1}^{M_k} p^j \, \mathcal{N}\big(\mathbf{x}_k;\, \mathbf{x}_{k|k-1}^{(j)},\, \mathbf{P}_{k|k-1}^{(j)}\big), \tag{51}$$

where M_k, $\{p^j\}_j$, $\{\mathbf{x}_{k|k-1}^j\}_j$, and $\{\mathbf{P}_{k|k-1}^j\}_j$ are parameters which are to be estimated. This data clustering problem can be solved by the *Expectation Maximization* (EM) algorithm [11]. The result is a soft data assignment of each prior particle to all of the clusters:

$$\mathbf{x}_{k|k-1}^{(j)} = \sum_{i=1}^{N} w^{i,j} \mathbf{x}_{k|k-1}^i, \tag{52}$$

$$\mathbf{P}_{k|k-1}^{(j)} = \frac{1}{N-1} \sum_{i=1}^{N} w^{i,j} \cdot (\mathbf{x}_{k|k-1}^i - \mathbf{x}_{k|k-1}^{(j)})(\mathbf{x}_{k|k-1}^i - \mathbf{x}_{k|k-1}^{(j)})^\top, \tag{53}$$

$$p^j = \sum_{i=1}^{N} w^{i,j} \tag{54}$$

where $w^{i,j}$ is the probability of particle i being assigned to cluster j. This soft assignment can be used to obtain a linear combination of all gain matrices for each of the clusters. Thus,

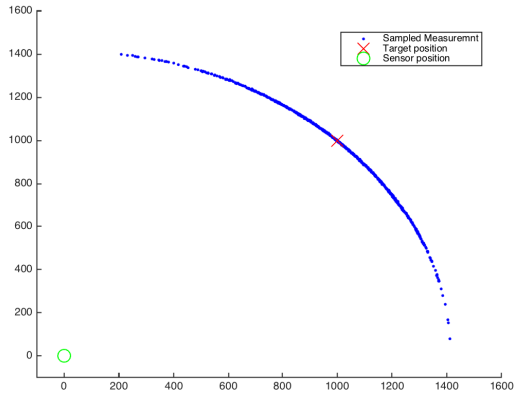

Fig. 2. In this figure, 1000 samples from the non–linear measurement likelihood were taken. The large standard deviation of 13.11° in bearing and low noise of $1m$ in range leads to a "banana shaped" likelihood when transformed into Cartesian coordinates.

for the jth cluster one obtains

$$\mathbf{P}_{xz}^{(N,j)} = \frac{1}{N-1} \sum_{i=1}^{N} w^{i,j} \cdot (\mathbf{x}_{k-1|k-1}^i - \mathbf{x}_{k|k-1}^{(N)})(\mathbf{z}_k - \mathbf{h}_k(\mathbf{x}_{k|k-1}^i))^\top \tag{55}$$

$$\mathbf{P}_{zz}^{(N,j)} = \frac{1}{N-1} \sum_{i=1}^{N} w^{i,j} \cdot (\mathbf{z}_k - \mathbf{h}_k(\mathbf{x}_{k|k-1}^i))(\mathbf{z}_k - \mathbf{h}_k(\mathbf{x}_{k|k-1}^i))^\top \tag{56}$$

$$\mathbf{W}_{k|k-1}^{(N,i)} = \sum_{j=1}^{M_k} w^{i,j} \mathbf{P}_{xz}^{(N,j)} (\mathbf{P}_{zz}^{(N,j)})^{-1}. \tag{57}$$

This yields for the ith sample of the posterior distribution the following update scheme:

$$\mathbf{x}_{k|k}^i = \mathbf{x}_{k|k-1}^i + \mathbf{W}_{k|k-1}^{(N,i)} (\mathbf{z}_k - \mathbf{h}_k(\mathbf{x}_{k|k-1}^i) + \mathbf{u}_k^i). \tag{58}$$

VI. Numerical Evaluation

In this section the EnKF and EM-EnKF filters are evaluated and compared to other filtering algorithms. At first, a linear scenario is considered to show that the result of the EnKF is close to the optimal solution given by the Kalman filter estimate. Then, a tracking scenario with a two–dimensional range–bearing sensor is simulated. The problem is non–linear and also non–Gaussian in Cartesian coordinates since a large standard deviation in cross–range and a rather low standard deviation in range lead to "banana shaped" densities. This can be seen in Figure 2, where 1000 samples from the measurement distribution for a fixed target position were taken.

Linear Scenario: In the linear scenario, the two dimensional position of a target is measured with a single sensor once per second. The target is located at the origin and goes along a random trajectory according to the Discretized White Noise Acceleration (DWNA) model with an initial

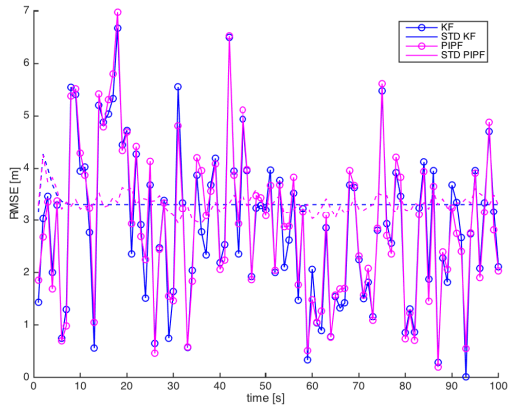

Fig. 3. RMSE for the linear scenario: The RMSE for a single simulation run shows that the estimate and the standard deviation (STD) of the estimate are close to an optimal Kalman filter (KF).

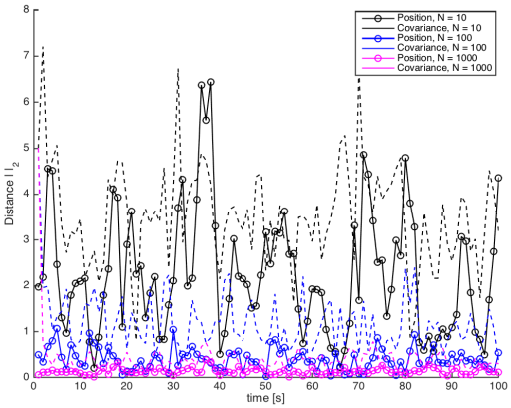

Fig. 4. The convergence towards the probability density function of the Kalman filter is shown. For an increasing number N of particles, the $|\cdot|_2-$ norm of the difference in the position estimates as well as the norm of the difference in the estimation error covariances tend to zero.

velocity of $1\frac{m}{s}$ and a noise density with $q = 1.0\frac{m^2}{s^3}$. The position measurements are disturbed by an additive, zero–mean Gaussian noise with a variance of $10m^2$ in $x-$ and $y-$ coordinates, respectively. The EnKF was set up with 100 particles only.

In Figure 3, the estimation error of the target position for a single simulation run is shown. One can see that the estimation quality is close to equal. This is particularly stressed in Figure 4. For an increasing number of particles $N = 10$ (black), $N = 100$ (blue), $N = 1000$ (magenta), the distance to the optimal posterior density given by the Kalman filter result is compared at each time step of a single simulation run. The lines with circles are the Euclidean distances in the position estimates whereas the slashed lines are the distances in the matrix 2–norm for the estimation error covariances.

Non–Linear Scenario: For the non–linear case a two dimensional range–bearing sensor located at the origin was simulated. Once per second the sensor is taking a measurement of the target which starts at $(1000, 1000)^\top$ with a velocity vector $(-5, 5)^\top$. The measurements are corrupted by an additive Gaussian distributed zero–mean noise with standard deviation $\sigma_r = 1m$ in range and $\sigma_\theta = 13.11°$ in bearing, respectively. This rather low noise in range and high noise in cross–range is also known as the *banana problem* [14]. The EM-EnKF is compared to the EnKF, standard SIR particle filter (SIR–PF), and an extended Kalman Filter (EKF). Both, the SIR–PF is using $N = 1000$ particles, whereas the EnKF and the EM-EnKF are using 200 particles only. In Figure 6, the results of the root mean squared error (RMSE) of 100 Monte–Carlo runs are shown. For all kind of particle filters, it is crucial to compare the processing time, too, as increasing the number of particles will lead to lower RMSEs but also to a higher processing time. The computational load of the compared algorithms is shown in Figure 5.

It can be seen that the EKF is extremely fast due to its analytic solution, which is well–known. However, the EKF also has the worst estimation error of all algorithms. The average performace of the EnKF is comparable to the SIR–PF while there is a significant reduction in the processing time compared to the SIR–PF. The estimation performance can be improved by using the EM induced Gaussian mixture approximation as proposed in this paper. This also leads to higher computation time, however, the resulting algorithm still is faster than a SIR–PF. This is due to the fact that no resampling is required for the ensemble filters.

VII. Conclusion

The Ensemble Kalman Filter (EnKF) is a Kalman based particle filter which was introduced to solve large scale data assimilation problems where the state space is of very large dimensionality. It also achieves good results when applied

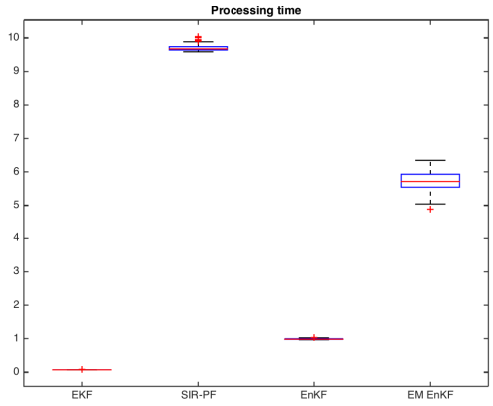

Fig. 5. Box–plot of the processing time for a single simulation run of 100 time steps for each algorithm.

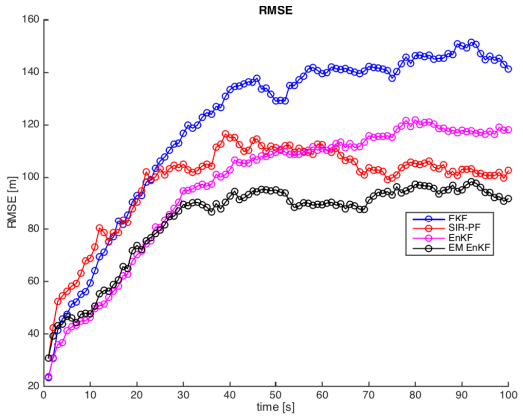

Fig. 6. RMSE of the EKF, SIR–PF, EnKF, and EM–EnKF where a sampled gain was used for the Ensemble filters.

to a target tracking problem, however, due to its Gaussian assumption for the prior density, the performance can be improved by introducing Gaussian mixtures. In this paper, a new derivation of the EnKF is presented which is based on a duality between Gaussian products and particle densities. A relaxation of the Gaussian assumption is then achieved by introducing a particle clustering into Gaussian Mixtures by means of the Expectation Maximization (EM) algorithm and to apply the EnKF on the clusters. The soft assignment of the EM allows all Gaussian components to contribute to each of the particles. The higher computation time for this algorithm is compensated by the fact that resampling is avoided as impoverishment of weights does not occur. In the evaluation, it was shown that the EM–EnKF particle filter has lower computation time and lower estimation error in comparison to a standard resampling particle filter.

ACKNOWLEDGEMENTS

The authors would like to thank Uwe Hanebeck for his valuable comments on this work. Peter Willett was supported by ONR under contract N00014-13-1-0231 and by NPS NPS via ONR N00244-14-1-0033.

APPENDIX

Product Formula for Gaussians in Identical Variables

Given two Gaussians in the same variable $\mathbf{x}$, the following equation holds:

$$\mathcal{N}(\mathbf{x};\, \mathbf{a},\, \mathbf{A})\; \mathcal{N}(\mathbf{x};\, \mathbf{b},\, \mathbf{B}) = \mathcal{N}(\mathbf{a};\, \mathbf{b},\, \mathbf{A}+\mathbf{B}) \cdot \mathcal{N}(\mathbf{x};\, \mathbf{y},\, \mathbf{P}), \tag{59}$$

where the abbreviations $\mathbf{y}$ and $\mathbf{P}$ are given by

$$\mathbf{P} = \left(\mathbf{A}^{-1}+\mathbf{B}^{-1}\right)^{-1} \text{ and } \mathbf{y} = \mathbf{P}\left(\mathbf{A}^{-1}\mathbf{a}+\mathbf{B}^{-1}\mathbf{b}\right). \tag{60}$$

Product Formula for Linearly Conditional Gaussians

For matrices of suitable dimensions the following formula for products of Gaussians holds:

$$\mathcal{N}(\mathbf{z};\, \mathbf{H}\mathbf{x},\, \mathbf{R})\,\mathcal{N}(\mathbf{x};\, \mathbf{y},\, \mathbf{P}) = \mathcal{N}(\mathbf{z};\, \mathbf{H}\mathbf{y},\, \mathbf{S}) \begin{cases} \mathcal{N}(\mathbf{x};\, \mathbf{y}+\mathbf{W}\boldsymbol{\nu},\, \mathbf{P}-\mathbf{W}\mathbf{S}\mathbf{W}^\top) \\ \mathcal{N}(\mathbf{x};\, \mathbf{Q}(\mathbf{P}^{-1}\mathbf{y}+\mathbf{H}^\top\mathbf{R}^{-1}\mathbf{z}),\, \mathbf{Q}) \end{cases} \tag{61}$$

with the following abbreviations:

$$\boldsymbol{\nu} = \mathbf{z}-\mathbf{H}\mathbf{y} \tag{62}$$

$$\mathbf{S} = \mathbf{H}\mathbf{P}\mathbf{H}^\top+\mathbf{R} \tag{63}$$

$$\mathbf{W} = \mathbf{P}\mathbf{H}^\top\mathbf{S}^{-1} \tag{64}$$

$$\mathbf{Q}^{-1} = \mathbf{P}^{-1}+\mathbf{H}^\top\mathbf{R}^{-1}\mathbf{H}. \tag{65}$$

Parameter Convergence in the Linear Gaussian Case

In the section, the convergence of the posterior particle representation to the Kalman filter mean and covariance is shown. Since the prediction step is equivalent to the standard prediction for particle filters, we restrict ourselves to the update step. According to the update formula of all particles in (34), the mean of the posterior distribution is given by

$$\mathbf{x}_{k|k}^{(N)} = \frac{1}{N}\sum_{i=1}^{N}\mathbf{x}_{k|k}^{i} \tag{66}$$

$$= \frac{1}{N}\sum_{i=1}^{N}\mathbf{x}_{k|k-1}^{i}+\mathbf{W}_{k|k-1}(\mathbf{z}_k-\mathbf{H}_k\mathbf{x}_{k|k-1}^{i}+\mathbf{u}_k^{i}) \tag{67}$$

where $\mathbf{W}_{k|k-1}$ is the Kalman gain and the drawn noise $\mathbf{u}_k^i$ is zero–mean. According to the assumption of converging prior parameters, one obtains

$$\lim_{N\to\infty}\mathbf{x}_{k|k}^{(N)} = \mathbf{x}_{k|k-1}+\mathbf{W}_{k|k-1}(\mathbf{z}_k-\mathbf{H}_k\mathbf{x}_{k|k-1}) \tag{68}$$

$$= \mathbf{x}_{k|k}. \tag{69}$$

Similarly, the covariance matches the true covariance of the optimal estimate given that enough particles are used. This can be seen by using the update equation, again. One obtains for the covariance

$$\mathbf{P}_{k|k}^{(N)} = \frac{1}{N-1}\sum_{i=1}^{N}(\mathbf{x}_{k|k}^{i}-\mathbf{x}_{k|k}^{(N)})^2 \tag{70}$$

where the abbreviation $(\cdot)^2 = (\cdot)(\cdot)^\top$ is used. Since it is assumed that the prior samples are distributed with covariance $\mathbf{P}_{k|k-1}$ the covariance of the posterior distribution equals by taking the limit $N\to\infty$:

$$\mathbf{P}_{k|k}^{(N)} = \mathbf{P}_{k|k-1}-2\mathbf{P}_{k|k-1}\mathbf{H}_k^\top\mathbf{S}_k^{-1}\mathbf{H}_k\mathbf{P}_{k|k-1} + \mathbf{P}_{k|k-1}\mathbf{H}_k^\top\mathbf{S}_k^{-1}(\mathbf{H}_k\mathbf{P}_{k|k-1}\mathbf{H}_k^\top+\mathbf{R}_k)\mathbf{S}_k^{-1}\mathbf{H}_k\mathbf{P}_{k|k-1} \tag{71}$$

$$= \mathbf{P}_{k|k-1}-\mathbf{W}_{k|k-1}\mathbf{S}_k\mathbf{W}_{k|k-1}^\top \tag{72}$$

$$= \mathbf{P}_{k|k}. \tag{73}$$

This concludes the proof.

REFERENCES

[1] W. Koch, *Tracking and Sensor Data Fusion: Methodological Framework and Selected Applications*. Springer, 2014.

[2] B. Ristic, S. Arulampalam, and N. Gordon, *Beyond the Kalman Filter: Particle Filters for Tracking Applications*. Artech House, 2004.

[3] N. Gordon, D. Salmond, and A. Smith, "Novel approach to nonlinear/non-gaussian bayesian state estimation," *Radar and Signal Processing, IEE Proceedings F*, vol. 140, no. 2, pp. 107–113, Apr 1993.

[4] F. Daum and J. Huang, "Particle flow for nonlinear filters, bayesian decisions and transport," in *Information Fusion (FUSION), 2013 16th International Conference on*, July 2013, pp. 1072–1079.

[5] M. Khan and M. Ulmke, "Non-linear and non-gaussian state estimation using log-homotopy based particle flow filters," in *Sensor Data Fusion: Trends, Solutions, Applications (SDF), 2014*, Oct 2014, pp. 1–6.

[6] U. Hanebeck, "Pgf 42: Progressive gaussian filtering with a twist," in *Information Fusion (FUSION), 2013 16th International Conference on*, July 2013, pp. 1103–1110.

[7] T. Yang, P. Mehta, and S. Meyn, "Feedback particle filter," *Automatic Control, IEEE Transactions on*, vol. 58, no. 10, pp. 2465–2480, Oct 2013.

[8] G. Evensen, "Sequential data assimilation with a nonlinear quasi-geostrophic model using monte carlo methods to forecast error statistics," *J. Geophys. Res*, vol. 99, pp. 10 143–10 162, 1994.

[9] S. Reich, "A Gaussian-mixture ensemble transform filter," *Quarterly Journal of the Royal Meteorological Society*, vol. 138, pp. 222–233, Jan. 2012.

[10] M. Glodek, M. Schels, and F. Schwenker, "Ensemble gaussian mixture models for probability density estimation.(report)," *Computational Statistics*, vol. 28, no. 1, p. 127(12), 2013-02-01.

[11] M. A. Tanner, *Tools for statistical inference : methods for the exploration of posterior distributions and likelihood functions*, ser. Springer series in statistics. New York: Springer, 1996. [Online]. Available: http://opac.inria.fr/record=b1077915

[12] Y. Bar-Shalom, X. Li, and T. Kirubarajan, *Estimation with Applications to Tracking and Navigation*. Wiley-Interscience, 2001.

[13] A. Shahid, D. Ustebay, and M. Coates, "Distributed ensemble kalman filtering," in *Sensor Array and Multichannel Signal Processing Workshop (SAM), 2014 IEEE 8th*, June 2014, pp. 217–220.

[14] K. Romeo, P. Willett, and Y. Bar-Shalom, "Particle filter tracking for the banana problem," pp. 885 709–885 709–19, 2013. [Online]. Available: http://dx.doi.org/10.1117/12.2023564