LRKF Revisited: The Smart Sampling Kalman Filter (S^2KF)

JANNIK STEINBRING
UWE D. HANEBECK

An accurate Linear Regression Kalman Filter (LRKF) for nonlinear systems called Smart Sampling Kalman Filter (S^2KF) is introduced. In order to get a better understanding of this new filter, a general introduction to Nonlinear Kalman Filters based on statistical linearization and LRKFs is given. The S^2KF is based on a new low-discrepancy Dirac mixture approximation of Gaussian densities. This approximation comprises an arbitrary number of optimally and deterministically placed samples in the relevant regions of the state space, so that the filter resolution can be adapted to either achieve high-quality results or to meet computational constraints. The S^2KF contains the UKF with equally weighted samples as a special case when using the same amount of samples. With an increasing number of samples, the new filter converges to the (typically unfeasible) exact analytic statistical linearization. Hence, the S^2KF can be seen as the ultimate generalization of all LRKFs such as the UKF, sigma-point filters, higher-order variants etc., as it homogeneously covers the state space with a freely chosen number of samples. It is evaluated against state-of-the-art LRKFs by performing nonlinear prediction and extended target tracking.

Manuscript received January 17, 2014; revised July 2, 2014; released for publication October 13, 2014.

Authors' address: Intelligent Sensor-Actuator-Systems Laboratory (ISAS), Institute for Anthropomatics and Robotics, Karlsruhe Institute of Technology (KIT), Germany (e-mail: jannik.steinbring@kit.edu, uwe.hanebeck@ieee.org).

1557-6418/14/$17.00 © 2014 JAIF

I. INTRODUCTION

We consider estimating the hidden state of a discrete-time stochastic nonlinear dynamic system based on noisy measurements through Bayesian inference. This is an important problem in many fields of current research such as (extended) object and group tracking [1]–[6], human motion tracking [7], object shape estimation [5], [8], [9], robotics [10], or estimation of extrinsic camera parameters [11].

Bayesian inference is a versatile approach for performing state estimation, but in general one has to cope with complex, e.g., multi-modal or non-Gaussian, state and noise probability density functions, which prohibits almost always the derivation of closed-form solutions. Particle Filters [12], [13] are elaborate Bayesian estimation techniques that try to deal with and maintain particle approximations of such complex densities. However, their main drawbacks are the high computational effort due to large sample sets, the problem of sample degeneration, non-reproducible results, and the need for explicit likelihood functions.

Therefore, simplifications are required in order to derive more efficient but still powerful estimators. A common first step of simplification is to get rid of maintaining the true complex state density by approximating it as a single Gaussian. Estimators using this approximation are grouped into the class of Gaussian Filters. But even with this convenient state density, closed-form solutions for state prediction, and especially for incorporating newly received measurements into the state estimate, are rarely available. Hence, specific Gaussian Filters are needed that try to overcome this problem by delivering approximative solutions for the state prediction and measurement updates. Such filters are for example the Gaussian Particle Filter (GPF) [14] or the Progressive Gaussian Filter (PGF) [15], [16] that make use of sample representations of the occurring Gaussian densities. Nevertheless, these filters, and in particular their measurement updates, are still costly.

For that reason, a further common step is to simplify the measurement update by computing linear approximations of the nonlinear mapping between the hidden system state and the noisy measurements, that is, obtaining a linear relationship between them. Such linearizations can be computed in several ways, e.g., using Taylor series, as will be discussed below. A key result of this obtained linear relationship is the possibility to perform backward inference without an explicit likelihood function. Instead, the well-known Kalman Filter formulas can be used [17]. Consequently, these estimators are referred to as *Nonlinear Kalman Filters*, as the Kalman Filter is applied to nonlinear systems [18].

In case of an already linear system corrupted by additive Gaussian noise, no linearization is required and the resulting filter is identical to the Kalman Filter [17], which yields optimal estimation results in the sense of a Minimum Mean Square Error (MMSE) [2]. However,

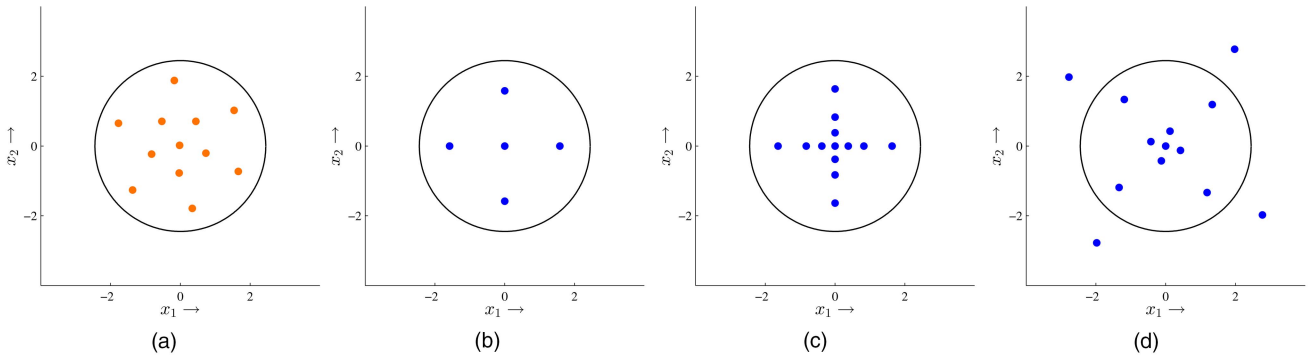

Fig. 1. Sampling of a two-dimensional standard normal distribution by the new S^{2}KF (orange points) and state-of-the-art LRKFs (blue points). Covariance matrices with confidence interval of 95% (black circles). (a) New S^{2}KF with 12 samples. (b) UKF with equal weights. (c) GF with 13 samples. (d) RUKF with 13 samples.

in case of nonlinear systems linearization is required which might be a strong simplification depending on the degree of the nonlinearity. The consequence is a diminished estimation performance compared to the more general Gaussian Filters not making use of such linearization. An option to mitigate linearization errors and therefore improve the estimation quality is to reduce the degree of nonlinearity by augmenting the actual nonlinear measurement model with additional, properly chosen mappings as proposed in [19].

One way to perform such linearization is *statistical linearization* [20], [21]. Basically, implementing a Nonlinear Kalman Filter based on statistical linearization only amounts to calculating the first- and second-order moments of (nonlinear) transformed densities, depending on the given system and measurement equations. For some equations, including polynomials, trigonometric functions, and their combinations, these moments can be calculated analytically [22]. Hence, this provides the filter based on statistical linearization with the best possible estimation quality and is referred to as analytic statistical linearization. However, this approach requires an individual treatment of each occurring equation which is time-consuming, error-prone, and prevents a generic filter applicable to any system and measurement equation, regardless of its complexity.

A widespread solution for these problems are sample-based approaches, where the occurring state and noise densities are represented as a set of samples, selected in a random or deterministic way. This allows to perform statistical linearization in the form of *statistical linear regression* [21], [23]. Nonlinear Kalman Filters making use of statistical linear regression are called Linear Regression Kalman Filters (LRKFs) [23], [24]. As a consequence of using samples instead of continuous densities, time and measurement updates have to be adapted in order to handle this density representation. On the one hand, the samples are propagated individually through the given system and measurement equations. On the other hand, occurring analytic moment calculations are turned into their sample-based counterparts, i.e., sample mean and sample covariance. Of course, this introduces a further approximation step (compared to the analytic statistical linearization) that may negatively affect the estimation performance. Nevertheless, employing an LRKF offers several advantages. First, due to the lack of an explicit use of a likelihood, the problem of sample degeneration is avoided,[1] and second, we obtain a generic filter that allows us to work with black box systems, e.g., systems given as (binary) programs, or to switch easily between different system and measurement models without any additional effort. Moreover, this facilitates filter design in the sense of rapid prototyping, as a newly designed system or measurement model can be tried out immediately.

Despite all these advantages of LRKFs, in order to improve overall estimation quality of filters based on statistical linearization, a mixture of analytic and sample-based moment calculation (semi-analytic approach) [22], [25] should be used whenever possible.

A. Contribution

In this paper, we introduce a new LRKF called the Smart Sampling Kalman Filter (S^{2}KF), which can be seen as the ultimate generalization of all LRKFs. For that purpose, we compute deterministic approximations of multivariate standard normal distributions comprising a predefined arbitrary number of optimally placed samples in the relevant regions of the state space (see Fig. 1(a)). These sets of deterministically chosen samples serve as the fundamental basis for the new filter. In contrast to approaches using non-deterministic sampling, this lets the filter compute reproducible results and is more efficient, as a much smaller amount of samples has to be employed.

By simply increasing the number of employed samples, the new filter converges to the analytic statisti-

[1]This is in contrast to filters explicitly using a likelihood, where backward inference implies a sample re-weighting that typically leads to a significantly reduced amount of samples contributing to the computation of the posterior moments, and consequently, to inaccurate results.

cal linearization as the resulting approximation of the standard normal distribution becomes more accurate. Moreover, this approach requires only a single and intuitive optimization parameter, i.e., the number of utilized samples. This makes filter fine-tuning simple, even for people not very familiar with sample-based Kalman filtering. Moreover, as opposed to state-of-the-art LRKFs, the number of samples are completely independent of the concrete dimension of the normal distribution and, hence, can be chosen freely. There are no restrictions such as a linear or exponential increase. This offers the possibility to (automatically) adapt the number of utilized samples for each time step individually depending on the concrete filtering problem, e.g., use less samples in situations of mild nonlinearities and more samples in case of high nonlinearities.

B. Related Work

One of the most popular LRKFs is the Unscented Kalman Filter (UKF) [26], [27]. It employs $2N + 1$ systematically chosen, axis-aligned samples, where N denotes the dimension of the standard normal distribution required for the time and measurement updates (see Fig. 1(b)). One of its greatest advantages is the ease with which the sample set can be created as well as the low computational effort due to the small and fixed amount of used samples. However, this property is also its main drawback. First, it is not possible to increase the number of samples for a concrete dimension N in order to improve the estimation quality. A consequence is that the even moments of a Gaussian greater than the second-order cannot precisely matched [27]. Second, the state space coverage suffers from the fact that the samples are placed solely on some principal axes.[2] Both of these factors have a negative impact on the estimation quality. And third, the small amount of employed samples makes it possible to compute non-positive definite covariance matrices and, thus, makes it hard for filtering applications to work reliably. For example, consider the nonlinear transformation of a Gaussian density using a sine-shaped function. If all samples of the Gaussian fall onto the zeros, the transformed Gaussian will be a density with zero variance [28].

Another drawback of the UKF is a rather unintuitive parameter that controls the sample spread and weighting. Besides the use of heuristics, maximum likelihood estimators can also be employed for determining these parameters. In [29], the authors select a limited set of possible values for the scaling parameter. During a filter step they perform an update for all selected scaling values individually and then choose the update that best matches the given measurement. Instead of simply trying various parameters during a filter step, the authors in [28] propose a parameter determination based on a Gaussian process optimization. Both approaches can improve the estimation quality, but also introduce new parameters (the possible scaling values and parameters controlling the optimization) that have to be determined in some way. Moreover, despite the additional computational effort due to the several computed updates for one filter step, the number of samples remains the same and, hence, the problems of insufficient state space coverage and non-positive definite covariance matrices are left unchanged.

A first step to improve the situation is done by the Gaussian Filter (GF) [30]. It enhances the sampling by deterministically placing an arbitrary number of samples on each principal axis (see Fig. 1(c)). Although the number of samples can easily be adjusted, which solves the problem of a fixed amount of samples and makes the covariance computation more reliable, the state space coverage still remains sparse due to the axes-only sample placement.

In order to overcome the problem of a sparse state space coverage, a non-deterministic sampling approach called Randomized Unscented Kalman Filter (RUKF) is introduced in [31]. Here, the moments for the time and measurement updates are calculated with the aid of an iterative stochastic integration rule, where each iteration uses an additional UKF sample set with random scaling and rotation (see Fig. 1(d)). In contrast to simple Gaussian random sampling, this guarantees that mean and covariance are always captured correctly. Furthermore, it is possible to create sample sets of arbitrary size with samples not only placed on the principal axes of the state space. Hence, the entire state space is covered in an adjustable manner. However, even though no complex parameters are required and the state space coverage is improved, this approach relies on the law of large numbers. Therefore, a large amount of samples is required to produce satisfying estimation results, particularly in larger state spaces. In addition, estimation results are not reproducible due to its non-deterministic nature. This is based on the fact that during each time and measurement update an individual set of samples is drawn randomly, which makes the filter outcome unpredictable.[3]

Table I summarizes the advantages and disadvantages of these state-of-the-art LRKFs and the new S^2KF. We emphasize that only the S^2KF is capable of producing reproducible results by using an arbitrary amount of samples placed in the entire state space (not only on the axes).

A completely different approach to compute the moments required by statistical linearization is to approximate the nonlinear system and measurement models with the aid of polynomials. That is, instead of the Gaus-

[2]Depending on the matrix square root method used for transforming the sample set to a non-standard Gaussian, i.e., the matrix factorization of the involved covariance matrices. For example, the Cholesky decomposition or the eigendecomposition.

[3]An option would be to reuse a single randomly generated sample set for all updates. However, this would conflict with the idea of random sampling, where sample sets representing the Gaussian distribution unfavorably are averaged out over time.

TABLE I
Comparison of important filter properties between state-of-the-art LRKFs and the new Smart Sampling Kalman Filter.

LRKF	**UKF**	**GF**	**RUKF**	**S^{2}KF**
Sample placement constraint	Axes	Axes	None	None
User-defined sample amount	No	Yes	Yes	Yes
Filter results	Reproducible	Reproducible	Not reproducible	Reproducible

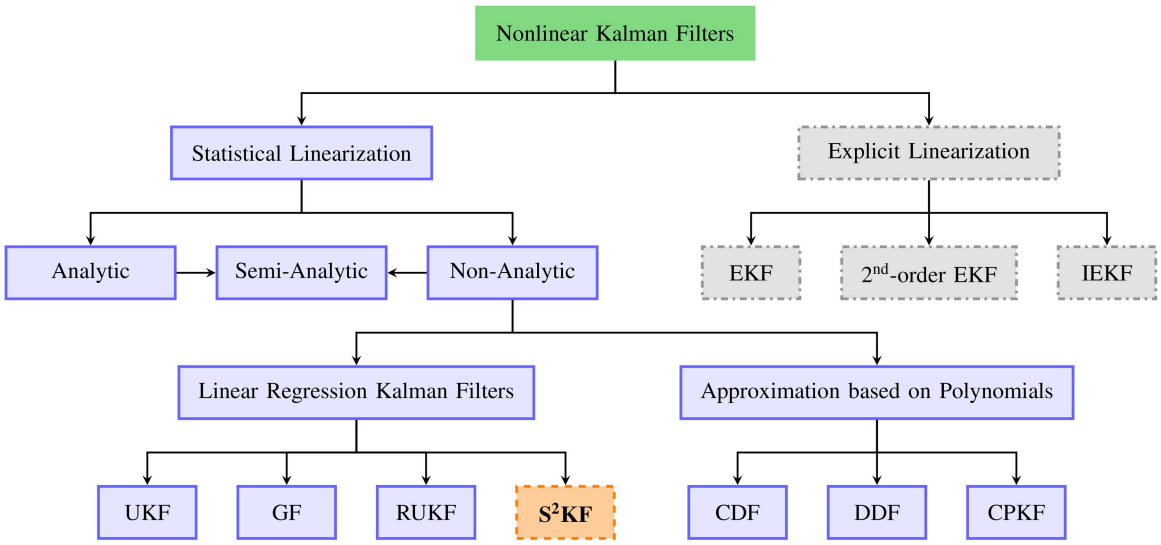

Fig. 2. Taxonomy of the discussed Nonlinear Kalman Filters relying on statistical linearization (solid blue), including the new S^{2}KF (dashed orange), and explicit linearization (dash-dotted gray).

sian distributions, the nonlinear models themselves get approximated. By doing so, formulas can be obtained that allow derivative-free and closed-form moment calculations which require only multiple evaluations of the nonlinear models. One option is to use polynomial interpolations for approximating the nonlinear models. Such filters are for example the Central Difference Filter (CDF) [32] and the Divided Difference Filter (DDF) [33]. Although these filters evaluate the nonlinear models at the same points as the UKF does, the filter results are different due to their different ways of computing the desired moments [21]. Another option is to approximate the nonlinear models by means of Chebyshev polynomials series expansion which results in the Chebyshev Polynomial Kalman Filter (CPKF) as proposed in [34]. Here, the actual polynomial approximation is obtained by using discrete cosine transformations. However, the proposed approach only works for a one-dimensional state space.

In contrast to statistical linearization, an explicit linearization of the system and measurement models based on Taylor series approximation is also possible. That is, the Kalman Filter formulas are still being used, only the type of linearization is changing. The Extended Kalman Filter (EKF) uses first-order Taylor expansions at the prior state mean for system and measurement model linearization, whereas its iterated version, the Iterated Extended Kalman Filter (IEKF), tries to improve estimation quality by finding a better point for linearization to take a given measurement into account [18]. Second-order variants of the EKF exist [18], [35], but the additional complexity has prohibited its widespread use [13]. One problem of this type of linearization is the need for explicit derivatives. In the best case, these can be taken analytically which unfortunately entails the same problems occurring in case of analytic statistical linearization: no easy exchange between different system and measurement models is possible, and it is time-consuming and error-prone. In all other cases, approximations of the derivatives will be inevitable. Another problem is that the linearization is only performed at a single point, that is, not the entire statistical information of the prior state estimate is taken into account during linearization. This typically leads to inferior estimation results compared to statistical linearization [22]. Moreover, this makes the filter also sensitive to the specific point used for the linearization, that is, to the prior state mean.

Fig. 2 shows a taxonomy of the above discussed Nonlinear Kalman Filters including the new S^{2}KF (dashed orange). It underlines the important difference between those filters relying on statistical linearization (solid blue) and the EKF variants using explicit linearization (dash-dotted gray). Additionally, all the fil-

ters in the bottom row try to achieve results as close as possible to the analytic statistical linearization.

C. Overview

The remainder of this paper is structured as follows. In Sec. II, we give a detailed formulation of the general Gaussian filtering problem using Bayesian inference. Then, in Sec. III, the class of Nonlinear Kalman Filters based on statistical linearization is presented. Sec. IV describes the transition from analytic statistical linearization to statistical linear regression in the form of the general LRKF approach. After that, in Sec. V, we describe how to compute optimal standard normal approximations using the idea of Localized Cumulative Distributions. Based on this, we introduce the new S^2KF. Extensive evaluation of the new filter is performed in Sec. VI. Finally, the conclusions are presented in Sec. VII.

II. PROBLEM FORMULATION

We consider estimating the hidden state $\underline{x}_k$ of a discrete-time stochastic nonlinear dynamic system based on noisy measurements $\tilde{\underline{y}}_k$.[4] The dynamic system is modeled by the system equation

$$\underline{x}_k = \underline{a}_k(\underline{x}_{k-1}, \underline{w}_k) \tag{1}$$

and the measurement equation

$$\underline{y}_k = \underline{h}_k(\underline{x}_k, \underline{v}_k), \tag{2}$$

where $\underline{y}_k$ denotes the measurement random variable from which the measurements $\tilde{\underline{y}}_k$ originate, and $\underline{w}_k$ as well as $\underline{v}_k$ Gaussian white noise. It is assumed that both noise processes are mutually independent and also independent of the system state. The system equation (1) models the temporal evolution of the system state, whereas the measurement equation (2) models the relation between the received noisy measurements $\tilde{\underline{y}}_k$ and the not direct observable system state $\underline{x}_k$.

We denote the probability density function (pdf) of the state at time step k conditioned on the k received measurements $\tilde{\underline{y}}_1, \ldots, \tilde{\underline{y}}_{k-1}, \tilde{\underline{y}}_k$ as

$$f_k^e(\underline{x}_k) = f(\underline{x}_k \mid \tilde{\underline{y}}_k, \tilde{\underline{y}}_{k-1}, \ldots, \tilde{\underline{y}}_1) = f(\underline{x}_k \mid \tilde{\underline{y}}_{k:1}), \tag{3}$$

and the predicted state density, i.e., the pdf of the state at time step k conditioned only on the measurements $\tilde{\underline{y}}_1, \ldots, \tilde{\underline{y}}_{k-2}, \tilde{\underline{y}}_{k-1}$, as

$$f_k^p(\underline{x}_k) = f(\underline{x}_k \mid \tilde{\underline{y}}_{k-1}, \tilde{\underline{y}}_{k-2}, \ldots, \tilde{\underline{y}}_1) = f(\underline{x}_k \mid \tilde{\underline{y}}_{k-1:1}). \tag{4}$$

The noise pdfs are given by

$$f_k^w(\underline{w}_k) = \mathcal{N}(\underline{w}_k; \hat{\underline{w}}_k, \mathbf{C}_k^w)$$

and

$$f_k^v(\underline{v}_k) = \mathcal{N}(\underline{v}_k; \hat{\underline{v}}_k, \mathbf{C}_k^v),$$

with means $\hat{\underline{w}}_k$ and $\hat{\underline{v}}_k$, and covariance matrices $\mathbf{C}_k^w$ and $\mathbf{C}_k^v$, respectively.

As computing the true conditional state pdfs (3) and (4) is intractable, our goal is to maintain Gaussian approximations of (3) and (4) recursively over time and incorporate new measurements by exploiting Bayes' rule. Such a recursive Bayesian estimator consists of two alternating steps, namely the time update and the measurement update.

A. Time Update

The objective of the time update, also called prediction step, is to propagate the last known Gaussian state estimate $f_{k-1}^e(\underline{x}_{k-1})$ (from the past) to the present by exploiting the given system model (1) in the form of its state-transition density $f_k^a(\underline{x}_k \mid \underline{x}_{k-1})$. This yields the predicted state estimate $f_k^p(\underline{x}_k)$ according to the Chapman-Kolomogorov equation [2]

$$\begin{aligned} f_k^p(\underline{x}_k) &= \int f_k^a(\underline{x}_k \mid \underline{x}_{k-1}) \cdot f_{k-1}^e(\underline{x}_{k-1}) d\underline{x}_{k-1} \\ &= \int\int \delta(\underline{x}_k - \underline{a}_k(\underline{x}_{k-1}, \underline{w}_k)) \\ &\quad \cdot f_{k-1}^e(\underline{x}_{k-1}) \cdot f_k^w(\underline{w}_k) d\underline{x}_{k-1} d\underline{w}_k, \end{aligned}$$

where $\delta(\cdot)$ denotes the Dirac delta function.

However, even though the state density $f_{k-1}^e(\underline{x}_{k-1})$ is Gaussian, this in general does not hold for the predicted state density $f_k^p(\underline{x}_k)$. Therefore, we have to perform a subsequent moment matching in order to fulfill our forced Gaussian state approximation. This is done by computing the predicted state mean

$$\begin{aligned} \hat{\underline{x}}_k^p &= \int \underline{x}_k \cdot f_k^p(\underline{x}_k) d\underline{x}_k \\ &= \int\int \underline{a}_k(\underline{x}_{k-1}, \underline{w}_k) \\ &\quad \cdot f_{k-1}^e(\underline{x}_{k-1}) \cdot f_k^w(\underline{w}_k) d\underline{x}_{k-1} d\underline{w}_k, \end{aligned} \tag{5}$$

and the predicted state covariance

$$\begin{aligned} \mathbf{C}_k^p &= \int (\underline{x}_k - \hat{\underline{x}}_k^p) \cdot (\underline{x}_k - \hat{\underline{x}}_k^p)^{\mathrm{T}} \cdot f_k^p(\underline{x}_k) d\underline{x}_k \\ &= \int\int \underline{a}_k(\underline{x}_{k-1}, \underline{w}_k) \cdot \underline{a}_k(\underline{x}_{k-1}, \underline{w}_k)^{\mathrm{T}} \\ &\quad \cdot f_{k-1}^e(\underline{x}_{k-1}) \cdot f_k^w(\underline{w}_k) d\underline{x}_{k-1} d\underline{w}_k - \hat{\underline{x}}_k^p \cdot (\hat{\underline{x}}_k^p)^{\mathrm{T}} \end{aligned} \tag{6}$$

of $f_k^p(\underline{x}_k)$, and finally approximating the predicted state density according to

$$f_k^p(\underline{x}_k) \approx \mathcal{N}(\underline{x}_k; \hat{\underline{x}}_k^p, \mathbf{C}_k^p). \tag{7}$$

This Gaussian state distribution will serve as basis for the measurement update.

B. Measurement Update

The measurement update or filter step incorporates a given measurement $\tilde{\underline{y}}_k$ at time step k into the predicted

[4] Vectors are underlined, matrices are printed in bold face, and the subscript k denotes the discrete time step.

state estimate (7) to correct it. This is done by using Bayes' rule. However, this requires that the measurement model (2) is turned into its corresponding likelihood function $f_k^h(\tilde{\underline{y}}_k \mid \underline{x}_k)$ by assuming that the current measurement $\tilde{\underline{y}}_k$ is conditionally independent of the already processed measurements $\tilde{\underline{y}}_{k-1:1}$ given the predicted state estimate. Then, the corrected state estimate can be obtained according to

$$f_k^e(\underline{x}_k) = \frac{f_k^h(\tilde{\underline{y}}_k \mid \underline{x}_k) \cdot f_k^p(\underline{x}_k)}{f_k^y(\tilde{\underline{y}}_k \mid \tilde{\underline{y}}_{k-1:1})}, \tag{8}$$

where $f_k^y(\tilde{\underline{y}}_k \mid \tilde{\underline{y}}_{k-1:1})$ is only a normalization constant. More precisely, the measurement distribution $f_k^y(\underline{y}_k \mid \tilde{\underline{y}}_{k-1:1})$ encodes how probable a distinct measurement is, given all prior received measurements $\tilde{\underline{y}}_{k-1:1}$. A concrete measurement $\tilde{\underline{y}}_k$ in turn is a realization of this distribution.

Equation (8) can be rewritten in the form of the joint density $f_k^{x,y}(\underline{x}_k, \underline{y}_k \mid \tilde{\underline{y}}_{k-1:1})$ of prior state and measurement as

$$f_k^e(\underline{x}_k) = \frac{f_k^{x,y}(\underline{x}_k, \tilde{\underline{y}}_k \mid \tilde{\underline{y}}_{k-1:1})}{f_k^y(\tilde{\underline{y}}_k \mid \tilde{\underline{y}}_{k-1:1})}. \tag{9}$$

Thus, the Bayesian measurement update can be interpreted as a given measurement $\tilde{\underline{y}}_k$ determining where to slice the joint density $f_k^{x,y}(\underline{x}_k, \underline{y}_k \mid \tilde{\underline{y}}_{k-1:1})$ in order to get the posterior state density $f_k^e(\underline{x}_k)$.

As with the predicted state density, the obtained posterior state density $f_k^e(\underline{x}_k)$ is not necessarily Gaussian. Consequently, the posterior state density also has to be reapproximated as a Gaussian by means of moment matching afterwards.

C. Bayesian Estimator

The alternating use of the introduced time and measurement updates, together with a given initial state estimate

$$f_0^e(\underline{x}_0) \approx \mathcal{N}(\underline{x}_0; \hat{\underline{x}}_0^e, \mathbf{C}_0^e)$$

with initial mean $\hat{\underline{x}}_0^e$ and initial covariance $\mathbf{C}_0^e$, yields the desired recursive state estimation in form of a Bayesian estimator. It is important to note that this estimator is a restricted variant of the general recursive Bayesian estimator, as we force the state distribution to be Gaussian all the time.

III. NONLINEAR KALMAN FILTERING BASED ON STATISTICAL LINEARIZATION

Although the estimator introduced in Sec. II is a much simpler variant of the general Bayesian estimator, its measurement update is still demanding. First, an explicit likelihood function is required, which is hard to derive in case of non-additive measurement noise. Second, even if one is at hand, it is still almost always impossible to compute the measurement update analytically.

However, as we already force the posterior state density $f_k^e(\underline{x}_k)$ to be Gaussian, the measurement update can be strongly simplified by additional approximating the joint density of prior state and measurement in (9) as a Gaussian, that is,

$$f_k^{x,y}(\underline{x}_k, \underline{y}_k \mid \tilde{\underline{y}}_{k-1:1}) \approx \mathcal{N}\left(\begin{bmatrix} \underline{x}_k \\ \underline{y}_k \end{bmatrix}; \begin{bmatrix} \hat{\underline{x}}_k^p \\ \hat{\underline{y}}_k \end{bmatrix}, \begin{bmatrix} \mathbf{C}_k^p & \mathbf{C}_k^{x,y} \\ (\mathbf{C}_k^{x,y})^{\mathrm{T}} & \mathbf{C}_k^y \end{bmatrix} \right), \tag{10}$$

where $\hat{\underline{y}}_k$ and $\mathbf{C}_k^y$ denote the measurement mean and covariance, and $\mathbf{C}_k^{x,y}$ the cross-covariance matrix of state and measurement. As a direct consequence of this simplification, the posterior state density becomes also Gaussian [2]

$$\begin{aligned} f_k^e(\underline{x}_k) &\approx \frac{\mathcal{N}\left(\begin{bmatrix} \underline{x}_k \\ \tilde{\underline{y}}_k \end{bmatrix}; \begin{bmatrix} \hat{\underline{x}}_k^p \\ \hat{\underline{y}}_k \end{bmatrix}, \begin{bmatrix} \mathbf{C}_k^p & \mathbf{C}_k^{x,y} \\ (\mathbf{C}_k^{x,y})^{\mathrm{T}} & \mathbf{C}_k^y \end{bmatrix} \right)}{f_k^y(\tilde{\underline{y}}_k \mid \tilde{\underline{y}}_{k-1:1})} \\ &= \mathcal{N}(\underline{x}_k; \hat{\underline{x}}_k^e, \mathbf{C}_k^e), \end{aligned} \tag{11}$$

with posterior mean

$$\hat{\underline{x}}_k^e = \hat{\underline{x}}_k^p + \mathbf{C}_k^{x,y} \cdot (\mathbf{C}_k^y)^{-1} \cdot (\tilde{\underline{y}}_k - \hat{\underline{y}}_k) \tag{12}$$

and covariance

$$\mathbf{C}_k^e = \mathbf{C}_k^p - \mathbf{C}_k^{x,y} \cdot (\mathbf{C}_k^y)^{-1} \cdot (\mathbf{C}_k^{x,y})^{\mathrm{T}}, \tag{13}$$

which in fact are the well-known Kalman Filter formulas [17]. Hence, an estimator that uses this measurement update is called Nonlinear Kalman Filter.

Fig. 3 illustrates the Nonlinear Kalman Filter measurement update in case of a scalar state and measurement. The exemplary Gaussian joint density of state and measurement $f^{x,y}(x, y)$ is depicted in Fig. 3(a) and is sliced by the given measurement $\tilde{y}$ in Fig. 3(b) to obtain the posterior state density $f^e(x)$. It should be noted that the variance of the state density $f^e(x)$, i.e., the uncertainty of the posterior state estimate, is smaller than the one of the prior state density $f^p(x)$ due to the existing correlation between prior state and measurement, that is, $\mathbf{C}^{x,y} \neq 0$ (a non-axis-aligned Gaussian joint density). Furthermore, the state mean also changes due to the additional difference between the expected measurement $\hat{y}$ and the given measurement $\tilde{y}$.

As a result of (10), the measurement distribution $f_k^y(\underline{y}_k \mid \tilde{\underline{y}}_{k-1:1})$ becomes a Gaussian, too (see Fig. 3(a)). By this means, the relation between prior state and measurement, i.e., the measurement model (2), gets *implicitly linearized*. This is a direct consequence of the fact that there exists always an equivalent linear transformation from the prior Gaussian state distribution to this Gaussian measurement distribution.

In order to obtain the posterior Gaussian density (11), that is, perform the measurement update, the three moments $\hat{\underline{y}}_k$, $\mathbf{C}_k^y$, and $\mathbf{C}_k^{x,y}$ are required. Based on the given measurement model $\underline{h}_k(\underline{x}_k, \underline{v}_k)$, measurement noise

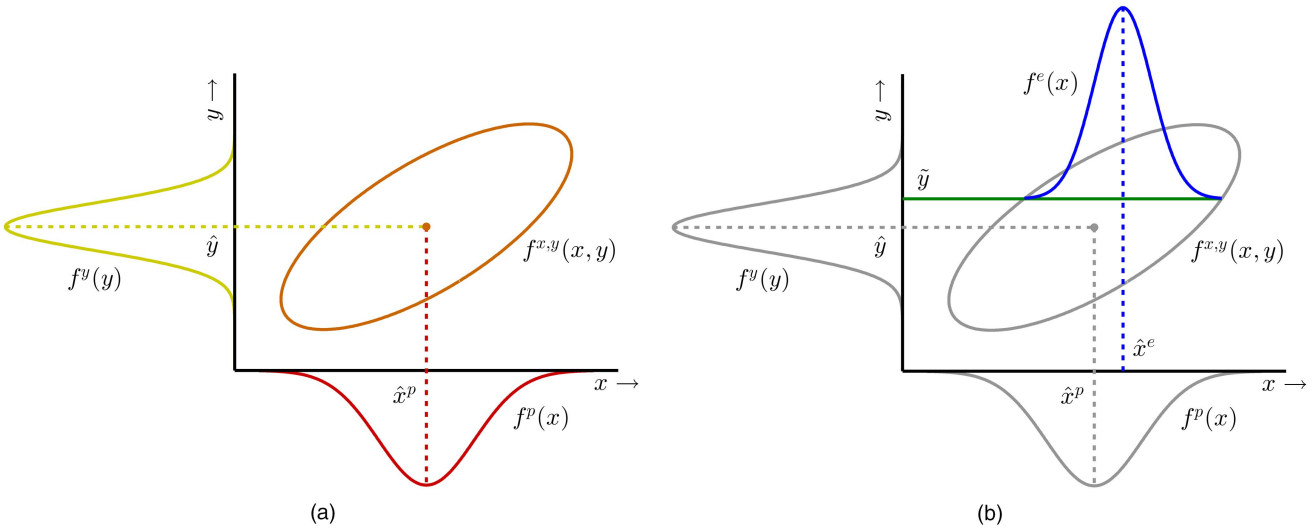

Fig. 3. Linearized measurement update in case of a scalar state x and measurement y. For readability, the time index k is omitted here. (a) Prior state density (red), measurement density (yellow), and Gaussian joint density (orange) of prior state and measurement. (b) The given measurement $\tilde{y}$ slices the joint density (green line) to obtain the posterior state density (blue).

density $f_k^v(\underline{v}_k)$, and predicted state density $f_k^p(\underline{x}_k)$, we can compute the measurement mean according to

$$\begin{aligned}\hat{\underline{y}}_k &= \int \underline{y}_k \cdot f_k^y(\underline{y}_k) d\underline{y}_k \\ &= \int\int \underline{h}_k(\underline{x}_k, \underline{v}_k) \cdot f_k^p(\underline{x}_k) \cdot f_k^v(\underline{v}_k) d\underline{x}_k d\underline{v}_k, \end{aligned} \quad (14)$$

the measurement covariance according to

$$\begin{aligned}\mathbf{C}_k^y &= \int (\underline{y}_k - \hat{\underline{y}}_k) \cdot (\underline{y}_k - \hat{\underline{y}}_k)^{\mathrm{T}} \cdot f_k^y(\underline{y}_k) d\underline{y}_k \\ &= \int\int \underline{h}_k(\underline{x}_k, \underline{v}_k) \cdot \underline{h}_k(\underline{x}_k, \underline{v}_k)^{\mathrm{T}} \\ &\quad \cdot f_k^p(\underline{x}_k) \cdot f_k^v(\underline{v}_k) d\underline{x}_k d\underline{v}_k - \hat{\underline{y}}_k \cdot \hat{\underline{y}}_k^{\mathrm{T}}, \end{aligned} \quad (15)$$

and the state measurement cross-covariance according to

$$\begin{aligned}\mathbf{C}_k^{x,y} &= \int\int (\underline{x}_k - \hat{\underline{x}}_k^p) \cdot (\underline{y}_k - \hat{\underline{y}}_k)^{\mathrm{T}} \\ &\quad \cdot f_k^{x,y}(\underline{x}_k, \underline{y}_k) d\underline{x}_k d\underline{y}_k \\ &= \int\int \underline{x}_k \cdot \underline{h}_k(\underline{x}_k, \underline{v}_k)^{\mathrm{T}} \\ &\quad \cdot f_k^p(\underline{x}_k) \cdot f_k^v(\underline{v}_k) d\underline{x}_k d\underline{v}_k - \hat{\underline{x}}_k^p \cdot \hat{\underline{y}}_k^{\mathrm{T}}, \end{aligned} \quad (16)$$

respectively. This moment calculation approach yields the so-called statistical linearization, as the implicit linearization of the measurement model takes the entire statistical information of the prior state estimate and the measurement noise into account. The result is a Nonlinear Kalman Filter based on statistical linearization. Such implicit linearization can be obtained in several ways, for example, by computing all the moments analytically or by computing them approximatively using samples.

One should keep in mind that this simplified measurement update comes at the expense of a diminished state estimation quality, depending on the degree of the concrete measurement model nonlinearity. In other words, the implicit linearization of a measurement model that is highly nonlinear around the prior state estimate can lead to large errors in the posterior state estimate compared to the unmodified measurement update introduced in Sec. II-B.

IV. THE LINEAR REGRESSION KALMAN FILTER

The Nonlinear Kalman Filter introduced in Sec. III requires the calculation of certain moments to perform time and measurement updates. Doing this analytically provides the Nonlinear Kalman Filter based on statistical linearization with the best possible estimation quality, and should be the means of choice whenever feasible. But, in case of non-existent closed-form solutions, or complicated system and measurement equations, approximative moment calculations have to be performed.

One way to achieve this is to replace the occurring state and noise densities with proper *Dirac mixture* densities, that is, sample-based density representations. This turns the statistical linearization into an approximate statistical linear regression. Consequently, all Nonlinear Kalman Filters using this technique, regardless of whether random or deterministic sampling is used, fall in the class of Linear Regression Kalman Filters.

As only a limited number of samples can be used, this approach always entails a *density approximation*. Therefore, Linear Regression Kalman Filters possess an, in general, inferior estimation quality compared to Nonlinear Kalman Filters based on analytic statistical linearization. Nevertheless, these filters are still efficient and, as no analytic moment calculation is required, are much easier to use.

A. Dirac Mixtures

A Dirac mixture approximation of an arbitrary density function $f_k(\underline{s}_k)$ of an N-dimensional random vector $\underline{s}_k$, encompassing L samples, has the form of

$$\sum_{i=1}^{L} \alpha_{k,i} \cdot \delta(\underline{s}_k - \underline{s}_{k,i}), \tag{17}$$

with sample positions $\underline{s}_{k,i}$ and positive scalar sample weights $\alpha_{k,i}$, for which

$$\sum_{i=1}^{L} \alpha_{k,i} = 1$$

holds [22], [36]. Therefore, the information of the true density $f_k(\underline{s}_k)$ is lossy encoded in the $L \cdot (N+1)$ Dirac mixture parameters. These parameters can be determined in a random fashion by drawing samples randomly according to the true density $f_k(\underline{s}_k)$, or in a deterministic fashion by systematically minimizing a certain distance measure between the true density $f_k(\underline{s}_k)$ and its Dirac mixture approximation (17). Moreover, a combination of both techniques is also possible.

B. Time Update

Our goal is to compute the necessary moments (5) and (6) for the Nonlinear Kalman Filter time update based on Dirac mixtures. Therefore, we have to replace the density product $f_{k-1}^{e}(\underline{x}_{k-1}) \cdot f_k^{w}(\underline{w}_k)$ with an appropriate Dirac mixture. Of course, each density could be approximated separately and the product of the resulting Dirac mixtures built afterwards. However, the result of this density product would be the Cartesian product of the employed state and noise Dirac mixtures, i.e., a Dirac mixture with $L \cdot M$ samples, where L and M denote the respective number of samples of the state and noise Dirac mixtures. This approach would not scale efficiently with an increasing number of employed samples. Fig. 4 illustrates this problem in case of scalar state x_{k-1} and system noise w_k. The state density is approximated with $L = 9$ samples whereas the Dirac mixture for the system noise employs $M = 5$ samples.

Nevertheless, we can do better by exploiting the fact that the state $\underline{x}_{k-1}$ as well as the system noise $\underline{w}_k$ are independent of each other and their respective densities, $f_{k-1}^{e}(\underline{x}_{k-1})$ and $f_k^{w}(\underline{w}_k)$, are Gaussian. That is, the product is equivalent to their, also Gaussian, joint density $f_k^{x,w}(\underline{x}_{k-1}, \underline{w}_k)$ with a zero cross-covariance matrix $\mathbf{C}_k^{x,w}$. Hence, we can avoid the Cartesian product by directly approximating the joint density

$$\begin{aligned} f_k^{x,w}(\underline{x}_{k-1}, \underline{w}_k) &= f_{k-1}^{e}(\underline{x}_{k-1}) \cdot f_k^{w}(\underline{w}_k) \\ &= \mathcal{N}\left(\begin{bmatrix} \underline{x}_{k-1} \\ \underline{w}_k \end{bmatrix}; \begin{bmatrix} \hat{\underline{x}}_{k-1}^{e} \\ \hat{\underline{w}}_k \end{bmatrix}, \begin{bmatrix} \mathbf{C}_{k-1}^{e} & \mathbf{0} \\ \mathbf{0} & \mathbf{C}_k^{w} \end{bmatrix} \right) \end{aligned}$$

using L_k^p samples according to

$$\sum_{i=1}^{L_k^p} \alpha_{k,i}^{p} \cdot \delta\left(\begin{bmatrix} \underline{x}_{k-1} \\ \underline{w}_k \end{bmatrix} - \begin{bmatrix} \underline{x}_{k-1,i} \\ \underline{w}_{k,i} \end{bmatrix} \right), \tag{18}$$

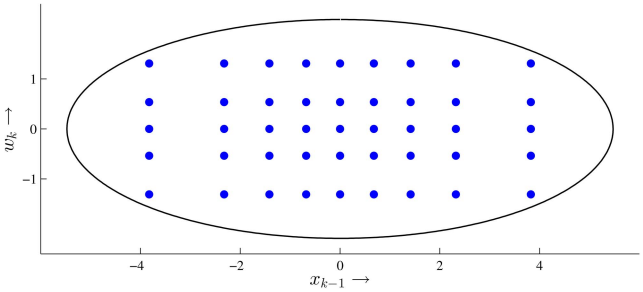

Fig. 4. Cartesian product of separate scalar state and system noise Dirac mixture approximations (blue dots). Covariance matrix (black ellipse) of the true Gaussian joint density $f_k^{x,w}(x_{k-1}, w_k)$ with confidence interval of 95%.

where $\alpha_{k,i}^{p}$ denotes the sample weights and $[\underline{x}_{k-1,i}^{\mathrm{T}}, \underline{w}_{k,i}^{\mathrm{T}}]^{\mathrm{T}}$ the sample positions in the joint space of state and system noise. Plugging this into (5) and (6), and exploiting the Dirac sifting property, we obtain the desired predicted state sample mean

$$\hat{\underline{x}}_k^{p} \approx \sum_{i=1}^{L_k^p} \alpha_{k,i}^{p} \cdot \underline{a}_k(\underline{x}_{k-1,i}, \underline{w}_{k,i}), \tag{19}$$

and predicted state sample covariance

$$\begin{aligned} \mathbf{C}_k^{p} \approx \sum_{i=1}^{L_k^p} & \alpha_{k,i}^{p} \cdot \underline{a}_k(\underline{x}_{k-1,i}, \underline{w}_{k,i}) \\ & \cdot \underline{a}_k(\underline{x}_{k-1,i}, \underline{w}_{k,i})^{\mathrm{T}} - \hat{\underline{x}}_k^{p} \cdot (\hat{\underline{x}}_k^{p})^{\mathrm{T}}, \end{aligned} \tag{20}$$

respectively.

C. Measurement Update

The LRKF measurement update can be computed in the same manner. First, we approximate the joint density

$$\begin{aligned} f_k^{x,v}(\underline{x}_k, \underline{v}_k) &= f_k^{p}(\underline{x}_k) \cdot f_k^{v}(\underline{v}_k) \\ &= \mathcal{N}\left(\begin{bmatrix} \underline{x}_k \\ \underline{v}_k \end{bmatrix}; \begin{bmatrix} \hat{\underline{x}}_k^{p} \\ \hat{\underline{v}}_k \end{bmatrix}, \begin{bmatrix} \mathbf{C}_k^{p} & \mathbf{0} \\ \mathbf{0} & \mathbf{C}_k^{v} \end{bmatrix} \right) \end{aligned}$$

of prior state and measurement noise with the Dirac mixture

$$\sum_{i=1}^{L_k^e} \alpha_{k,i}^{e} \cdot \delta\left(\begin{bmatrix} \underline{x}_k \\ \underline{v}_k \end{bmatrix} - \begin{bmatrix} \underline{x}_{k,i} \\ \underline{v}_{k,i} \end{bmatrix} \right) \tag{21}$$

encompassing L_k^e samples with weights $\alpha_{k,i}^{e}$ and positions $[\underline{x}_{k,i}^{\mathrm{T}}, \underline{v}_{k,i}^{\mathrm{T}}]^{\mathrm{T}}$. Second, plugging this into (14), (15), and (16) yields the measurement sample mean

$$\hat{\underline{y}}_k \approx \sum_{i=1}^{L_k^e} \alpha_{k,i}^{e} \cdot \underline{h}_k(\underline{x}_{k,i}, \underline{v}_{k,i}), \tag{22}$$

measurement sample covariance

$$\begin{aligned} \mathbf{C}_k^{y} \approx \sum_{i=1}^{L_k^e} & \alpha_{k,i}^{e} \cdot \underline{h}_k(\underline{x}_{k,i}, \underline{v}_{k,i}) \\ & \cdot \underline{h}_k(\underline{x}_{k,i}, \underline{v}_{k,i})^{\mathrm{T}} - \hat{\underline{y}}_k \cdot \hat{\underline{y}}_k^{\mathrm{T}}, \end{aligned} \tag{23}$$

and state measurement sample cross-covariance

$$\mathbf{C}_k^{x,y} \approx \sum_{i=1}^{L_k^e} \alpha_{k,i}^e \cdot \underline{x}_{k,i} \cdot \underline{h}_k(\underline{x}_{k,i}, \underline{v}_{k,i})^{\mathrm{T}} - \hat{\underline{x}}_k^p \cdot \hat{\underline{y}}_k^{\mathrm{T}}, \tag{24}$$

respectively. Finally, the desired posterior state mean and covariance are computed by using the Kalman Filter formulas (12) and (13).

D. The LRKF

Algorithm 1 summarizes the general procedure of a Linear Regression Kalman Filter. It is important to note that the Dirac mixture approximations (18) and (21) can be determined in completely different ways (although this is usually not the case) and do not have to utilize the same number of samples. Moreover, in case of pure additive system or measurement noise, the moment calculation can be simplified so that only the state distribution has to be sampled. This reduces the computational burden and improves the estimation quality of the LRKF.

ALGORITHM 1 *Linear Regression Kalman Filter*

Set $f_0^e(\underline{x}_0) = \mathcal{N}(\underline{x}_0, \hat{\underline{x}}_0, \mathbf{C}_0)$
for $k = 1, 2, \ldots$ **do**
  Time Update:
  Compute Dirac mixture approximation (18)
  Compute predicted state moments $\hat{\underline{x}}_k^p$ and $\mathbf{C}_k^p$ according to (19) and (20)
  Set $f_k^p(\underline{x}_k) = \mathcal{N}(\underline{x}_k; \hat{\underline{x}}_k^p, \mathbf{C}_k^p)$
  if measurement $\tilde{\underline{y}}_k$ is available **then**
    Measurement Update:
    Compute Dirac mixture approximation (21)
    Compute measurement moments $\hat{\underline{y}}_k$, $\mathbf{C}_k^y$, and $\mathbf{C}_k^{x,y}$ according to (22), (23), and (24)
    Compute posterior state moments $\hat{\underline{x}}_k^e$ and $\mathbf{C}_k^e$ according to (12) and (13)
    Set $f_k^e(\underline{x}_k) = \mathcal{N}(\underline{x}_k; \hat{\underline{x}}_k^e, \mathbf{C}_k^e)$
  else
    Set $f_k^e(\underline{x}_k) = f_k^p(\underline{x}_k)$
  end if
end for

V. THE SMART SAMPLING KALMAN FILTER

Sec. IV dealt with the general Linear Regression Kalman Filter. In order to use it, appropriate Dirac mixture approximations of the non-standard Gaussian joint densities (18) and (21) have to be determined, i.e., sets of samples with their respective positions and weights.

Our goal is to create sample sets in a deterministic manner encompassing an arbitrary number of equally weighted samples placed in the entire relevant regions of the state space, i.e., not only on the principal axes. For that reason, we turn this density approximation problem into an optimization problem by utilizing a Dirac mixture approximation procedure based on a combination of the Localized Cumulative Distribution (LCD) and a modified Cramér-von Mises distance as described in [37], [38].

Even though the LCD approach can approximate any non-standard Gaussian, it is computationally expensive due to its costly optimization procedure and, thus, is not well suited for online filter execution. But thanks to the deterministic manner of the LCD approach, we can reuse a single computed Dirac mixture approximation for every time and measurement update. By this means, we circumvent the costly online optimization of a non-standard Gaussian by computing a Dirac mixture approximation of a standard normal distribution offline and only transforming it online (during time and measurement updates) to any non-standard Gaussian using the Mahalanobis transformation [39]. In the following, we recapitulate this optimization problem and its main definitions from [38].

A. The LCD Approach

The considered problem is to determine the optimal sample positions $\underline{s}_i$ of an equally weighted Dirac mixture approximation

$$\frac{1}{L} \sum_{i=1}^{L} \delta(\underline{s} - \underline{s}_i) \tag{25}$$

of an N-dimensional standard normal distribution $\mathcal{N}(\underline{s}; \underline{0}, \mathbf{I})$. We denote the L samples, that is, the $N \cdot L$ sampling parameters, as the set

$$S := \{\underline{s}_1, \ldots, \underline{s}_L\}.$$

In order to determine S in an optimal way, we have to assess the quality of the Dirac mixture approximation (25) by defining some distance measure between both densities. Unfortunately, the classical Cumulative Distribution Function (CDF), which is often used for one-dimensional distributions, cannot be used for the multivariate case due to its non-uniqueness and asymmetry [37]. A solution is to use the Localized Cumulative Distribution, which considers the probability mass around each point of the distribution in a certain manner.

DEFINITION V.1 (Localized Cumulative Distribution) Let $f(\underline{s})$ be an N-dimensional density function. The corresponding Localized Cumulative Distribution is defined as

$$F(\underline{m}, b) = \int_{\mathbb{R}^N} f(\underline{s}) \cdot K(\underline{s} - \underline{m}, b) d\underline{s},$$

with $b \in \mathbb{R}_+$ and the symmetric and integrable kernel

$$K(\underline{s} - \underline{m}, b) = \prod_{k=1}^{N} \exp\left(-\frac{1}{2} \frac{(s^{(k)} - m^{(k)})^2}{b^2} \right).$$

Here, $\underline{m}$ characterizes the location of the kernel and b its size.

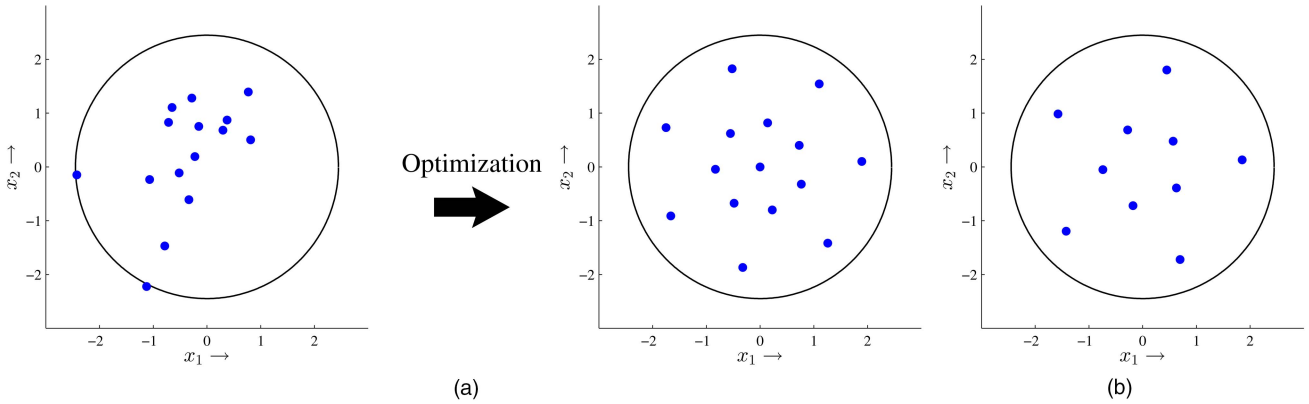

Fig. 5. LCD sampling of a two-dimensional standard normal distribution. Covariance matrices with confidence interval of 95% (black circles). The excellent state space coverage can be clearly seen. (a) LCD approach with 15 samples. Random Dirac mixture initialization on the left and optimization result (final Dirac mixture) on the right. (b) An optimization result in case of 10 samples.

Using the Dirac sifting property, the LCD of the Dirac mixture (25) can be obtained according to

$$F_{\mathrm{DM}}(S,\underline{m},b) = \frac{1}{L}\sum_{i=1}^{L}\prod_{k=1}^{N}\exp\left(-\frac{1}{2}\frac{(s_i^{(k)}-m^{(k)})^2}{b^2}\right),$$

whereas the LCD of an N-dimensional standard Gaussian is given as

$$F_{\mathcal{N}}(\underline{m},b) = \frac{b^N}{\left(\sqrt{1+b^2}\right)^N}\prod_{k=1}^{N}\exp\left(-\frac{1}{2}\frac{(m^{(k)})^2}{(1+b^2)}\right).$$

Now, we can compare both densities by comparing their respective LCDs using a modified Cramér-von Mises distance defined as follows.

DEFINITION V.2 (Modified Cramér-von Mises Distance) The modified Cramér-von Mises distance D between two LCDs $F(\underline{m},b)$ and $\tilde{F}(\underline{m},b)$ is given by

$$D = \int_{\mathbb{R}_+} w(b)\int_{\mathbb{R}^N}(F(\underline{m},b)-\tilde{F}(\underline{m},b))^2 d\underline{m}db$$

with weighting function

$$w(b) = \begin{cases}\dfrac{1}{b^{N-1}}, & b\in(0,b_{\max}]\\ 0, & \text{elsewhere.}\end{cases}$$

The modified Cramér-von Mises distance between the LCDs $F_{\mathrm{DM}}(\cdot,\cdot,\cdot)$ and $F_{\mathcal{N}}(\cdot,\cdot)$ is given by

$$D(S) = D_1 - 2D_2(S) + D_3(S) \quad \text{with } D_i = \int_{\mathbb{R}_+} P_i db, \tag{26}$$

and the sample-independent part

$$P_1 = \frac{\pi^{N/2}b^{N+1}}{\left(\sqrt{1+b^2}\right)^N},$$

as well as the sample-dependent parts

$$P_2(S) = \frac{(2\pi)^{N/2}b^{N+1}}{L\left(\sqrt{1+2b^2}\right)^N}\cdot\sum_{i=1}^{L}\exp\left(-\frac{1}{2}\sum_{k=1}^{N}\frac{(s_i^{(k)})^2}{1+2b^2}\right)$$

and

$$P_3(S) = \frac{\pi^{N/2}b}{L^2}\cdot\sum_{i=1}^{L}\sum_{j=1}^{L}\exp\left(-\frac{1}{2}\sum_{k=1}^{N}\frac{(s_i^{(k)}-s_j^{(k)})^2}{2b^2}\right).$$

Given this distance measure, the optimal sample positions $\underline{s}_i$ are computed as follows. One starts by randomly choosing initial sampling parameters S, i.e., placing L N-dimensional samples randomly in state space, where L is the cardinality of the desired optimal Dirac mixture approximation and N the dimension of the considered standard normal distribution. Then, an optimization procedure, e.g., a quasi-Newton method (L-BFGS) [40], [41], changes these initial sampling parameters S, i.e., moves the samples in state space, such that the distance measure (26) between the standard normal and its Dirac mixture approximation is minimized. Thus, we perform a *global optimization* as all sample positions $\underline{s}_i$ are optimized at once. Another solution would be to use greedy optimizations such as [42], where an existent Dirac mixture is extended by simply adding additional samples and leaving the existing samples unchanged. Unfortunately, this leads to suboptimal approximation results and, hence, is not considered here.

Fig. 5(a) illustrates the proposed LCD approach for the case of a two-dimensional standard normal distribution and $L = 15$ samples. The random initialization is shown on the left, whereas the final optimal approximation is shown on the right. Fig. 5(b) depicts another optimization result with $L = 10$ samples.

Now, given a non-standard Gaussian distribution

$$\mathcal{N}(\underline{z};\hat{\underline{z}},\mathbf{C}^z) \tag{27}$$

during filter execution, i.e., the joint densities $f_k^{x,w}(\underline{x}_{k-1},\underline{w}_k)$ and $f_k^{x,v}(\underline{x}_k,\underline{v}_k)$, we compute the matrix square root $\sqrt{\mathbf{C}^z}$ of $\mathbf{C}^z$ using the Cholesky decomposition,[5] and individually translate, rotate, and scale each

[5] Other matrix square root operations, such as the eigendecomposition, are also possible.

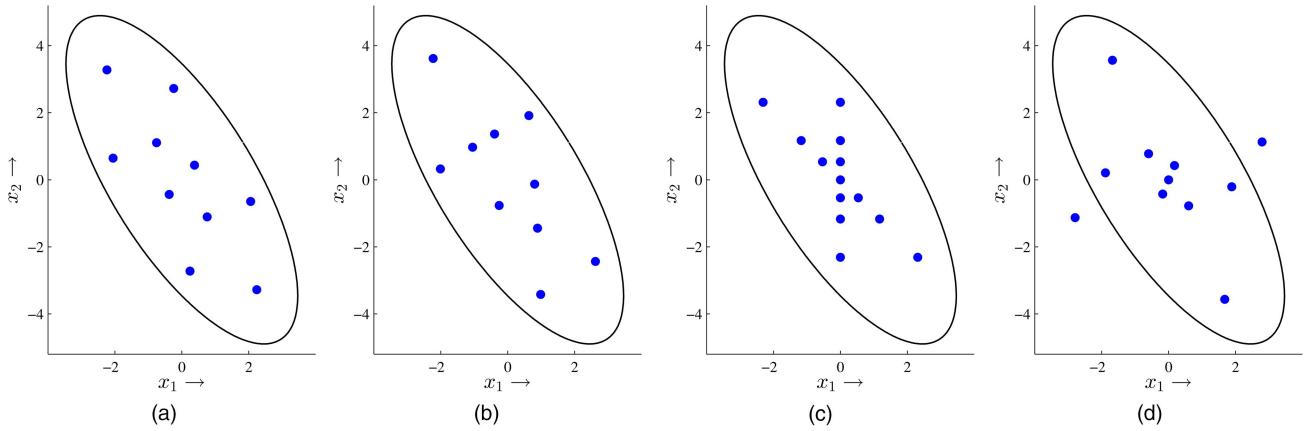

Fig. 6. Difference between the direct LCD approximation (a) and the standard normal LCD approximation (same samples as depicted in Fig. 5(b)) with subsequent transformation using a Cholesky decomposition (b). For comparison, we transformed sample sets from the GF and the RUKF (same samples as depicted in Fig. 1) as well. (a) Direct approximation. (b) Suboptimal approximation. (c) GF approximation. (d) RUKF approximation.

sample of (25) according to

$$\underline{z}_i = \sqrt{\mathbf{C}^z} \cdot \underline{s}_i + \hat{\underline{z}} \quad \forall i \in \{1,\ldots,L\}, \tag{28}$$

so that the new sample set

$$Z := \{\underline{z}_1,\ldots,\underline{z}_L\}$$

forms the Dirac mixture approximation

$$\frac{1}{L}\sum_{i=1}^{L}\delta(\underline{z}-\underline{z}_i)$$

of the non-standard Gaussian (27). Note that these non-standard Gaussian samples are still equally weighted.

It is important to note that this combination of standard normal approximation with subsequent transformation delivers only suboptimal results compared to a direct LCD-based approximation of the non-standard Gaussian. Fig. 6 exemplifies this problem for a two-dimensional non-standard, i.e., rotated and scaled, Gaussian distribution and Dirac mixtures with 10 samples. The direct LCD approximation of the considered Gaussian is depicted in Fig. 6(a). One can see an optimal, regular placement of the samples, covering the relevant state space regions homogeneously. In contrast to this, a suboptimal solution is shown in Fig. 6(b). Here, larger regions of the relevant state space are uncovered, e.g., the top-left region of the Gaussian or its middle part. Nevertheless, all state-of-the-art LRKFs suffer from the problem of a suboptimal non-standard Gaussian approximation, as they rely on the online Mahalanobis transformation, too. To demonstrate this, we transformed sample sets from the GF and the RUKF as well (see Fig. 6(c) and Fig. 6(d)).

B. The New LRKF

By using offline computed LCD-based Dirac mixture approximations of standard normal distributions (25) in combination with online transformations (28) during the LRKF time and measurement updates, we introduce the new Smart Sampling Kalman Filter (S^{2}KF), with its powerful feature of using an arbitrary number of optimally placed samples in the entire state space. There exist no sampling constraints such as axis-aligned samples or that the number of samples must be a multiple of the state dimension as with the UKF, GF, or RUKF.

As will be shown in the evaluation, with an increasing number of used samples in (25) the S^{2}KF converges to the analytic statistical linearization as the resulting Dirac mixture approximation of the standard normal distribution becomes more accurate. In contrast to the UKF with its fixed-size sample set, this allows an extensive evaluation of the given system and measurement models, as more and more samples become available in the relevant regions of the state space. Moreover, this makes a non-positive definite state covariance matrix very unlikely and the filter more reliable. As a consequence, the estimation quality can be easily improved by simply increasing the number of employed samples, which offers an intuitive optimization parameter. Of course, this effect is also true for filters relying on random sampling. But, due to the optimal sample placement, the S^{2}KF converges much faster, so that already a small number of samples provides an excellent estimation quality. Regarding the filter complexity, assuming that the Dirac mixture approximations of the required standard normal distributions are already computed, the computational complexity of the S^{2}KF grows only linearly with the number of used samples L for a fixed dimension N.

One should keep in mind that the LCD approach cannot create unique Dirac mixture approximations for a given dimension N and number of samples L as the standard normal distribution is rotation-invariant and the optimization procedure is initialized with random samples. However, due to the *reuse* of offline computed Dirac mixture approximations, the S^{2}KF results still become reproducible. In other words, executing the S^{2}KF

with the same inputs multiple times, i.e., model parameters, initial state, and measurements, will always produce the same results as the same sample approximations are used for each execution.

C. The Sample Cache

The proposed S^{2}KF needs several LCD-based Dirac mixture approximations of standard normal distributions, one for each required combination of dimension N and number of employed samples L, depending on the concrete filtering problem (state dimension, noise dimension, and selected filter accuracy). An option to obtain these approximations would be to recompute all required approximations before each program start. However, for large dimensions and/or number of samples, this can be very time-consuming. Moreover, the estimation results from different program executions would be different as always new sample sets would be used.

For that reason, we introduce a sample storage called *Sample Cache*. Whenever a requested sample set for a given combination of dimension N and number of samples L is not available during S^{2}KF execution, it is computed on demand,[6] that is, *transparent for the user*, and subsequently stored persistently in the file system for later reuse. Over time, the Sample Cache grows and the necessity for time-consuming sample generation becomes more unlikely. Of course, if the user knows all the needed approximations before filter execution, all of them can be computed and stored in Sample Cache in advance so that no sample computation is required at all during filter execution.

VI. EVALUATION

In this section, we compare the new S^{2}KF with state-of-the-art LRKFs by performing recursive state estimation using various nonlinear system and measurement equations. In the first evaluation, the focus lies on nonlinear prediction, whereas in the second evaluation the filters have to cope with nonlinear measurement updates.

As every LRKF is an approximation of the Kalman Filter based on analytic statistical linearization, an LRKF estimate should be as close as possible to the this estimate. A considerable difference between both estimates can only result from inaccurate moment calculations by the LRKF and, hence, its utilized Gaussian sampling technique. Consequently, in order to assess the investigated LRKFs and their used sampling techniques properly, the state estimates obtained by analytic statistical linearization will serve in both evaluations as reference (ground truth) here. Using another ground truth, for example the true system state, would be unfavorable to detect such inaccurate moment calculations. The reason is that an LRKF estimate that is close to the true system state does not indicate whether the moments were calculated correctly, as the estimate obtained by analytic statistical linearization might be quite different from the true system state.

A. Batch Reactor

We consider the gas-phase reaction proposed in [43], resulting in the estimation problem consisting of a two-dimensional state $\underline{x}_k = [x_{a,k}, x_{b,k}]^{\mathrm{T}}$, which obeys the time-invariant nonlinear system model

$$\begin{aligned}\underline{x}_k &= \underline{a}(\underline{x}_{k-1}, \Delta t, \underline{w}) \\ &= \underline{x}_{k-1} + \Delta t \cdot \begin{bmatrix} -0.32 \cdot x_{a,k-1}^2 \\ 0.16 \cdot x_{a,k-1}^2 \end{bmatrix} + \underline{w}, \end{aligned} \tag{29}$$

with input $\Delta t = 0.1$ and time-invariant, additive, and zero-mean Gaussian white noise $\underline{w}$ with covariance

$$\mathbf{C}^w = \mathrm{diag}(10^{-5}, 10^{-5}).$$

Over time, we receive scalar measurements $\tilde{y}_k$ according to the time-invariant linear mapping

$$y_k = h(\underline{x}_k, v) = [1 \;\; 1] \cdot \underline{x}_k + v, \tag{30}$$

where v denotes time-invariant, additive, and zero-mean Gaussian white noise with variance $\mathbf{C}^v = 0.1$.

We compare the following estimators:

- exact, analytic statistical linearization using [44], for which closed-form expressions are given in the Appendix,
- the UKF with equally weighted samples,
- the GF with 25 samples on each principal axis,
- the RUKF with 12 iterations, and finally
- the new S^{2}KF with 10, 20, 50, 100, and 150 samples, respectively, in order to demonstrate the convergence of the S^{2}KF towards the analytic statistical linearization.

As the measurement equation is linear in this simulation, we can accurately evaluate the nonlinear prediction performance of the investigated filters. More precisely, the measurement update is calculated in closed-form by all filters, i.e., the optimal closed-form Kalman Filter update is used. Additionally, due to the fact that the system model (29) is corrupted by pure additive noise, sampling is reduced to the two-dimensional state space (see Sec. IV-D). Table II summarizes the resulting numbers of samples used by each LRKF for the prediction step.

The simulation consists of $R = 1000$ Monte Carlo runs. For each Monte Carlo run, the true system state is obtained by initializing it with $\underline{x}_0 = [3, 1]^{\mathrm{T}}$ and recursively propagating it 50 times, together with noise realizations of $\underline{w}$, through the system model (29), resulting in a simulation with 50 time steps. Additionally, we simulate one noisy measurement each time step by using (30) together with a noise realization of v. All filters are initialized with mean $\hat{\underline{x}}_0^e = [0.5, 3.5]^{\mathrm{T}}$ and covariance $\mathbf{C}_0^e = \mathrm{diag}(10, 10)$.

[6] As a consequence, in such a case the filter execution stalls until the sample set is computed.

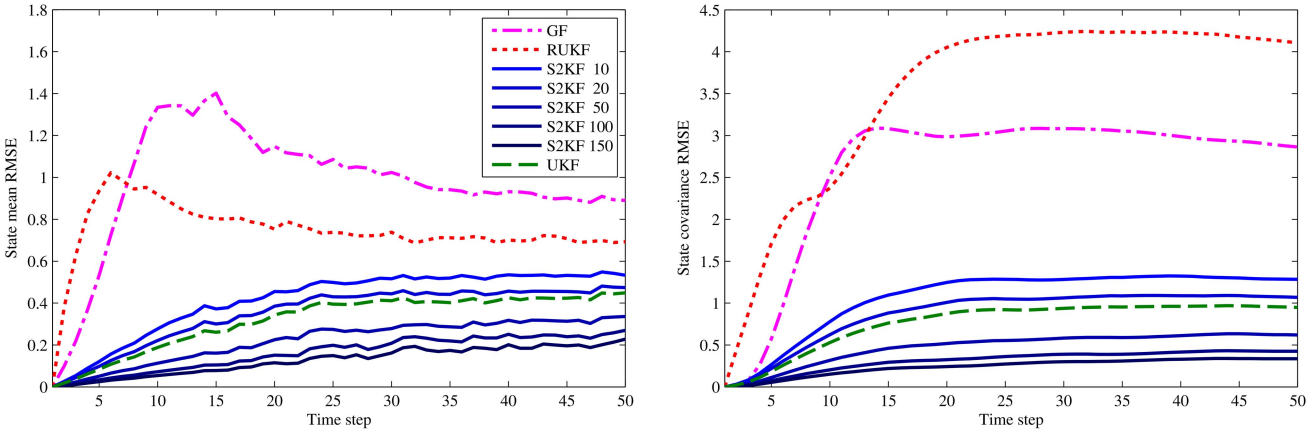

Fig. 7. Batch reactor simulation over 50 time steps. (a) State mean RMSE $\bar{x}_k$. (b) State covariance RMSE $\bar{C}_k$.

TABLE II
Employed LRKFs and their respective sampling settings for the batch reactor simulation.

LRKF	Number of samples	Sample placement
UKF	$2 \cdot 2 + 1 = 5$	Axes only
GF	$24 \cdot 2 + 1 = 49$	Axes only
RUKF	$12 \cdot (2 \cdot 2) + 1 = 49$	Entire state space
S^2KF	10	Entire state space
S^2KF	20	Entire state space
S^2KF	50	Entire state space
S^2KF	100	Entire state space
S^2KF	150	Entire state space

In order to assess the estimation quality of each LRKF, we compute the Root Mean Square Error (RMSE) of their posterior state mean over all simulation runs with respect to the analytic statistical linearization posterior state mean, that is,

$$\bar{x}_k = \sqrt{\frac{1}{R}\sum_{r=1}^{R} \|\hat{\underline{x}}_k^{(r)} - \hat{\underline{x}}_{a,k}^{(r)}\|_2^2},$$

where $\hat{\underline{x}}_k^{(r)}$ denotes the respective LRKF state mean and $\hat{\underline{x}}_{a,k}^{(r)}$ the state mean of the analytic statistical linearization. The results are depicted in Fig. 7(a). Here, the GF shows a very high RMSE at the beginning. Over time, the RMSE decreases but remains at a relatively high level. The RUKF does not possess such extreme RMSE but it is constantly at a higher level compared to the UKF and S^2KF estimates. As opposed to this, the UKF delivers quite good results although the GF uses much more samples per axis. Moreover, all S^2KF instances are also much better than the GF and RUKF, and the S^2KF instances using 50 or more samples deliver the best posterior state means of all investigated LRKFs. The expected convergence of the S^2KF with an increasing number of samples towards the analytic statistical linearization can be clearly seen.

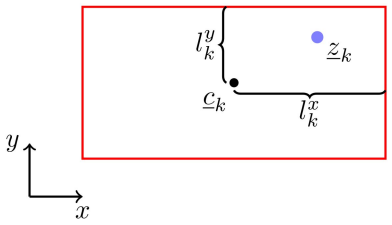

Fig. 8. Axis-aligned extended rectangular target with position $\underline{c}_k$, extent $\underline{l}_k$, and a target surface point $\underline{z}_k$.

Moreover, we compute the RMSE for their posterior state covariance in a similar manner according to

$$\bar{C}_k = \sqrt{\frac{1}{R}\sum_{r=1}^{R} \|\mathbf{C}_k^{(r)} - \mathbf{C}_{a,k}^{(r)}\|^2},$$

where $\|\cdot\|$ denotes the Frobenius norm, $\mathbf{C}_k^{(r)}$ the respective LRKF state covariance, and $\mathbf{C}_{a,k}^{(r)}$ the state covariance of the analytic statistical linearization. When looking at the results shown in Fig. 7(b), one should notice that the GF as well as the RUKF estimate themselves much too uncertain compared to the analytic moment calculation. Both errors increase quickly and decrease only at a slow pace over time. In contrast, the covariance of the UKF is much closer to the analytic statistical linearization than these filters. However, as with the state mean, the S^2KF instances can outperform the UKF and its convergence towards analytic statistical linearization is as expected.

B. Extended Target Tracking

In this section, we evaluate the S^2KF by means of tracking an extended target modeled as multiplicative noise. Our goal is to estimate the position $\underline{c}_k = [c_k^x, c_k^y]^\mathrm{T}$ and extent $\underline{l}_k = [l_k^x, l_k^y]^\mathrm{T}$ of a two-dimensional axis-aligned rectangular target (see Fig. 8). The hidden system state is given by $\underline{x}_k = [\underline{l}_k^\mathrm{T}, \underline{c}_k^\mathrm{T}]^\mathrm{T}$.

In order to incorporate target information into our state estimate, we assume uniformly distributed, noisy measurements stemming from the surface of the target.

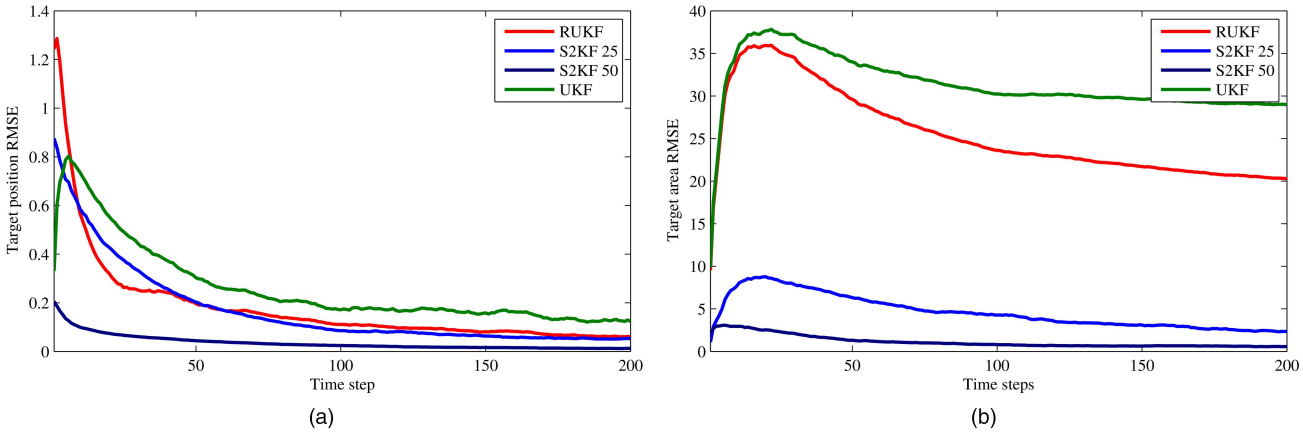

Fig. 9. Extended target tracking evaluation results. (a) RMSE for the target position. (b) RMSE for the target area.

For this purpose, we extend the approach proposed in [6]. The basic idea is that each point of the target surface can be reached by scaling the axis lengths l_k^x and l_k^y individually and adding the center $\underline{c}_k$, i.e.,

$$\underline{z}_k = \mathbf{H} \cdot \underline{l}_k + \underline{c}_k,$$

with uncorrelated state independent multiplicative noise

$$\mathbf{H} = \mathrm{diag}(h^x, h^y).$$

As the measurements are uniformly distributed, h^x and h^y also have to be uniformly distributed in the interval $[-1,1]$ (see Fig. 8). Taking the measurement noise into account yields the preliminary nonlinear measurement equation

$$\underline{m}_k = \underline{z}_k + \underline{w} = \mathbf{H} \cdot \underline{l}_k + \underline{c}_k + \underline{w}, \tag{31}$$

where $\underline{w}$ denotes additive, zero-mean Gaussian white noise with unit covariance.

Unfortunately, as mentioned in [6], linear estimators, including the S^{2}KF as well, are unsuitable for tracking extended targets modeled this way. To overcome this issue, we pick up on the author's suggestion and extend the measurement equation (31) to match the best quadratic estimator according to

$$\underline{y}_k = \begin{bmatrix} \underline{m}_k \\ \underline{m}_k^2 \end{bmatrix} = \begin{bmatrix} \mathbf{H} \cdot \underline{l}_k + \underline{c}_k + \underline{w} \\ (\mathbf{H} \cdot \underline{l}_k + \underline{c}_k + \underline{w})^2 \end{bmatrix}.$$

To keep things simple, this evaluation uses a static target with extent $\underline{l} = [4,2]^\mathrm{T}$ located at $\underline{c} = [3,5]^\mathrm{T}$. Thus, the temporal evolution of $\underline{x}_k$ is modeled as random walk, i.e., employing the linear system equation

$$\underline{x}_k^p = \underline{x}_{k-1}^e + \underline{v},$$

where $\underline{v}$ is an additive, zero-mean Gaussian white noise with covariance

$$\mathbf{C}^v = \mathrm{diag}(10^{-4}, 10^{-4}, 10^{-3}, 10^{-3}).$$

We compare the following estimators:

- exact, analytic statistical linearization using [44],
- the UKF with $2 \cdot 8 + 1 = 17$ equally weighted samples,
- the RUKF with 10 iterations (resulting in $10 \cdot (2 \cdot 8) + 1 = 161$ samples), and
- two S^{2}KF instances with 25 and 50 samples, respectively, in order to demonstrate the convergence of the S^{2}KF towards the analytic statistical linearization.

Due to the fact that the S^{2}KF and the RUKF require a measurement noise described in terms of a Gaussian distribution,[7] and while the UKF only considers the first two moments of the measurement noise, we approximate the uniformly distributed multiplicative noise $\mathbf{H}$ as Gaussian distribution by means of moment matching. The simulation consists of $R = 100$ Monte Carlo runs. For each Monte Carlo run, the initial state estimate is set to $\hat{\underline{x}}_0^e = [1,1,0,0]^\mathrm{T}$ and $\mathbf{C}_0^e = \mathbf{I}_4$, and at each time step we receive a single noisy measurement from the target surface.

Similar to the batch reactor simulation, we assess the estimation quality of each LRKF by computing the RMSE of their target position estimate over all simulation runs with respect to the analytic statistical linearization target position estimate, that is,

$$\bar{c}_k = \sqrt{\frac{1}{R} \sum_{r=1}^{R} \| \hat{\underline{c}}_k^{(r)} - \hat{\underline{c}}_{a,k}^{(r)} \|_2^2},$$

where $\hat{\underline{c}}_k^{(r)}$ denotes the respective LRKF position estimate and $\hat{\underline{c}}_{a,k}^{(r)}$ the analytic statistical linearization position estimate. The results of the target position RMSE are depicted in Fig. 9(a). Here, all LRKFs quickly decrease their RMSE over time. However, the UKF converges to a little higher RMSE than the other LRKFs. The S^{2}KF using 50 samples converges quickly to an error nearly zero and yields the best estimation result of all LRKFs.

Additionally, we compare their target extent estimate by computing the RMSE of the estimated target area in

[7] This is a consequence of the fact that these filters rely on explicit sampling a Gaussian distribution.

a similar manner according to

$$\bar{A}_k = \sqrt{\frac{1}{R}\sum_{r=1}^{R}(4\cdot(\hat{l}_k^x)^{(r)}\cdot(\hat{l}_k^y)^{(r)} - 4\cdot(\hat{l}_{a,k}^x)^{(r)}\cdot(\hat{l}_{a,k}^y)^{(r)})^2},$$

where $[\hat{l}_k^x, \hat{l}_k^y]^{(r)}$ denotes the respective LRKF target extent estimate and $[\hat{l}_{a,k}^x, \hat{l}_{a,k}^y]^{(r)}$ the analytic statistical linearization target extent estimate. Fig. 9(b) shows the RMSE for the estimated target area. The 50 sample S^{2}KF is the only filter that directly converges to a small error, whereas the UKF and the RUKF quickly increase to a high RMSE and decrease only at a very low rate. The UKF area estimate is clearly incorrect, as its area error converges to approximately 28 m^2. As opposed to this, the S^{2}KF using 25 samples corrects its estimate relatively fast.

Fig. 10 shows a representative simulation run after incorporating 75 measurements. One can see that the UKF leaves its initial state estimate of a target of 4 m^2 completely unchanged which coincides with its area RMSE of approximately 28 m^2, and that the RUKF estimates the target much too small. In contrast to this, the S^{2}KF using 50 samples is almost identical to the analytic statistical linearization estimate. The general S^{2}KF convergence towards the analytic statistical linearization concerning both target position and extent is evident as the 50 sample instance yields the much better tracking results than the S^{2}KF using 25 samples.

C. Summary

The performed evaluations showed a general problem of sample-based filtering: not only the amount of samples and their placement are important for the estimation results, but also their interaction with the underlying system and measurement models. This was seen in the two following cases. On the one hand, the axis-aligned placement of the UKF samples deliver quite good results with the batch reactor model, but completely failed during the extended object tracking. On the other hand, the RUKF had problems in both evaluations although it places its samples not only the axes. In contrast, with the ability to use an arbitrary amount of samples with optimal placement in the relevant regions of the state space, the new S^{2}KF can easily be tuned to perform well in both filtering problems.

VII. CONCLUSIONS

In this paper, we introduced a new accurate LRKF called Smart Sampling Kalman Filter (S^{2}KF). It is based on LCD-based Dirac mixture approximations of standard normal distributions comprising an arbitrary number of samples, which are placed optimally in the relevant regions of the state space, that is, not only on the principal axes. Hence, the S^{2}KF can be seen as the ultimate generalization of all sample-based Nonlinear Kalman Filters.

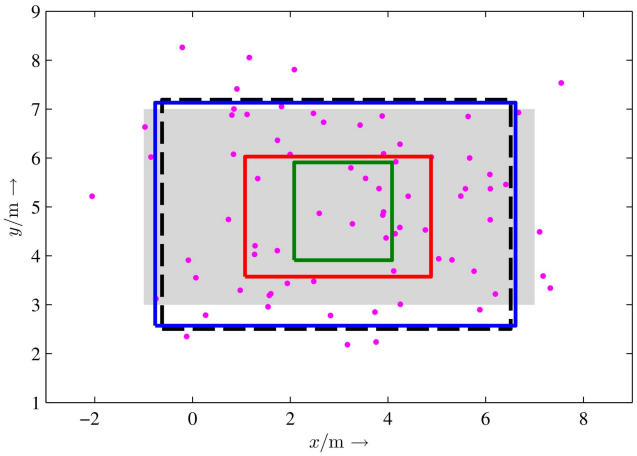

Fig. 10. Representative simulation run with extended target (gray rectangle), randomly generated noisy measurements (magenta dots), analytic statistical linearization estimate (black dashed line), 50 sample S^{2}KF estimate (blue line), UKF estimate (green line), and RUKF estimate (red line).

First, we gave a general introduction to Gaussian estimators and Nonlinear Kalman Filters. We explained the idea of analytic statistical linearization and its approximation in form of the Linear Regression Kalman Filter. After that, we described the optimal Gaussian sampling using the LCD approach and based on this introduced the S^{2}KF. Moreover, we proposed the idea of a Sample Cache that stores computed Dirac mixture approximations of standard normal distributions persistent in the file system for later reuse. Finally, we evaluated the S^{2}KF by means a nonlinear prediction scenario and extended target tracking against state-of-the-art LRKFs. The new filter showed the expected convergence towards the analytic statistical linearization and outperformed state-of-the-art LRKFs including the UKF and RUKF.

As the S^{2}KF encompasses the same structure as the UKF, the S^{2}KF can easily replace it in order to enhance existing and future filtering applications. Morever, the S^{2}KF can be directly used in Gaussian mixture LRKFs [25], [45] and LRKF square root implementations such as described in [46].

ACKNOWLEDGMENT

We would like to thank the anonymous reviewers for their positive feedback and helpful suggestions.

APPENDIX

Our goal is to calculate the predicted state mean and covariance of the nonlinear dynamic system considered in Sec. VI-A in closed-form. Given the state mean

$$\underline{\hat{x}}_{k-1}^e = [\hat{x}_{a,k-1}, \hat{x}_{b,k-1}]^{\mathrm{T}}$$

and state covariance

$$\mathbf{C}_{k-1}^e = \begin{bmatrix} c_{aa,k-1}^x & c_{ab,k-1}^x \\ c_{ab,k-1}^x & c_{bb,k-1}^x \end{bmatrix}$$

from the last time step $k-1$, and the system noise statistics

$$\underline{\hat{w}}_k = [\hat{x}_{a,k}, \hat{x}_{b,k}]^{\mathrm{T}}$$

and

$$\mathbf{C}_k^w = \begin{bmatrix} c_{aa,k}^w & 0 \\ 0 & c_{bb,k}^w \end{bmatrix}$$

from the current time step k, we obtain the moments

$$\begin{aligned}
\mathrm{E}[x_{a,k-1}^2] &= \hat{x}_{a,k-1}^2 + c_{aa,k-1}^x, \\
\mathrm{E}[x_{a,k-1}^3] &= \hat{x}_{a,k-1}^3 + 3 \cdot \hat{x}_{a,k-1} \cdot c_{aa,k-1}^x, \\
\mathrm{E}[x_{a,k-1}^4] &= \hat{x}_{a,k-1}^4 + 6 \cdot \hat{x}_{a,k-1}^2 \cdot c_{aa,k-1}^x \\
&\quad + 3 \cdot (c_{aa,k-1}^x)^2,
\end{aligned}$$

$$\begin{aligned}
\mathrm{E}[x_{b,k-1}^2] &= \hat{x}_{b,k-1}^2 + c_{bb,k-1}^x, \\
\mathrm{E}[x_{a,k-1} x_{b,k-1}] &= \hat{x}_{a,k-1} \hat{x}_{b,k-1} + c_{ab,k-1}^x, \\
\mathrm{E}[x_{a,k-1}^2 x_{b,k-1}] &= \hat{x}_{a,k-1}^2 \hat{x}_{b,k-1} + c_{aa,k-1}^x \cdot \hat{x}_{b,k-1} \\
&\quad + 2 \cdot c_{ab,k-1}^x \cdot \hat{x}_{a,k-1}, \\
\mathrm{E}[w_{a,k}^2] &= \hat{x}_{a,k}^2 + c_{aa,k}^w, \\
\mathrm{E}[w_{b,k}^2] &= \hat{x}_{b,k}^2 + c_{bb,k}^w,
\end{aligned}$$

respectively. Using these moments together with $p = -0.32$ and $q = 0.16$, we obtain the predicted state mean according to

$$\underline{\hat{x}}_k^p = \begin{bmatrix} \hat{x}_{a,k-1} + p \cdot \Delta t \cdot \mathrm{E}[x_{a,k-1}^2] \\ \hat{x}_{b,k-1} + q \cdot \Delta t \cdot \mathrm{E}[x_{a,k-1}^2] \end{bmatrix} + \underline{\hat{w}}_k,$$

and the predicted state covariance matrix according to

$$\mathbf{C}_k^p = \begin{bmatrix} m_{aa,k} & m_{ab,k} \\ m_{ab,k} & m_{bb,k} \end{bmatrix} - \underline{\hat{x}}_k^p \cdot (\underline{\hat{x}}_k^p)^{\mathrm{T}},$$

with

$$\begin{aligned}
m_{aa,k} &= \mathrm{E}[x_{a,k-1}^2] + (p \cdot \Delta t)^2 \cdot \mathrm{E}[x_{a,k-1}^4] + \mathrm{E}[w_{a,k}^2] \\
&\quad + 2 \cdot (p \cdot \Delta t \cdot \mathrm{E}[x_{a,k-1}^3] + \hat{x}_{a,k-1} \cdot \hat{x}_{a,k} \\
&\quad + p \cdot \Delta t \cdot \mathrm{E}[x_{a,k-1}^2] \cdot \hat{x}_{a,k}), \\
m_{bb,k} &= \mathrm{E}[x_{b,k-1}^2] + (q \cdot \Delta t)^2 \cdot \mathrm{E}[x_{a,k-1}^4] + \mathrm{E}[w_{b,k}^2] \\
&\quad + 2 \cdot (q \cdot \Delta t \cdot \mathrm{E}[x_{a,k-1}^2 x_{b,k-1}] + \hat{x}_{b,k-1} \cdot \hat{x}_{b,k} \\
&\quad + q \cdot \Delta t \cdot \mathrm{E}[x_{a,k-1}^2] \cdot \hat{x}_{b,k}), \\
m_{ab,k} &= \mathrm{E}[x_{a,k-1} x_{b,k-1}] + q \cdot \Delta t \cdot \mathrm{E}[x_{a,k-1}^3] \\
&\quad + \hat{x}_{a,k-1} \cdot \hat{x}_{b,k} + p \cdot \Delta t \cdot \mathrm{E}[x_{a,k-1}^2 x_{b,k-1}] \\
&\quad + p \cdot q \cdot \Delta t^2 \cdot \mathrm{E}[x_{a,k-1}^4] \\
&\quad + p \cdot \Delta t \cdot \mathrm{E}[x_{a,k-1}^2] \cdot \hat{x}_{b,k} \\
&\quad + \hat{x}_{a,k} \cdot (\hat{x}_{b,k-1} + q \cdot \Delta t \cdot \mathrm{E}[x_{a,k-1}^2] + \hat{x}_{b,k}).
\end{aligned}$$

REFERENCES

[1] Branko Ristic, Sanjeev Arulampalam, and Neil Gordon
Beyond the Kalman Filter: Particle Filters for Tracking Applications.
Artech House Publishers, 2004.

[2] Yaakov Bar-Shalom, X. Rong Li, and Thiagalingam Kirubarajan
Estimation with Applications to Tracking and Navigation.
New York Chichester Weinheim Brisbane Singapore Toronto: Wiley-Interscience, 2001.

[3] Johann Wolfgang Koch
"Bayesian Approach to Extended Object and Cluster Tracking Using Random Matrices,"
IEEE Transactions on Aerospace and Electronic Systems, vol. 44, no. 3, pp. 1042–1059, Jul. 2008.

[4] Frederik Beutler, Marco F. Huber, and Uwe D. Hanebeck
"Semi-Analytic Stochastic Linearization for Range-Based Pose Tracking,"
in *Proceedings of the 2010 IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI 2010)*, Salt Lake City, USA, Sep. 2010, pp. 44–49.

[5] Marcus Baum and Uwe D. Hanebeck
"Shape Tracking of Extended Objects and Group Targets with Star-Convex RHMs,"
in *Proceedings of the 14th International Conference on Information Fusion (Fusion 2011)*, Chicago, USA, Jul. 2011, pp. 1–8.

[6] Marcus Baum, Florian Faion, and Uwe D. Hanebeck
"Modeling the Target Extent with Multiplicative Noise,"
in *Proceedings of the 15th International Conference on Information Fusion (Fusion 2012)*, Singapore, Jul. 2012, pp. 2406–2412.

[7] Licong Zhang, Jürgen Sturm, Daniel Cremers, and Dongheui Lee
"Real-Time Human Motion Tracking Using Multiple Depth Cameras,"
in *Proceedings of the 2012 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2012)*, Vilamoura, Portugal, Oct. 2012, pp. 2389–2395.

[8] Marcus Baum and Uwe D. Hanebeck
"Fitting Conics to Noisy Data Using Stochastic Linearization,"
in *Proceedings of the 2011 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2011)*, San Francisco, USA, Sep. 2011, pp. 2050–2055.

[9] Antonio Zea, Florian Faion, Marcus Baum, and Uwe D. Hanebeck
"Level-Set Random Hypersurface Models for Tracking Non-Convex Extended Objects,"
in *Proceedings of the 16th International Conference on Information Fusion (Fusion 2013)*, Istanbul, Turkey, Jul. 2013.

[10] Sebastian Thrun, Wolfram Burgard, and Dieter Fox
Probabilistic Robotics.
Cambridge, London: MIT Press, 2005.

[11] Florian Faion, Patrick Ruoff, Antonio Zea, and Uwe D. Hanebeck
"Recursive Bayesian Calibration of Depth Sensors with Non-Overlapping Views,"
in *Proceedings of the 15th International Conference on Information Fusion (Fusion 2012)*, Singapore, Jul. 2012, pp. 757–762.

[12] Arnaud Doucet and Adam M. Johansen
"A Tutorial on Particle Filtering and Smoothing: Fifteen Years Later,"
in *Oxford Handbook of Nonlinear Filtering*, 2011, pp. 656–704.

[13] Sanjeev Arulampalam, Simon Maskell, Neil Gordon, and Tim Clapp
"A Tutorial on Particle Filters for Online Nonlinear/Non-Gaussian Bayesian Tracking,"
IEEE Transactions on Signal Processing, vol. 50, no. 2, pp. 174–188, Feb. 2002.

[14] Jayesh H. Kotecha and Petar M. Djuric
"Gaussian Particle Filtering,"
IEEE Transactions on Signal Processing, vol. 51, no. 10, pp. 2592–2601, Oct. 2003.

[15] Jannik Steinbring and Uwe D. Hanebeck
"Progressive Gaussian Filtering Using Explicit Likelihoods,"
in *Proceedings of the 17th International Conference on Information Fusion (Fusion 2014)*, Salamanca, Spain, Jul. 2014.

[16] Uwe D. Hanebeck
"PGF 42: Progressive Gaussian Filtering with a Twist,"
in *Proceedings of the 16th International Conference on Information Fusion (Fusion 2013)*, Istanbul, Turkey, Jul. 2013.

[17] Rudolf E. Kalman
"A New Approach to Linear Filtering and Prediction Problems,"
in *Transaction of the ASME—Journal of Basic Engineering*, Mar. 1960, pp. 35–45.

[18] Dan Simon
Optimal State Estimation,
1st ed. Wiley & Sons, 2006.

[19] X. Rong Li and Yu Liu
"Generalized Linear Minimum Mean-Square Error Estimation,"
in *Proceedings of the 16th International Conference on Information Fusion (Fusion 2013)*, Istanbul, Turkey, Jul. 2013, pp. 1819–1826.

[20] Tor Steinar Schei
"A Finite-Difference Method for Linearization in Nonlinear Estimation Algorithms,"
Automatica, vol. 33, no. 11, pp. 2053–2058, 1997.

[21] Rudolph van der Merwe
"Sigma-Point Kalman Filters for Probabilistic Inference in Dynamic State-Space Models,"
Ph.D. dissertation, OGI School of Science & Engineering, Oregon Health & Science University, Portland, Apr. 2004.

[22] Marco F. Huber, Frederik Beutler, and Uwe D. Hanebeck
"Semi-Analytic Gaussian Assumed Density Filter,"
in *Proceedings of the 2011 American Control Conference (ACC 2011)*, San Francisco, USA, Jun. 2011.

[23] Tine Lefebvre, Herman Bruyninckx, and Joris De Schutter
"Kalman Filters for Non-Linear Systems: A Comparison of Performance,"
International Journal of Control, vol. 77, no. 7, pp. 639–653, May 2004.

[24] ———
"Appendix A: The Linear Regression Kalman Filter,"
in *Nonlinear Kalman Filtering for Force-Controlled Robot Tasks*, ser. Springer Tracts in Advanced Robotics. Berlin Heidelberg: Springer, 2005, vol. 19.

[25] Marco F. Huber, Frederik Beutler, and Uwe D. Hanebeck
"(Semi-)Analytic Gaussian Mixture Filter,"
in *Proceedings of the 18th IFAC World Congress (IFAC 2011)*, Milano, Italy, Aug. 2011.

[26] Simon J. Julier and Jeffrey K. Uhlmann
"A New Extension of the Kalman Filter to Nonlinear Systems,"
in *11th Int. Symp. Aerospace/Defense Sensing, Simulation and Controls*, 1997, pp. 182–193.

[27] ———
"Unscented Filtering and Nonlinear Estimation,"
in *Proceedings of the IEEE*, vol. 92, Mar. 2004, pp. 401–422.

[28] Ryan Turner and Carl Edward Rasmussen
"Model Based Learning of Sigma Points in Unscented Kalman Filtering,"
Neurocomputing, vol. 80, pp. 47–53, 2012.

[29] Jindrich Dunik, Miroslav Simandl, and Ondrej Straka
"Unscented Kalman Filter: Aspects and Adaptive Setting of Scaling Parameter,"
IEEE Transactions on Automatic Control, vol. 57, no. 9, pp. 2411–2416, Sep. 2012.

[30] Marco F. Huber and Uwe D. Hanebeck
"Gaussian Filter Based on Deterministic Sampling for High Quality Nonlinear Estimation,"
in *Proceedings of the 17th IFAC World Congress (IFAC 2008)*, vol. 17, Seoul, Republic of Korea, Jul. 2008.

[31] Jindrich Dunik, Ondrej Straka, and Miroslav Simandl
"The Development of a Randomised Unscented Kalman Filter,"
in *Proceedings of the 18th IFAC World Congress*, Milano, Italy, Aug. 2011, pp. 8–13.

[32] Kazufumi Ito and Kaiqi Xiong
"Gaussian Filters for Nonlinear Filtering Problems,"
IEEE Transactions on Automatic Control, vol. 45, no. 5, pp. 910–927, May 2000.

[33] Magnus Nørgaard, Niels K. Poulsen, and Ole Ravn
"New Developments in State Estimation for Nonlinear Systems,"
Automatica, vol. 36, no. 11, pp. 1627–1638, 2000.

[34] Marco F. Huber
"Chebyshev Polynomial Kalman Filter,"
Digital Signal Processing, vol. 23, no. 5, pp. 1620–1629, Sep. 2013.

[35] Michael Roth and Fredrik Gustafsson
"An Efficient Implementation of the Second Order Extended Kalman Filter,"
in *Proceedings of the 14th International Conference on Information Fusion (Fusion 2011)*, Jul. 2011.

[36] Oliver C. Schrempf and Uwe D. Hanebeck
"Dirac Mixture Approximation for Nonlinear Stochastic Filtering,"
in *Informatics in Control, Automation and Robotics—Selected Papers from the International Conference on Informatics in Control, Automation and Robotics 2007*, ser. Lecture Notes in Electrical Engineering. Springer, Sep. 2008, vol. 24, pp. 287–300.

[37] Uwe D. Hanebeck, Marco F. Huber, and Vesa Klumpp
"Dirac Mixture Approximation of Multivariate Gaussian Densities,"
in *Proceedings of the 2009 IEEE Conference on Decision and Control (CDC 2009)*, Shanghai, China, Dec. 2009.

[38] Igor Gilitschenski and Uwe D. Hanebeck
"Efficient Deterministic Dirac Mixture Approximation of Gaussian Distributions,"
in *Proceedings of the 2013 American Control Conference (ACC 2013)*, Washington D.C., USA, Jun. 2013.

[39] Wolfgang Härdle and Léopold Simar
Applied Multivariate Statistical Analysis,
2nd ed. Berlin Heidelberg: Springer, 2008.

[40] Jorge Nocedal
"Updating Quasi-Newton Matrices with Limited Storage,"
Mathematics of Computation, vol. 35, no. 151, pp. 773–782, Jul. 1980.

[41] Dong C. Liu and Jorge Nocedal
"On the Limited Memory BFGS Method for Large Scale Optimization,"
Mathematical Programming, vol. 45, no. 1–3, pp. 503–528, Aug. 1989.

[42] Uwe D. Hanebeck and Oliver C. Schrempf
"Greedy Algorithms for Dirac Mixture Approximation of Arbitrary Probability Density Functions,"
in *Proceedings of the 2007 IEEE Conference on Decision and Control (CDC 2007)*, New Orleans, USA, Dec. 2007, pp. 3065–3071.

[43] Bruno O. S. Teixeira, Leonardo A. B. Tôrres, Luis A. Aguirre, and Dennis S. Bernstein
"On Unscented Kalman Filtering with State Interval Constraints,"
Journal of Process Control, vol. 20, no. 1, pp. 45–57, 2010.

[44] Raymond Kan
"From Moments of Sum to Moments of Product,"
Journal of Multivariate Analysis, vol. 99, no. 3, pp. 542–554, Mar. 2008.

[45] Ondrej Straka, Jindrich Dunik, and Miroslav Simandl
"Gaussian Sum Unscented Kalman Filter with Adaptive Scaling Parameters,"
in *Proceedings of the 14th International Conference on Information Fusion (Fusion 2011)*, Chicago, USA, Jul. 2011, pp. 1–8.

[46] Rudolph van der Merwe and Eric A. Wan
"The Square-Root Unscented Kalman Filter for State and Parameter-Estimation,"
in *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '01)*, vol. 6, Salt Lake City, USA, May 2001, pp. 3461–3464.

Jannik Steinbring received his Dipl.-Inform. in computer science from the Karlsruhe Institute of Technology (KIT), Germany, in 2012. Currently, he is working towards a Ph.D. degree at the Intelligent Sensor-Actuator-Systems Laboratory, Karlsruhe Institute of Technology (KIT), Germany. His research interests are in the fields of nonlinear state estimation and extended object tracking.

Uwe D. Hanebeck is a chaired professor of Computer Science at the Karlsruhe Institute of Technology (KIT) in Germany and director of the Intelligent Sensor-Actuator-Systems Laboratory (ISAS). Since 2005, he is the chairman of the Research Training Group RTG 1194 "Self-Organizing Sensor-Actuator-Networks" financed by the German Research Foundation. Prof. Hanebeck obtained his Ph.D. degree in 1997 and his habilitation degree in 2003, both in Electrical Engineering from the Technical University in Munich, Germany. His research interests are in the areas of information fusion, nonlinear state estimation, stochastic modeling, system identification, and control with a strong emphasis on theory-driven approaches based on stochastic system theory and uncertainty models. Research results are applied to various application topics like localization, human-robot-interaction, assistive systems, sensor-actuator-networks, medical engineering, distributed measuring system, and extended range telepresence. Research is pursued in many academic projects and in a variety of cooperations with industrial partners.

Uwe D. Hanebeck was the General Chair of the "2006 IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI 2006)," Program Co-Chair of the "11th International Conference on Information Fusion (Fusion 2008)," Program Co-Chair of the "2008 IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI 2008)," Regional Program Co-Chair for Europe for the "2010 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2010)," and will be General Chair of the "19th International Conference on Information Fusion (Fusion 2016)." He is a Member of the Board of Directors of the International Society of Information Fusion (ISIF), Editor-in-chief of its *Journal of Advances in Information Fusion (JAIF)*, and associate editor for the letter category of the *IEEE Transactions on Aerospace and Electronic Systems (TAES)*. He is author and coauthor of more than 300 publications in various high-ranking journals and conferences.