The Sliced Gaussian Mixture Filter for Efficient Nonlinear Estimation

Vesa Klumpp, Felix Sawo, and Uwe D. Hanebeck
Intelligent Sensor-Actuator-Systems Laboratory
Institute of Computer Science and Engineering
Universität Karlsruhe (TH), Germany
Email: {klumpp, sawo}@ira.uka.de, uwe.hanebeck@ieee.org

Dietrich Fränken
Data Fusion Algorithms and Software
EADS Deutschland GmbH
D-89077 Ulm, Germany
Email: dietrich.fraenken@eads.com

***Abstract*—This paper addresses the efficient state estimation for mixed linear/nonlinear dynamic systems with noisy measurements. Based on a novel density representation – *sliced Gaussian mixture density* – the decomposition into a (conditionally) linear and nonlinear estimation problem is derived. The systematic approximation procedure minimizing a certain distance measure allows the derivation of (close to) optimal and deterministic estimation results. This leads to high-quality representations of the measurement-conditioned density of the states and, hence, to an overall more efficient estimation process. The performance of the proposed estimator is compared to state-of-the-art estimators, like the well-known marginalized particle filter.**

Keywords: Nonlinear estimation, state space decomposition, sliced densities.

I. Introduction

The estimation of a system's state from noisy measurements is a common task in many applications. In some special cases, closed-form solutions to the estimation problem can be found. The most famous example is the case of linear systems with additive Gaussian noise where the Kalman filter provides the optimal solution (in the minimum mean square error sense). A lot of proposals have been made to find at least approximate solutions in more general (especially nonlinear) cases like, e. g, the extended Kalman filter, which is based on a linearization of the system equations, or the unscented Kalman filter, where a selected set of states is used to cover the second order statistics of the estimate. General applicability and ease of implementation have made particle filters (which track the distribution of the estimate by a number of samples) quite popular.

It has been shown that the performance of such general-purpose *nonlinear estimators* can be improved significantly with respect to both accuracy and implementation effort if the system model includes a linear substructure. In this case, a *decomposition* of the entire estimation problem into a (conditionally) *linear* problem and a *nonlinear* problem allows for an overall more efficient estimation process. Conditionally linear models arise in various applications, such as in positioning, navigation, and tracking [1], robot mapping and localization, speech processing [2], or the simultaneous state and parameter estimation of distributed systems [3], [4], just to name a few.

One possible approach to solve the combined linear/nonlinear estimation problem is the marginalized (or Rao-Blackwellized) particle filter [1], [2], [5]–[7]. Herein, marginalization over the linear subspace is used to reduce the dimensionality of the state space and the remaining density is subsequently represented by particles. This way, the standard particle filter is adopted to cope with the reduced nonlinear problem only. Once the history of particles is given, the Kalman filter is used to find the optimal estimate for the linear subspace associated with each individual particle.

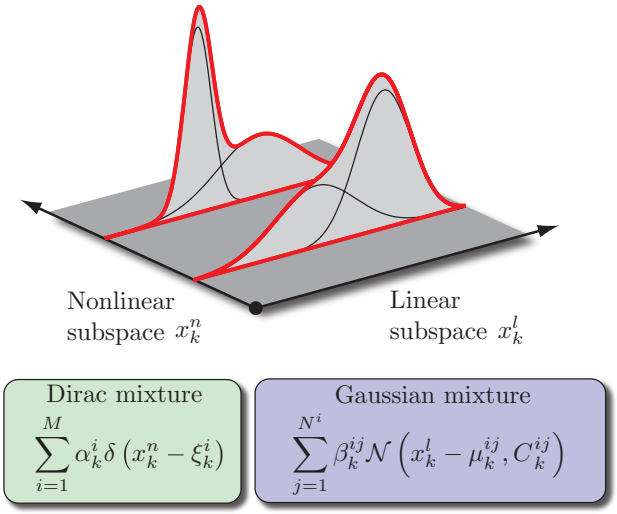

Figure 1. Visualization of a *sliced Gaussian mixture density* consisting of a Gaussian mixture in x_k^l direction and a Dirac mixture in x_k^n direction.

Despite the improved performance of the marginalized filter in comparison with the standard particle filter, some drawbacks still remain. Just like with the standard filter, special measures have to be taken in order to avoid effects like sample degeneration and impoverishment. Although the filter yields estimates for both state and estimation error, it does not provide a measure on how well the estimated joint density actually represents the true one. Consequently, there appears to be no other way than simulation when it comes to determining which number of particles actually is high enough for good performances.

The framework of the marginalized particle filter as presented in [7] is restricted to Gaussian densities in the linear

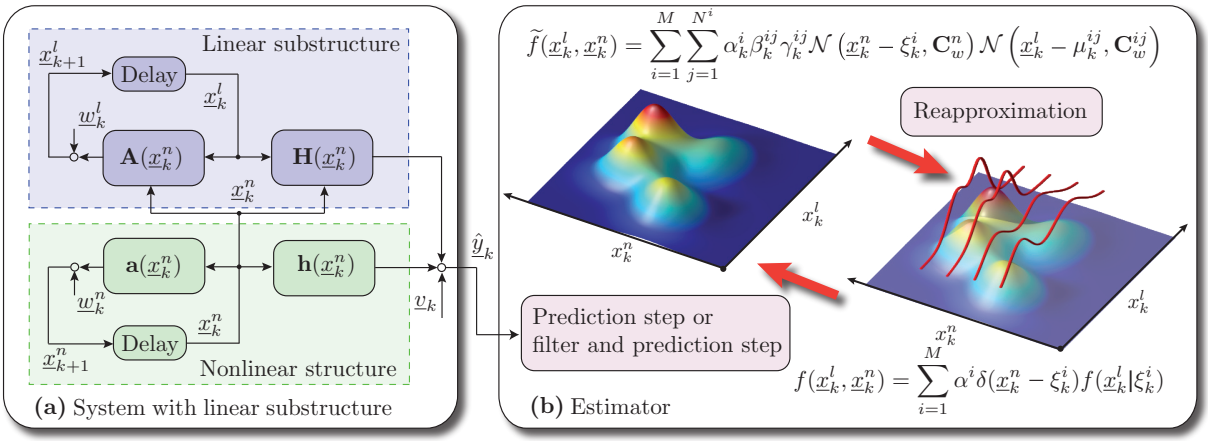

Figure 2. **(a)** Visualization of a dynamic system with a linear substructure. The nonlinear substate $\underline{x}^n_k$ characterizes the system and measurement matrices. For simplicity, the input $\underline{u}_k$ is omitted here. **(b)** Procedure of filter step and prediction step based on sliced Gaussian mixture density. The estimation result of the sliced Gaussian mixture density leads to a posterior Gaussian mixture density, which is then reapproximated.

subspace. Thus, several particles would be nececessary to represent densities that are multimodal in these dimensions. Under unfavorable circumstances, a loss of entire (possibly important) modes is possible due to the degeneration of single particles. In fact, it has been argued in [8] that particle filters are often incapable of maintaining multimodality and mixture tracking is proposed as a possible solution. However, that approach relies on clustering algorithms in order to reposition the particles of a completely discrete density representation and it is not clear how such an operation translates into the mixed representation of the marginalized particle filter. As it stands, there is no other interaction between the individual particles of the marginalized particle filter than stemming from the resampling step after the measurement update. In its standard implementation, this boils down to duplicating some samples while discarding others.

In this paper, a novel approach to recursively estimating the measurement-conditioned density of the states in a conditionally linear Gaussian system is presented. In addition to a decomposition of the state space into a linear and a nonlinear part (as in the marginalized particle filter), three key features lead to an estimation of significantly improved performance: **(a)** the usage of a special kind of density for approximation purposes, **(b)** a careful selection on exactly when an approximation is performed, and **(c)** a systematic approximation procedure that is (close to) optimal in a certain sense. To be more specific, a so-called *sliced Gaussian mixture density* is used as a density representation, see Fig. 1. Once the slices with the Gaussian mixture components have been positioned, the predicted density can be computed in an analytic fashion. Due to the conditionally linear dynamic model, this density turns out to be a Gaussian mixture whose parameters can be obtained by Kalman filter update and prediction steps. The subsequent reapproximation procedure can therefore take full advantage of the complete knowledge about the density to be approximated. The positioning of the slices is performed in a systematic (nonrandom) fashion that minimizes a specific distance measure between true and approximated density. Different approaches can be used for obtaining an (exactly or almost) optimal solution to the minimization problem. The final positions of the slices are not independent. Hence, a kind of interaction is enforced in an almost natural way within this framework.

The rest of the paper is organized as follows: Section II contains a rigorous formulation of the problem to be solved. In Section III, the *sliced Gaussian mixture densities* are introduced and it is shown how they can be used to approximate Gaussian mixture densities. Section IV describes the details of the recursive estimator based on this novel density representation. In Section V, we show some simulation results comparing the proposed estimator with a corresponding marginalized particle filter.

II. Problem Formulation

In this paper, we consider a nonlinear discrete-time dynamic system of the form

$$\underline{\boldsymbol{x}}_{k+1} = \underline{a}_k\left(\underline{\boldsymbol{x}}^l_k, \underline{\boldsymbol{x}}^n_k, \hat{\underline{u}}_k\right) + \underline{\boldsymbol{w}}_k \ , \tag{1}$$

with total random state vector $\underline{\boldsymbol{x}}_k$, input $\hat{\underline{u}}_k$ driving the sytem, the nonlinear system function $\underline{a}_k(\,\cdot\,)$, and additive white noise $\underline{\boldsymbol{w}}_k$. Furthermore, we consider the measurement equation

$$\underline{\boldsymbol{y}}_k = \underline{h}_k(\underline{\boldsymbol{x}}^l_k, \underline{\boldsymbol{x}}^n_k) + \underline{\boldsymbol{v}}_k \ , \tag{2}$$

where $\underline{\boldsymbol{y}}_k$ is the measurement value at discrete time step k. The nonlinear measurement function is denoted as $\underline{h}_k(\,\cdot\,)$, and the noise term $\underline{\boldsymbol{v}}_k$ is assumed to be white Gaussian noise. Note, that an actual measurement $\hat{\underline{y}}_k$ is a realization of the random variable $\underline{\boldsymbol{y}}_k$.

It is assumed that the nonlinear system function $\underline{a}_k(\,\cdot\,)$ and measurement function $\underline{h}_k(\,\cdot\,)$ contain a linear substructure allowing a decomposition of the total random state vector

$\underline{\boldsymbol{x}}_k \in \Omega$ into two substate vectors,

$$\underline{\boldsymbol{x}}_k = \left[(\underline{\boldsymbol{x}}_k^l)^T \quad (\underline{\boldsymbol{x}}_k^n)^T\right]^T \quad ,$$

with $\underline{\boldsymbol{x}}_k^l \in \mathbb{R}^r$ and $\underline{\boldsymbol{x}}_k^n \in \mathbb{R}^s$. The decomposition of the state vector is exploited to develop a more efficient estimator than an estimator working on the entire state vector. A dynamic system with a linear substructure is visualized in Fig. 2 (a).

The main goal of the estimator is to get an accurate estimation for $\underline{\boldsymbol{x}}_k$ at every time step k in terms of a density function $f(\underline{x}_k)$. In general, the estimator consists of two steps being performed alternately in order to derive the density function $f(\underline{x}_k)$ as precisely as possible, namely prediction step and filter step.

Prediction step: The purpose of the prediction step is to determine, for a given prior density $f^e(\underline{x}_k)$ for $\underline{\boldsymbol{x}}_k$, the predicted density $f^p(\underline{x}_{k+1})$ of $\underline{\boldsymbol{x}}_{k+1}$ for the next discrete time step. This can be achieved by evaluating the well-known Chapman-Kolmogorov equation

$$f^p(\underline{x}_{k+1}) = \int_\Omega f^T(\underline{x}_{k+1}|\underline{x}_k) f^e(\underline{x}_k) \mathrm{d}\underline{x}_k \quad , \tag{3}$$

where $f^T(\underline{x}_{k+1}|\underline{x}_k)$ is the so-called transition density depending on the dynamic system model.

Filter step: The information of measurements $\hat{\underline{y}}_k$ can be incorporated into the processing scheme in order to improve the estimation of $\underline{\boldsymbol{x}}_k$. The estimated density $f^e(\underline{x}_k)$ can be determined by the famous Bayes' formula

$$f^e(\underline{x}_k) = c_k \cdot f^L(\hat{\underline{y}}_k|\underline{x}_k) \cdot f^p(\underline{x}_k) \quad , \tag{4}$$

where $f^L(\hat{\underline{y}}_k|\underline{x}_k)$ is the so-called likelihood, which can be regarded as the conditional density for the occurrence of the measurement $\hat{\underline{y}}_k$ for given $\underline{x}_k$. The coefficient c_k is a normalization constant for the density product $f^L(\hat{\underline{y}}_k|\underline{x}_k) \cdot f^p(\underline{x}_k)$.

For arbitrary prior density functions and general dynamic systems, both prediction step (3) and filter step (4) cannot be solved analytically. Hence, approximation methods are inevitable in order to derive the density function $f(\underline{x}_k)$ of $\underline{\boldsymbol{x}}_k$. In the special case of linear systems and Gaussian densities, the estimation can be performed analytically. Furthermore, there exist many *nonlinear* systems with a *linear* substructure, for example the system depicted in Fig. 2 (a). For such systems, the prediction step (3) and filter step (4) can be calculated more efficiently by a decomposition into a *linear* and a *nonlinear* estimation problem. In this paper, the decomposition of the estimation problem is mainly achieved by a novel density representation, the so-called *sliced Gaussian mixture densities*.

III. Density Approximation

In this section, we introduce the *sliced Gaussian mixture density*. Furthermore, we explain how it can be used for approximating arbitrary densities in a systematic fashion.

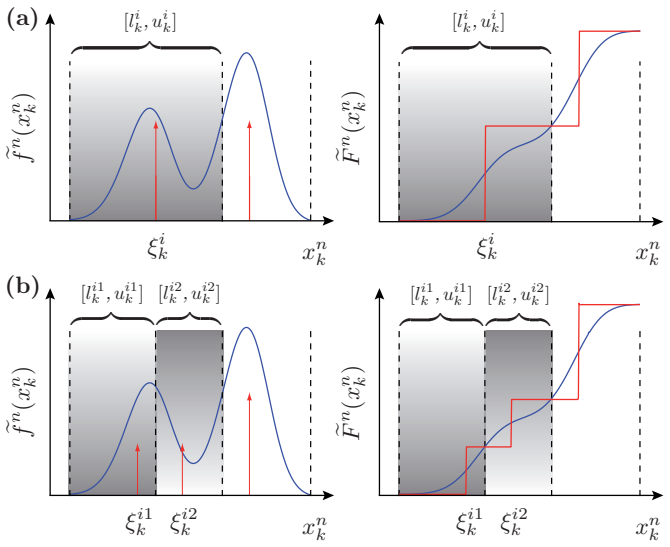

Figure 3. Visualization of the approximation process in the nonlinear subspace, and thus the determination of the position of the slices. The Dirac component at location ξ_k^i is replaced by two Dirac components at positions ξ_k^{i1} and ξ_k^{i2}.

A. Representation

The *sliced Gaussian mixture density* $f(\underline{x}_k)$ is represented by a Dirac mixture density in the *nonlinear* subspace $\underline{x}_k^n$ and Gaussian mixture density in the *linear* subspace $\underline{x}_k^l$,

$$f(\underline{x}_k^l, \underline{x}_k^n) = \sum_{i=1}^{M} \alpha_k^i \underbrace{\delta(\underline{x}_k^n - \underline{\xi}_k^i)}_{\textit{Dirac mixture}} \underbrace{f(\underline{x}_k^l|\underline{\xi}_k^i)}_{\text{Gaussian mixture}} \quad , \tag{5}$$

where $\underline{\xi}_k^i \in \mathbb{R}^s$ can be regarded as the position of the sliced density function $f(\underline{x}_k^l, \underline{x}_k^n)$ as shown in Figure 1.

The marginal density in the nonlinear subspace $\underline{x}_k^n$ is given by a Dirac mixture function according to

$$f(\underline{x}_k^n) = \sum_{i=1}^{M} \alpha_k^i \, \delta(\underline{x}_k^n - \underline{\xi}_k^i) \quad , \tag{6}$$

where α_k^i and $\underline{\xi}_k^i$ represent the weights and positions of the Dirac functions, respectively. The density representation along the individual slices is assumed to be a Gaussian mixture density

$$f(\underline{x}_k^l|\underline{\xi}_k^i) = \sum_{j=1}^{N^i} \beta_k^{ij} \mathcal{N}\left(\underline{x}_k^l - \underline{\mu}_k^{ij}, \mathbf{C}_k^{ij}\right) \quad , \tag{7}$$

with β_k^{ij}, $\underline{\mu}_k^{ij} \in \mathbb{R}^r$, and $\mathbf{C}_k^{ij} \in \mathbb{R}^{r \times r}$ denoting the weights, means, and covariance matrices of the i-th component of the Gaussian mixture density of the j-th slice.

In general, sliced Gaussian mixtures as given in (5) can be used for approximating arbitrary density functions arising in estimators for nonlinear dynamic systems. However, the full advantage of such a density representation can be exploited in the case of system structures allowing a decomposition into a *linear* and a *nonlinear* estimation problem, as shown in Sec. IV.

B. Approximation with consideration of marginal density

In the following, we will assume that the true density to be approximated is given as a Gaussian mixture density

$$\widetilde{f}(\underline{x}_k^l, \underline{x}_k^n) = \sum_{j=1}^{L} \widetilde{\alpha}_k^j \mathcal{N}\left(\underline{x}_k^l - \underline{\widetilde{\mu}}_k^j, \widetilde{\mathbf{C}}_k^{lj}\right) \mathcal{N}\left(\underline{x}_k^n - \underline{\widetilde{\xi}}_k^j, \widetilde{\mathbf{C}}_k^{nj}\right) \ ,$$

with weighting factors $\widetilde{\alpha}_k^j$, means $\underline{\widetilde{\mu}}_k^j$ and $\underline{\widetilde{\xi}}_k^j$, and covariance matrices $\widetilde{\mathbf{C}}_k^{lj}$ and $\widetilde{\mathbf{C}}_k^{nj}$. The individual Gaussian components, each represented as a product of two Gaussians, consist of mutually independent linear and nonlinear parts. From now on, we use the tilde to distinguish between true densities $\widetilde{f}$ and their approximations f.

Two options exist for the approximation of the true density $\widetilde{f}(\underline{x}_k^l, \underline{x}_k^n)$ by a *sliced Gaussian mixture density*. The first option is based on a distance measure over the entire state space. The second option just considers the marginal density $\widetilde{f}^n(\underline{x}_k^n)$ in the nonlinear subspace, which is the approach used in this paper. For both options, the derivation of the slice locations is performed by either a batch approximation or a sequential approximation. The *batch approximation* is an efficient solution procedure for arbitrary true densities on the basis of homotopy continuation (Progressive Bayes) and results in an optimal solution [9]. The *sequential greedy algorithm* for Dirac mixture approximation of a arbitrary densities is based on inserting one component at a time [10].

For simplicity and brevity only the case of the one-dimensional nonlinear subspace is considered here, i.e., $s = 1$. For that purpose, we consider a variation of the sequential greedy algorithm proposed in [10] for deriving the position of the slices that can easily be extended to several dimensions.

In the greedy algorithm for the density approximation, every Dirac component corresponds to an interval $[l_k^i, u_k^i]$ in the nonlinear subspace of the state space and approximates the true marginal density $\widetilde{f}^n(x_k^n)$ only in the corresponding interval, see Fig. 3. Basically, the density approximation is based on splitting the intervals and Dirac components. The entire algorithm consisting of the phases – initialization phase, identification phase, and splitting phase – is shown in Alg. 1. It takes the number of components M, the density $\widetilde{f}^n$ and the interval $[l_k^0, u_k^0]$ that will be approximated. The *initialization phase* (lines 2–5) is used to set up the data structure, containing the set of Dirac components, with one tuple. In the *identification phase* (lines 7–11), the specific interval for inserting new Dirac components is determined, based on the maximum deviation between the distributions of the true prior density and the approximating Dirac mixture. Here, the Cramér-von Mises distance D^n is used, which is calculated as

$$D^n = \frac{1}{2} \int_{\mathbb{R}} \left(\widetilde{F}^n(x_k^n) - F^n(x_k^n)\right)^2 \mathrm{d}x_k^n \ .$$

In this algorithm, the deviation measure is assumed to be dependent on the interval width $u_l^i - l_l^i$ and the weight α_k^i of the Dirac component. This is calculated for every Dirac component (line 9) and the interval with the highest deviation is chosen for the splitting phase. This interval corresponds to the region where the approximation deviates most from the true density. In the *splitting phase* (lines 12–18), the Dirac component with position ξ_k^i, weight α_k^i, and interval $[l_k^i, u_k^i]$ is replaced by two tuples. The new Dirac positions ξ_k^{i1} and ξ_k^{i2} are chosen to split the probability mass of the true marginal density over the intervals into halves (lines 15 & 16) using the subroutine `Solve`, that numerically solves the given equation. The splitting of one Dirac component is shown in Fig. 3. With an increasing number of Dirac components, the approximation converges towards the true marginal distribution.

Algorithm 1 Derivation of the positions ξ_k^i and weights α_k^i of the slices by considering the marginal density $\widetilde{f}^n$.

1: Function `Approximation`$(M, \widetilde{f}^n, l_k^0, u_k^0)$
2: // Initialization Phase
3: $\alpha_k^0 \leftarrow \int_{l_k^0}^{u_k^0} \widetilde{f}^n(x_k^n) \mathrm{d}x_k^n$
4: $\xi_k^0 \leftarrow$ `Solve` $\left[\xi_k^0, \int_{l_k^0}^{\xi_k^0} \widetilde{f}^n(x_k^n)\, \mathrm{d}x_k^n \stackrel{!}{=} \frac{\alpha_k^0}{2}\right]$
5: $\mathcal{A} \leftarrow \left\{\left(\xi_k^0, \alpha_k^0, l_k^0, u_k^0\right)\right\}$
6: **for** $c = 1$ to $M - 1$ **do**
7: // Identification Phase
8: **for all** $(\xi_k^j, \alpha_k^j, l_k^j, u_k^j)$ in $\mathcal{A}$ **do**
9: $D^n(j) \leftarrow (u_k^j - l_k^j) \cdot \alpha_k^j$
10: **end for**
11: $i \leftarrow \arg\max_j \{D^n(j)\}$
12: // Splitting Phase
13: $l_k^{i1} \leftarrow l_k^i$; $u_l^{i1} \leftarrow \xi_k^i$; $\alpha_k^{i1} \leftarrow \frac{\alpha_k^i}{2}$
14: $l_k^{i2} \leftarrow \xi_k^i$; $u_l^{i2} \leftarrow u_k^i$; $\alpha_k^{i2} \leftarrow \frac{\alpha_k^i}{2}$
15: $\xi_k^{i1} \leftarrow$ `Solve` $\left[\xi_k^{i1}, \int_{l_k^{i1}}^{\xi_k^{i1}} \widetilde{f}^n(x_k^n)\, \mathrm{d}x_k^n \stackrel{!}{=} \frac{\alpha_k^{i1}}{2}\right]$
16: $\xi_k^{i2} \leftarrow$ `Solve` $\left[\xi_k^{i2}, \int_{l_k^{i2}}^{\xi_k^{i2}} \widetilde{f}^n(x_k^n)\, \mathrm{d}x_k^n \stackrel{!}{=} \frac{\alpha_k^{i2}}{2}\right]$
17: $\mathcal{A} \leftarrow \mathcal{A} \setminus \left\{\left(\xi_k^i, \alpha_k^i, l_k^i, u_k^i\right)\right\}$
18: $\mathcal{A} \leftarrow \mathcal{A} \cup \left\{\left(\xi_k^{i1}, \alpha_k^{i1}, l_k^{i1}, u_k^{i1}\right), \left(\xi_k^{i2}, \alpha_k^{i2}, l_k^{i2}, u_k^{i2}\right)\right\}$
19: **end for**
20: **return** $\mathcal{A}$

After the approximation of the marginal density $\widetilde{f}^n(x_k^n)$ in the nonlinear subspace, the Dirac approximation is extended to a sliced Gaussian mixture representation over the entire state space. This is achieved by the evaluation of the Gaussian mixture density $\widetilde{f}(\underline{x}_k^l, x_k^n)$ at every Dirac position determined by the aforementioned algorithm, leading to a Gaussian mixture density in the linear subspace

$$\begin{aligned} f(\underline{x}_k^l | \xi_k^i) &= c_k \cdot \widetilde{f}(\underline{x}_k^l, \xi_k^i) \\ &= \sum_{j=1}^{L} \underbrace{c_k \cdot \widetilde{\alpha}_k^j \mathcal{N}\left(\xi_k^i - \widetilde{\xi}_k^j, \widetilde{\mathbf{C}}_k^{nj}\right)}_{=:\beta_k^{ij}} \cdot \mathcal{N}\left(\underline{x}_k^l - \underline{\widetilde{\mu}}_k^j, \widetilde{\mathbf{C}}_k^{lj}\right) \end{aligned}$$

for every slice $i = 1 \ldots M$ with L components. The corresponding parameters for the Gaussian components are assigned as

$$\beta_k^{ij} = \frac{\widetilde{\alpha}_k^j \mathcal{N}\left(\xi_k^i - \widetilde{\xi}_k^j, \widetilde{\mathbf{C}}_k^{nj}\right)}{\sum_{j=1}^{L} \beta_k^{ij}} \ , \ \underline{\mu}_k^{ij} = \underline{\widetilde{\mu}}_k^j \ , \text{ and } \mathbf{C}_k^{ij} = \widetilde{\mathbf{C}}_k^{lj} \ .$$

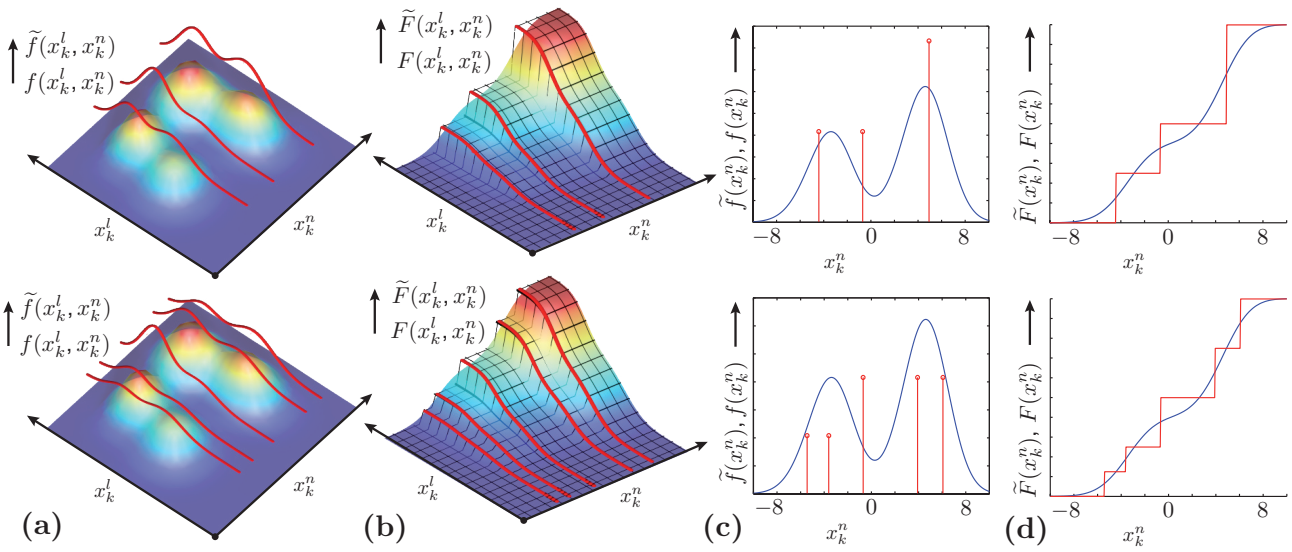

Figure 4. Approximation of a Gaussian mixture density $\widetilde{f}(x_k^l, x_k^n)$ with 4 components for a different number of density slices $M = 3$ and $M = 5$. **(a)–(b)** True joint density $\widetilde{f}(x_k^l, x_k^n)$ and joint distribution $\widetilde{F}(x_k^l, x_k^n)$ to be approximated (*surface*) and their respective approximation $f(x_k^l, x_k^n)$ and $F(x_k^l, x_k^n)$ (*red lines*). **(c)–(d)** True marginal density $\widetilde{f}(x_k^n)$ and true marginal distribution $\widetilde{F}(x_k^n)$ to be approximated (*blue line*), and their respective approximations $f(x_k^n)$ and $F(x_k^n)$ (*red line*).

Example 1 (Density approximation)
In this example, a Gaussian mixture with 4 components is approximated by a sliced Gaussian mixture. The mean values of the state vector $\underline{x}_k = \begin{bmatrix} x_k^l & x_k^n \end{bmatrix}^T$ of the individual components are given by

$$\hat{\underline{x}}_k^{(1)} = \begin{bmatrix} 5 \\ -3 \end{bmatrix}, \quad \hat{\underline{x}}_k^{(2)} = \begin{bmatrix} 0 \\ -4 \end{bmatrix}, \quad \hat{\underline{x}}_k^{(3)} = \begin{bmatrix} 5 \\ 4 \end{bmatrix}, \quad \hat{\underline{x}}_k^{(4)} = \begin{bmatrix} -1 \\ 5 \end{bmatrix} \ ,$$

the covariance matrices are assumed to be

$$\mathbf{C}_k^{(1)} = \begin{bmatrix} 2 & 0 \\ 0 & 3 \end{bmatrix}, \mathbf{C}_k^{(2)} = \begin{bmatrix} 4 & 0 \\ 0 & 4 \end{bmatrix}, \mathbf{C}_k^{(3)} = \begin{bmatrix} 3 & 0 \\ 0 & 4 \end{bmatrix}, \mathbf{C}_k^{(4)} = \begin{bmatrix} 5 & 0 \\ 0 & 3 \end{bmatrix} \ ,$$

and the weights are assumed to be equal to 0.25. The approximation of the Gaussian mixture density for different numbers of density slices is visualized in Fig. 4. The upper and lower parts show the approximation with $M = 3$ and $M = 5$ slices.

IV. Sliced Gaussian Mixture Filter

This section is devoted to the Bayesian prediction step and the combined filter and prediction step based on *sliced Gaussian mixture densities* (5) introduced in Sec. III-A. It is shown, how this novel density representation can be exploited for decomposing the general prediction step (3) and filter step (4) into a *linear* and *nonlinear part*. By this means, a more efficient closed-form calculation of the estimation problem for certain nonlinear dynamic systems is possible.

A. Considered Model Structure

The key idea of decomposing the estimation problem by sliced Gaussian mixture densities and a systematic reapproximation can be applied to various dynamic systems. However, for the sake of simplicity and in order to show the novelties of the new approach, we consider a specific system structure similar to the model structure introduced in [7].

Example 2 (Considered model structure)
In this paper, we restrict our attention to a nonlinear discrete-time dynamic system with a linear substructure

$$\begin{aligned} \underline{\boldsymbol{x}}_{k+1}^l &= \mathbf{A}_k\left(\underline{\boldsymbol{x}}_k^n\right) \underline{\boldsymbol{x}}_k^l + \mathbf{B}_k\left(\underline{\boldsymbol{x}}_k^n\right) \hat{\underline{u}}_k + \underline{\boldsymbol{w}}_k^l \ , \\ \underline{\boldsymbol{x}}_{k+1}^n &= \underline{a}_k\left(\underline{\boldsymbol{x}}_k^n\right) + \underline{\boldsymbol{w}}_k^n \ , \end{aligned} \tag{8}$$

where $\underline{\boldsymbol{x}}_k^l \in \mathbb{R}^r$ and $\underline{\boldsymbol{x}}_k^n \in \mathbb{R}^s$ denote the linear and nonlinear substate vectors decomposed by the *sliced Gaussian mixture density* (5). The noise terms $\underline{\boldsymbol{w}}_k^l$ and $\underline{\boldsymbol{w}}_k^n$ are assumed to be white Gaussian with covariance matrix

$$\mathrm{Cov}\left\{ \begin{bmatrix} \underline{\boldsymbol{w}}_k^l \\ \underline{\boldsymbol{w}}_k^n \end{bmatrix} \right\} = \begin{bmatrix} \mathbf{C}_w^l & \mathbf{0} \\ \mathbf{0} & \mathbf{C}_w^n \end{bmatrix} = \mathbf{C}_w \ , \tag{9}$$

which means that the process noise in the linear and the nonlinear subspace is stochastically independent. Furthermore, the measurement model is given by

$$\underline{\boldsymbol{y}}_k = \mathbf{H}_k\left(\underline{\boldsymbol{x}}_k^n\right) \underline{\boldsymbol{x}}_k^l + \underline{h}_k\left(\underline{\boldsymbol{x}}_k^n\right) + \underline{\boldsymbol{v}}_k \ , \tag{10}$$

where the noise term $\underline{\boldsymbol{v}}_k$ is assumed to be white Gaussian noise with covariance matrix $\mathbf{C}_v$. This model structure is quite often referred to as a *conditionally linear dynamic model* [11]. That means, given the trajectory of the nonlinear state vector $\underline{x}_k^n$, the system can be regarded as linear, visualized in Fig. 2 **(a)**.

For the sake of simplicity and in order to keep the equations simple, the following abbreviations are used:

$$\mathbf{A}_k^i := \mathbf{A}_k\left(\underline{\xi}_k^i\right), \quad \mathbf{B}_k^i := \mathbf{B}_k\left(\underline{\xi}_k^i\right), \quad \mathbf{H}_k^i := \mathbf{H}_k\left(\underline{\xi}_k^i\right) \ ,$$

which can be seen as the system matrix $\mathbf{A}_k$, input matrix $\mathbf{B}_k$, and measurement matrix $\mathbf{H}_k$ conditioned on the *nonlinear* dimension $\underline{\xi}_k^i$, i.e., on the position of the individual density slices.

B. Prediction Step

Based on the system structure given in Example 2, we now explain the Bayesian prediction step and derive an efficient time update for the prior density $f^e(\underline{x}_k^l, \underline{x}_k^n)$ given as *sliced Gaussian mixture density* representation (5).

The predicted density $\widetilde{f}^p(\underline{x}_{k+1}^l, \underline{x}_{k+1}^n)$ for the next discrete time step can be determined by substituting the prior density into the Chapman-Kolmogorov equation (3), leading to

$$\widetilde{f}^p(\underline{x}_{k+1}^l, \underline{x}_{k+1}^n) = \int_{\mathbb{R}^s} \int_{\mathbb{R}^r} f^T\left(\underline{x}_{k+1}^l, \underline{x}_{k+1}^n | \underline{x}_k^l, \underline{x}_k^n\right) \cdot \sum_{i=1}^{M} \alpha_k^i \, \delta(\underline{x}_k^n - \underline{\xi}_k^i) \, f^e(\underline{x}_k^l | \underline{\xi}_k^{ei}) \mathrm{d}\underline{x}_k^l \mathrm{d}\underline{x}_k^n \ ,$$

which in general cannot be solved analytically for nonlinear systems and arbitrary density representations for $f^e(\underline{x}_k^l | \underline{\xi}_k^{ei})$. In view of the system model (8), the transition density f^T is given by

$$f^T(\,\cdot\,) = \mathcal{N}\left(\begin{bmatrix} \underline{x}_{k+1}^l - \mathbf{A}_k(\underline{x}_k^n)\underline{x}_k^l - \mathbf{B}_k(\underline{x}_k^n)\hat{\underline{u}}_k \\ \underline{x}_{k+1}^n - \underline{a}_k(\underline{x}_k^n) \end{bmatrix}, \mathbf{C}_w\right) \ ,$$

with uncorrelated process noise between the linear and nonlinear subspace, i.e., the covariance matrix $\mathbf{C}_w$ is structured according to (9). In this case, the transition density f^T can be written as a product of two density functions. Applying the sifting property of Dirac's delta function leads to

$$\widetilde{f}^p(\underline{x}_{k+1}^l, \underline{x}_{k+1}^n) = \sum_{i=1}^{M} \alpha_k^i \mathcal{N}\left(\underline{x}_{k+1}^n - \underline{a}_k(\underline{\xi}_k^{ei}), \mathbf{C}_w^n\right) \cdot \int_{\mathbb{R}^r} \mathcal{N}\left(\underline{x}_{k+1}^l - \mathbf{A}_k^i \underline{x}_k^l - \mathbf{B}_k^i \underline{u}_k, \mathbf{C}_w^l\right) f^e(\underline{x}_k^l | \underline{\xi}_k^{ei}) \mathrm{d}\underline{x}_k^l \ , \quad (11)$$

where the estimated density $f^e(\underline{x}_k^l | \underline{\xi}_k^{ei})$ is represented by a Gaussian mixture density (7). Thanks to the *conditionally linear dynamic model* (8), the integral in (11) can be solved analytically using a linear prediction step for a Gaussian mixture, i.e., a bank of independent Kalman Filters.

Finally, the predicted density $\widetilde{f}^p$ results in a Gaussian mixture in both linear $\underline{x}_k^l$ and nonlinear subspace $\underline{x}_k^n$,

$$\widetilde{f}^p(\underline{x}_{k+1}^l, \underline{x}_{k+1}^n) = \sum_{i=1}^{M} \sum_{j=1}^{N^i} \alpha_k^i \beta_k^{ij} \cdot \mathcal{N}\left(\underline{x}_{k+1}^n - \underline{\xi}_{k+1}^{pi}, \mathbf{C}_w^n\right) \mathcal{N}\left(\underline{x}_{k+1}^l - \underline{\mu}_{k+1}^{pij}, \mathbf{C}_{k+1}^{pij}\right) \ , \quad (12)$$

where means and covariance matrices in the linear subspace $\underline{\boldsymbol{x}}_k^l$ are calculated by applying the standard Kalman prediction step. The means in the nonlinear subspace $\underline{\boldsymbol{x}}_k^n$ are derived by simply repositioning the density slices according to the nonlinear system equation (8), see Table I.

Here, it is worthwhile mentioning that, although $\underline{\xi}_k^{pi}$ represents the individual position of the Dirac mixture density f^e, the $\underline{\xi}_{k+1}^{pi}$ denotes the means of the resulting Gaussian mixture density in nonlinear dimension. In order to keep the density representation (5), the predicted Gaussian mixture density $\widetilde{f}^p$ needs to be reapproximated in a systematic fashion as already shown in Sec. III-B. Furthermore, for bounding the complexity, the number of components of the resulting Gaussian mixture density (12) or the individual components of the slices (7) can be reduced by component reduction algorithms for Gaussian mixtures, e.g., [12], [13], and [14].

Table I
PREDICTION STEP: PARAMETERS OF THE PREDICTED DENSITY.

Nonlinear dimension	Conditionally linear dimension
$\underline{\xi}_{k+1}^{pi} := \underline{a}_k\left(\underline{\xi}_k^{ei}\right)$	$\underline{\mu}_{k+1}^{pij} := \mathbf{A}_k^i \underline{\mu}_k^{eij} + \mathbf{B}_k^i \hat{\underline{u}}_k$
$\mathbf{C}_w^n$	$\mathbf{C}_{k+1}^{pij} := \mathbf{A}_k^i \mathbf{C}_k^{eij} {\mathbf{A}_k^i}^T + \mathbf{C}_w^l$

Table II
FILTER STEP: PARAMETERS OF THE ESTIMATED DENSITY.

Conditionally linear dimension
$\gamma_k^{ij} := \mathcal{N}\left(\hat{\underline{y}}_k - \underline{h}_k\left(\underline{\xi}_k^{pi}\right) - \mathbf{H}_k^i \underline{\mu}_k^{pij}, \mathbf{H}_k^i \mathbf{C}_k^{pij} {\mathbf{H}_k^i}^T + \mathbf{C}_v\right)$
$\underline{\mu}_k^{eij} := \underline{\mu}_k^{pij} + \mathbf{K}\left(\hat{\underline{y}}_k - \underline{h}_k\left(\underline{\xi}_k^{pi}\right) - \mathbf{H}_k^i \underline{\mu}_k^{pij}\right)$
$\mathbf{C}_k^{eij} := \mathbf{C}_k^{pij} - \mathbf{K}\mathbf{H}_k^i \mathbf{C}_k^{pij}$
with $\mathbf{K} := \mathbf{C}_k^{pij} {\mathbf{H}_k^i}^T \left(\mathbf{C}_v + \mathbf{H}_k^i \mathbf{C}_k^{pij} {\mathbf{H}_k^i}^T\right)^{-1}$

C. Combined Filter and Prediction Step

In the combined filter and prediction step, first, the filter step is performed on a sliced Gaussian mixture, followed by the prediction step as introduced in the previous subsection. It is assumed that the the prior density f^p at time step k is given as a sliced Gaussian mixture density (5).

The predicted density at time step $k+1$ can be derived by the substitution of Bayes' formula (4) into the Chapman-Komogorov equation (3) according to

$$f^p(\underline{x}_{k+1}) = \int_{\Omega} f^T(\underline{x}_{k+1} | \underline{x}_k) \underbrace{c_k \cdot f^L(\hat{\underline{y}}_k | \underline{x}_k) f^p(\underline{x}_k)}_{=: f^e(\underline{x}_k)} \mathrm{d}\underline{x}_k$$

with a normalizing factor c_k.

The estimated density f^e results from the normalized product of the prior density f^p and the likelihood f^L according to (4). By representing the prior density f^p as a sliced Gaussian mixture, the likelihood is evaluated only at nonlinear slice positions. This results in a Kalman filter step in the linear subspace, modifying only the weights α_k^i and β_k^{ij}, the mean vectors $\underline{\mu}_k^{eij}$, and covariance matrices $\mathbf{C}_k^{eij}$ of the prior density according to the likelihood. The estimated density f^e is given as a sliced Gaussian mixture

$$f^e(\underline{x}_k^l, \underline{x}_k^n) = c_k \cdot \mathcal{N}\left(\hat{\underline{y}}_k - \mathbf{H}_k^i \underline{x}_k^l - \underline{h}_k\left(\underline{x}_k^n\right), \mathbf{C}_v\right) \cdot \sum_{i=1}^{M} \alpha_k^i \delta\left(\underline{x}_k^n - \underline{\xi}_k^{pi}\right) \cdot \sum_{j=1}^{N^i} \beta_k^{ij} \mathcal{N}\left(\underline{x}_k^l - \underline{\mu}_k^{pij}, \mathbf{C}_k^{pij}\right)$$
$$= c_k \cdot \sum_{i=1}^{M} \alpha_k^i \delta\left(\underline{x}_k^n - \underline{\xi}_k^{pi}\right) \sum_{j=1}^{N^i} \beta_k^{ij} \gamma_k^{ij} \mathcal{N}\left(\underline{x}_k^l - \underline{\mu}_k^{eij}, \mathbf{C}_k^{eij}\right) \ ,$$

with weights, means, and covariances according to Table II. It is worth mentioning that the positions of the slices $\underline{\xi}_k^{pi}$ are not affected by the filter step.

The estimated density $f^e(\underline{x}_k)$ has to be substituted into (11) in order to perform a prediction step. Applying the prediction step in Section IV-B leads to the predicted density

$$f^p(\underline{x}_{k+1}^l, \underline{x}_{k+1}^n) = c_k \cdot \sum_{i=1}^{M} \sum_{j=1}^{N^i} \alpha_k^i \beta_k^{ij} \gamma_k^{ij} \cdot \mathcal{N}\left(\underline{x}_{k+1}^n - \underline{\xi}_{k+1}^{pi}, \mathbf{C}_w^n\right) \mathcal{N}\left(\underline{x}_{k+1}^l - \underline{\mu}_{k+1}^{pij}, \mathbf{C}_{k+1}^{pij}\right),$$

with the normalizing factor $c_k = 1/\left(\sum_{i=1}^{M} \sum_{j=1}^{N^i} \alpha_k^i \beta_k^{ij} \gamma_k^{ij}\right)$. The parameters of the resulting function $f^p(\underline{x}_{k+1})$ can easily be computed by first applying the assignments in Table II and then Table I.

V. Simulation Results

In this section, the performance of the Sliced Gaussian Mixture Filter is demonstrated by means of simulation results for the following example system.

Example 3 (Nonlinear system)
In this example, we consider a nonlinear two-dimensional system with a linear substructure, which is exploited by the Sliced Gaussian Mixture Filter in order to perform the processing more efficiently. The system equation is given by

$$\boldsymbol{x}_{k+1}^l = \underbrace{(0.7 - 0.2\boldsymbol{x}_k^n)}_{\mathbf{A}_k^i} \boldsymbol{x}_k^l + \underbrace{(0.3 + 0.2\boldsymbol{x}_k^n)}_{\mathbf{B}_k^i} \hat{u}_k + \boldsymbol{w}_k^l \ ,$$
$$\boldsymbol{x}_{k+1}^n = \boldsymbol{x}_k^n + \boldsymbol{w}_k^n \ ,$$

where $\boldsymbol{w}_k^l$ and $\boldsymbol{w}_k^n$ are zero mean additive Gaussian noise with a variances of $C_w^l = 1$ and $C_w^n = 0.5$, respectively. The system input $\hat{u}_k$ is assumed to be $\hat{u}_k = -5 \sin(0.2\,k)$. The measurement equation is given by a polynomial of degree 5

$$\hat{y}_k = \underbrace{\boldsymbol{x}_k^n}_{\mathbf{H}_k^i} \boldsymbol{x}_k^l + h_k^i + \boldsymbol{v}_k \ ,$$
$$h_k^i = -0.32\,(\boldsymbol{x}_k^n)^5 - 1.6\,(\boldsymbol{x}_k^n)^4 - 5.6\,(\boldsymbol{x}_k^n)^2 - 16\boldsymbol{x}_k^n - 9.12 \ ,$$

where $\boldsymbol{v}_k$ denotes the measurement noise with a variance of $C_v = 20$. The system was simulated for 20 consecutive time steps and 68 independent Monte-Carlo simulation runs. The estimation results are shown in Fig. 5 and Fig. 6.

The density function after the combined filter and prediction step is calculated by brute force numerical integration for having a ground truth for comparison purposes, see Fig. 5 **(a)**. In Fig. 5 **(b)**, the simulation result of the marginalized paricle filter with 1000 particles is shown as a histogram of particles in the nonlinear subspace. The corresponding Gaussian components are added up in the linear subspace. Due to the randomly chosen position of the particles, the marginalized particle filter produces a joint density deviating from the true density. The resulting joint density after combined filter and prediction step for the Sliced Gaussian Mixture Filter using 20 slices is depicted in Fig. 5 **(c)**. It can easily be seen that the density derived by the Sliced Gaussian Mixture Filter hardly deviates

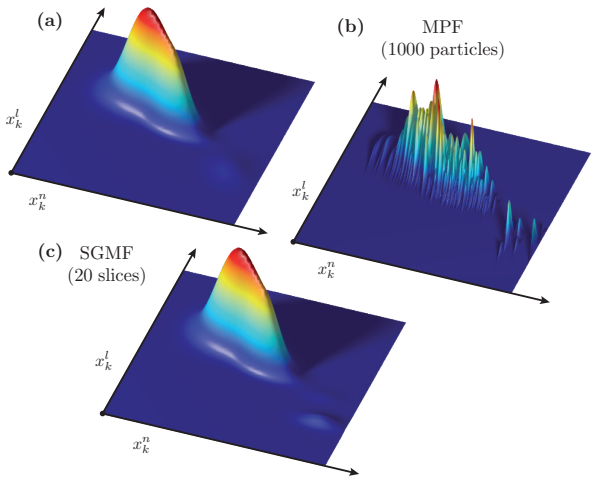

Figure 5. Visualization of joint densities for the system described in Example 3 after combined filter and prediction step at time $k = 2$: **(a)** brute force numerical integration, **(b)** marginalized particle filter ($M = 1000$ particles), and **(c)** Sliced Gaussian Mixture Filter ($M = 20$ slices).

from the brute force solution and appears much smoother compared to the density from the marginalized particle filter.

The mean squared integral deviation $\hat{D}$ over K time steps and over n Monte Carlo simulation runs is approximated by calculating the average according to

$$\hat{D} \approx \frac{1}{n\,K} \sum_{i=1}^{n} \sum_{k=1}^{K} \frac{1}{2} \int_{\mathbb{R}^2} \left(\widetilde{F}(\underline{x}_k) - F(\underline{x}_k)\right)^2 \mathrm{d}\underline{x}_k \ , \quad (13)$$

where $\widetilde{F}(\underline{x}_k)$ is the true distribution to be approximated and the sliced Gaussian mixture distribution is denoted as $F(\underline{x}_k)$. The approximation error $\hat{D}$ is shown in Fig. 6 **(a)**. During the simulation, a maximum number of 10 Gaussian components per slice was allowed, whereas 3.86 components were utilized on average. It is obvious that the deviation decreases with growing number of slices. The plateau at the right-hand side of the graph arises from discretization errors in the calculation of the true density function.

Fig. 6 **(b)** depicts a comparison between the Sliced Gaussian Mixture Filter and the marginalized particle filter with 50 times more particles than slices. This figure visualizes the mean squared integral deviation $\hat{D}$, i.e., similar to (13), between the distribution $F_B(x_k^l, x_k^n)$ derived by a brute force numerical integration method and the estimation result $F(x_k^l, x_k^n)$ from both the Sliced Gaussian Mixture Filter (red) and the marginalized particle filter (green). Furthermore, the dashed lines visualize the squared integral deviation variance C_D given by

$$C_D \approx \frac{1}{n-1} \sum_{i=1}^{n} \left(D^{(i)} - \hat{D}\right)^2 \ ,$$

where $\hat{D}$ is the mean squared integral deviation (13) and $D^{(i)}$ denotes the squared integral deviation of the individual Monte-Carlo simulation runs. The dashed lines in Fig. 6 **(b)** depict the squared integral deviation variance according to $\hat{D} \pm 3 \cdot C_D$.

It can be stated that the Sliced Gaussian Mixture Filter significantly outperforms the marginalized particle filter for this simulated example. Compared to the Sliced Gaussian Mixture Filter with $M = 15$ slices, the marginalized particle filter needs far more particles ($M \approx 2500$), in order to reach the same deviation error. Thanks to the low number of slices compared to the higher number of particles, the Sliced Gaussian Mixture Filter requires far less memory storage.

VI. Conclusion and Future Work

In this paper, based on a novel density approximation – *sliced Gaussian mixture density* – the linear substructure in certain dynamic systems is exploited in order to derive a more efficient estimation process. The systematic approximation approach reduces a distance measure between an approximation given as a sliced Gaussian mixture density and the true density. It is important to note that, although we have chosen the special model structure (8) for the sake of simplicity, the principles of the Sliced Gaussian Mixture Filter can be applied to more general conditionally linear models in a straightforward fashion.

Future work is devoted to the consideration of the joint density $f(\underline{\boldsymbol{x}}_k^l, \underline{\boldsymbol{x}}_k^n)$, i.e., both in linear and nonlinear dimension, for the positioning of the sliced Gaussian mixtures. By this means, the accuracy of the approximation could be further improved and less slices would be necessary to get an accurate density representation. It is believed that the approach introduced in this paper allows the systematic approximation of the joint densities. For further reducing the number of slices and increasing the performance of the filter, it is possible to consider both the system equation (transition density) and measurement equation (likelihood) for the determination of the positions of the slices.

The next step is to extend the Sliced Gaussian Mixture Filter to system models decomposable into several nonlinear subsystems. By applying separate nonlinear filters to the individual subspaces, it might be possible to derive an overall more efficient estimation processs.

VII. Acknowledgments

This work was partially supported by the German Research Foundation (DFG) within the Research Training Group GRK 1194 “Self-organizing Sensor-Actuator-Networks”.

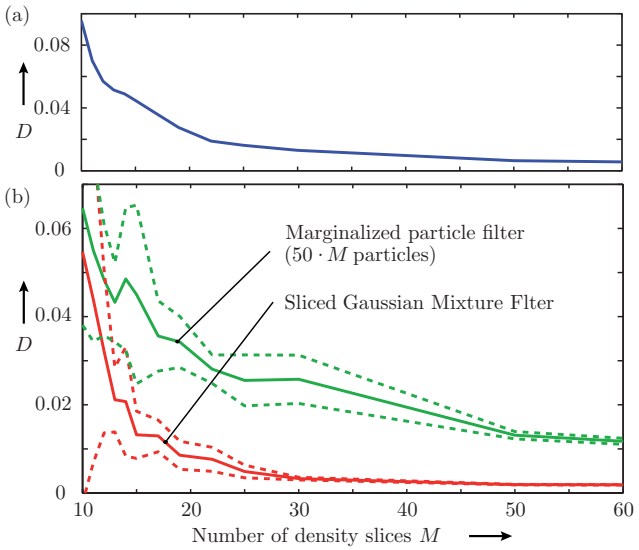

Figure 6. Simulation results for Example 3: **(a)** Visualization of the approximation error of the Sliced Gaussian Mixture Filter for various number of slices, i.e., $M = 10 \ldots 60$. **(b)** Mean and variance of squared integral deviation D between the distribution $F_B(\underline{x}_k)$ derived by brute force computation and the estimation result. Comparison of the Sliced Gaussian Mixture Filter (red) and the marginalized particle filter (green) with 50 times more particles than slices.

References

[1] F. Gustafsson, F. Gunnarsson, N. Bergman, U. Forssell, J. Jansson, R. Karlsson, and P.-J. Nordlund, “Particle Filters for Positioning, Navigation, and Tracking,” *IEEE Transactions on Signal Processing*, vol. 50, no. 2, pp. 425–437, 2002.

[2] J. Vermaak, C. Andrieu, A. Doucet, and S. J. Godsill, “Particle Methods for Bayesian Modeling and Enhancement of Speech Signals,” *IEEE Transactions on Speech and Audio Processing*, vol. 10, no. 3, pp. 173–185, March 2002.

[3] F. Sawo, M. F. Huber, and U. D. Hanebeck, “Parameter Identification and Reconstruction Based on Hybrid Density Filter for Distributed Phenomena,” in *10th International Conference on Information Fusion (Fusion 2007)*, Quebec, Canada, Jul. 2007. [Online]. Available: http://isas.uka.de/Publikationen/Fusion07_SawoHuber-ParamIdent.pdf

[4] F. Sawo, K. Roberts, and U. D. Hanebeck, “Bayesian Estimation of Distributed Phenomena using Discretized Representations of Partial Differential Equations,” in *3rd International Conference on Informatics in Control, Automation and Robotics (ICINCO’06)*, Setúbal, Portugal, Aug. 2006, pp. 16–23. [Online]. Available: http://isas.uka.de/Publikationen/ICINCO06_SawoRoberts.pdf

[5] G. Casella and C. P. Robert, “Rao-Blackwellisation of Sampling Schemes,” *Biometrika*, vol. 83, no. 1, pp. 81–94, 1996.

[6] C. Andrieu and A. Doucet, “Particle Filtering for Partially Observed Gaussian State Space Models,” *Journal of the Royal Statistical Society*, vol. 64, no. 4, pp. 827 – 836, 2002.

[7] T. Schön, F. Gustafsson, and P.-J. Nordlund, “Marginalized Particle Filters for Mixed Linear/Nonlinear State-Space Models,” *IEEE Transactions on Signal Processing*, vol. 53, no. 7, pp. 2279–2287, July 2005.

[8] J. Vermaak, A. Doucet, and P. Perez, “Maintaining Multimodality Through Mixture Tracking,” *Ninth IEEE International Conference on Computer Vision, 2003. Proceedings.*, vol. 2, pp. 1110–1116, 2003.

[9] O. C. Schrempf and U. D. Hanebeck, “A State Estimator for Nonlinear Stochastic Systems Based on Dirac Mixture Approximations,” in *Proceedings of the 4th International Conference on Informatics in Control, Automation and Robotics (ICINCO 2007)*, vol. SPSMC, Angers, France, May 2007, pp. 54–61. [Online]. Available: http://isas.uka.de/Publikationen/ICINCO07_Schrempf-DiracMixt.pdf

[10] U. D. Hanebeck and O. C. Schrempf, “Greedy Algorithms for Dirac Mixture Approximation of Arbitrary Probability Density Functions,” in *IEEE Conference on Decision and Control (CDC 2007)*, Dec. 2007. [Online]. Available: http://isas.uka.de/Publikationen/CDC07_HanebeckSchrempf.pdf

[11] R. Chen and J. S. Liu, “Mixture Kalman Filters,” *Journal of the Royal Statistical Society*, vol. 62, no. 3, pp. 493–508, 2000.

[12] A. R. Runnalls, “Kullback-Leibler Approach to Gaussian Mixture Reduction,” *IEEE Transactions on Aerospace and Electronic Systems*, vol. 43, no. 3, pp. 989–999, July 2007.

[13] P. S. Maybeck and B. D. Smith, “Multiple Model Tracker based on Gaussian Mixture Reduction for Maneuvering Targets in Clutter,” *8th International Conference on Information Fusion*, vol. 1, pp. 40–47, July 2005.

[14] D. J. Salmond, “Mixture Reduction Algorithms for Target Tracking in Clutter,” in *Proc. SPIE Vol. 1305, p. 434-445, Signal and Data Processing of Small Targets 1990*, ser. Presented at the Society of Photo-Optical Instrumentation Engineers (SPIE) Conference, O. E. Drummond, Ed., vol. 1305, Oct. 1990, pp. 434–445.