Moment Estimation Using a Marginalized Transform

Moment Estimation Using a Marginalized Transform

Fredrik Sandblom, Lennart Svensson, *Senior Member, IEEE*

***Abstract*—We present a method for estimating mean and covariance of a transformed Gaussian random variable. The method is based on evaluations of the transforming function and resembles the unscented transform and Gauss-Hermite integration in that respect. The information provided by the evaluations is used in a Bayesian framework to form a posterior description of the parameters in a model of the transforming function. Estimates are then derived by marginalizing these parameters from the analytical expression of the mean and covariance. An estimation algorithm, based on the assumption that the transforming function can be described using Hermite polynomials, is presented and applied to the non-linear filtering problem. The resulting marginalized transform (MT) estimator is compared to the cubature rule, the unscented transform and the divided difference estimator. The evaluations show that the presented method performs better than these methods, more specifically in estimating the covariance matrix. Contrary to the unscented transform, the resulting approximation of the covariance matrix is guaranteed to be positive-semidefinite.**

***Index Terms*—Filtering theory, statistical linearization, recursive estimation, Bayes methods, Kalman filter, Tracking.**

I. Introduction

Calculating the mean and covariance of stochastic variables is central to many estimation tasks, including, e.g., sensitivity analysis, which can be applied to a variety of systems including antenna characterization [1], power system analysis [2] and circuit design [3]. It is also frequently an essential component in recursive state estimation where the posterior mean and covariance often are used to characterize the distribution [4], [5]. The importance of this task, with applications ranging from surveillance to medicine, have motivated a large part of recent research within the area of moment estimation [5], [6], [7], [8], [9], [10].

The general Bayesian solution to the state estimation problem involves integration of probability density functions — integrals which are rarely mathematically tractable. The family of *Gaussian filters* solves the recursive estimation problem under the assumption that the concerned distributions are approximately Gaussian. The equations used to compute the posterior mean and covariance under this assumption are those of the linear minimum mean square error (LMMSE) estimator, which coincides with the well known Kalman filter for linear systems [4].

A variety of Gaussian filters have been proposed to cope with non-linear models [5], and the derivative-free filters [11], [6], [7], [8], [9] are particularly useful; with little or no adjustment, they can be applied to a wide range of problems. These filters use a transformed set of deterministically chosen points, often referred to as *sigma-points*, to approximate the mean and covariance. Arguably, the most well-known sigma-point method is the unscented transform (UT) [7], [11], that has been shown [12] to realize the fully symmetric integration formula presented in [13], which is exact for integration over third order polynomial functions. The (second order) divided difference filter (DD2) [6] calculates the mean and covariance matrix jointly, and both estimates are exact for a certain family of second order polynomials. An extensive analysis of the numerical integration perspective on Gaussian filters is given in [14].

Although easy to apply, derivative-free filters are not problem-free. The UT covariance matrix estimate is sometimes calculated such that it is not necessarily positive-semidefinite. This behavior was overcome with the recent introduction of the cubature integration rule [9], a special case of the UT, whose covariance matrix estimates are guaranteed to be non-negative definite. It performs well compared to methods of similar complexity [9], [15], [16], but unfortunately, the robustness comes at the expense of using a less accurate integration rule. Furthermore, similar to the UT, the mean and covariance are computed independently, which implies two different assumptions on the underlying mapping within the same method.

In this paper the transforming function is approximated with a linear combination of Hermite polynomials, for which closed-form expressions for the mean and covariance are well known. The polynomial coefficients are given a hierarchical prior, and the posterior distribution of these coefficients is computed conditioned on the transformed sigma-points. The desired mean and covariance can then be calculated by marginalizing the influence of the coefficients from the analytical expressions. The approximation of the function as a linear combination of Hermite polynomials, with unknown parameters, is the only approximate step in these calculations. Similar approaches have been suggested in [17] and [18], albeit using a non-parametric Gaussian process as a model of the transformation. A Bayesian approach towards learning such a process through evaluations was presented already in [19].

There are several reasons to derive sigma-point algorithms

F. Sandblom is with the department of Safety Functions & Electronics at Volvo Group Trucks Technology, SE-405 08 Gothenburg, Sweden, E-mail: fredrik.sandblom.2@volvo.com

L. Svensson and F. Sandblom are with the Department of Signals and Systems, Chalmers University of Technology, SE-412 96 Gothenburg, Sweden, E-mail: {lennart.svensson, fredrik.sandblom}@chalmers.se. They were supported by the Strategic Vehicle Research and Innovation Program (FFI), which is funded by the Swedish Governmental Agency for Innovation Systems (VINNOVA).

using Bayesian techniques. First, the mean and the covariance matrix estimates are calculated jointly, based on analytical expressions rather than a numerical approach. Hence, it is possible to guarantee a positive-semidefinite covariance matrix. Second, the model assumptions become clearly visible through the prior distribution, making it easier to understand the algorithm. Third, Bayesian methods are generally well performing in the sense that they are admissible under relatively loose assumptions [20] and that they are optimal when the performance is averaged over the prior. Finally, we know that the key to improve performance is the choice of the prior. Although the design of a prior can be difficult, we believe the choice is better made explicitly than implicitly. To illustrate this, we present a family of priors that result in the cubature, UT, and DD2 estimators, for certain choices of the prior. It is shown that the presented algorithm can provide very good estimates of the mean and covariance, and that the estimation error of the recursive filter is more accurately described using the proposed method. More specifically, we appear to provide more robust covariance estimates, when the underlying polynomials are not completely linear.

The paper is organized as follows. Section II describes the estimation task at hand and a summary of the sigma-point approach. The proposed marginalization technique is introduced in Section III, and is applied to Hermite polynomials in Section IV. Closed-form expressions for mean and covariance are derived in Section V together with a summary of the algorithm. Analytical results and a clarification of the relationship to other sigma-point methods are discussed in Section VI. Usage of the technique in a Kalman filter framework is demonstrated in Section VII, and estimation and tracking performance is evaluated in Section VIII. Our conclusions are listed in Section IX. Finally, Appendices A–C provide results regarding the positive-definiteness of the UT covariance matrix, properties of Hermite polynomials, and an interpretation of the sigma-point selection scheme.

II. Problem formulation

Consider a stochastic variable $\mathbf{x} \in \mathbb{R}^n$ with probability density function

$$\mathbf{x} \sim \mathcal{N}(\boldsymbol{\mu}_\mathbf{x}, \boldsymbol{P}_\mathbf{x}),$$

where $\boldsymbol{\mu}_\mathbf{x}$ and $\boldsymbol{P}_\mathbf{x}$ are known. We wish to calculate the mean and covariance of the variable $\mathbf{y} \in \mathbb{R}^m$:

$$\mathbf{y} = g(\mathbf{x}),$$

where $g : \mathbb{R}^n \to \mathbb{R}^m$ is a known transformation. The desired moments are given by the integral expressions

$$\mathbb{E}[\mathbf{y}] = \int_{\mathbb{R}^n} \mathcal{N}(\mathbf{x}\,; \boldsymbol{\mu}_\mathbf{x}, \boldsymbol{P}_\mathbf{x}) g(\mathbf{x}) d\mathbf{x} \tag{1}$$

$$\begin{aligned}\mathrm{Cov}(\mathbf{y}) &= \int_{\mathbb{R}^n} \mathcal{N}(\mathbf{x}; \boldsymbol{\mu}_\mathbf{x}, \boldsymbol{P}_\mathbf{x}) \big[g(\mathbf{x}) - \mathbb{E}[\mathbf{y}]\big] \big[g(\mathbf{x}) - \mathbb{E}[\mathbf{y}]\big]^T d\mathbf{x} \\ &= \int_{\mathbb{R}^n} \mathcal{N}(\mathbf{x}; \boldsymbol{\mu}_\mathbf{x}, \boldsymbol{P}_\mathbf{x}) g(\mathbf{x}) g(\mathbf{x})^T d\mathbf{x} - \mathbb{E}[\mathbf{y}]\mathbb{E}[\mathbf{y}]^T. \end{aligned} \tag{2}$$

Expressing the solutions to these integrals on a closed form is often impossible for transformations encountered in practice. Sigma-point methods provide approximate solutions to these integrals, and have demonstrated nice properties with respect to performance and simplicity. The question at hand is therefore how to use the sigma-points as efficiently as possible.

A. Summary of the sigma-point approach to statistical moment calculations

The family of sigma-point filters approximate integrals (1)–(2) using a weighted sum:

$$\int_{\mathbb{R}^n} \mathcal{N}(\mathbf{x}\,; \boldsymbol{\mu}_\mathbf{x}, \boldsymbol{P}_\mathbf{x}) g(\mathbf{x}) d\mathbf{x} \approx \sum_{i=0}^{2n} w_i g(\mathbf{x}^i). \tag{3}$$

The so-called sigma-points, $\{\mathbf{x}^0, \ldots, \mathbf{x}^{2n}\}$, and the associated weights, w_i, are chosen according to a deterministic scheme. For the unscented transform, they are:

$$\mathbf{x}^0 = \mathbb{E}[\mathbf{x}] \tag{4}$$

$$\mathbf{x}^i = \begin{cases} \mathbb{E}[\mathbf{x}] + \left(\sqrt{\frac{n}{(1-w_0)}\boldsymbol{P}_\mathbf{x}}\right)_i, & 1 \le i \le n \\ \mathbb{E}[\mathbf{x}] - \left(\sqrt{\frac{n}{(1-w_0)}\boldsymbol{P}_\mathbf{x}}\right)_{i-2n/2}, & n < i \le 2n \end{cases} \tag{5}$$

$$w_i = \frac{1-w_0}{2n}, \tag{6}$$

where $i = 1, \ldots, 2n$ and $\left(\sqrt{\boldsymbol{P}_\mathbf{x}}\right)_i$ is the i^{th} column of a matrix square root such that $\sqrt{\boldsymbol{P}_\mathbf{x}}\sqrt{\boldsymbol{P}_\mathbf{x}}^T = \boldsymbol{P}_\mathbf{x}$. When $\mathbf{x}$ is Gaussian, the suggested setting for the UT [7] is to use $w_0 = 1 - n/3$, whereas the cubature rule is obtained by setting $w_0 = 0$, effectively removing $\mathbf{x}^0$ from the set of sigma-points. This integral approximation strategy, applied to equation (1), yields the estimator

$$\mathbb{E}[g(\mathbf{x})] \approx \sum_{i=0}^{2n} w_i g(\mathbf{x}^i) \triangleq \bar{\mathbf{y}}. \tag{7}$$

The covariance matrix estimate, $\hat{\boldsymbol{P}}_\mathbf{y}$, is usually expressed in terms of the weighted sum of squares, but we prefer to view it on the form (2) to clarify that the integral approximation strategy is applied twice:

$$\begin{aligned}\mathrm{Cov}(\mathbf{y}) &\approx \sum_{i=0}^{2n} w_i [g(\mathbf{x}^i) - \bar{\mathbf{y}}][g(\mathbf{x}^i) - \bar{\mathbf{y}}]^T \\ &= \sum_{i=0}^{2n} w_i g(\mathbf{x}^i) g(\mathbf{x}^i)^T - \sum_{i=0}^{2n} w_i g(\mathbf{x}^i) \bar{\mathbf{y}}^T \\ &\quad - \bar{\mathbf{y}} \sum_{i=0}^{2n} w_i g(\mathbf{x}^i)^T + \sum_{i=0}^{2n} w_i \bar{\mathbf{y}} \bar{\mathbf{y}}^T \\ &= \sum_{i=0}^{2n} w_i g(\mathbf{x}^i) g(\mathbf{x}^i)^T - \bar{\mathbf{y}}\bar{\mathbf{y}}^T \triangleq \hat{\boldsymbol{P}}_\mathbf{y}. \end{aligned} \tag{8}$$

Since $g(\mathbf{x})g(\mathbf{x})^T$ generally has higher polynomial order than $g(\mathbf{x})$, a strategy which calculates the mean (7) accurately need not be appropriate for the covariance matrix (8). In fact, with

negative weights it may not even be positive-semidefinite; see proof in Appendix A.

The DD2 estimator uses the same sigma-point selection scheme (5), but is parameterized using a scalar $h^2 = \frac{n}{1-w_0}$. The sigma points and the weights used to calculate the mean are identical to those of the UT. The covariance matrix approximation, however, employs a different set of weights which are positive regardless of the dimensionality.

III. Proposed idea

Even though the transforming function g is known, we model it as a stochastic process with a prior distribution $\pi(g)$. Apart from the prior, the only available information is the evaluated points, $\chi = \left[\mathbf{x}^0, \ldots, \mathbf{x}^{2n}\right]$, and the function values at these points, $\mathbf{z} = \left[g(\mathbf{x}^0), \ldots, g(\mathbf{x}^{2n})\right]$. Using estimation terminology: χ and $\mathbf{z}$ are our measurements, the function g is a nuisance parameter with posterior distribution $p(g|\mathbf{z}, \chi)$ and our objective is to estimate the mean, $\bar{\mathbf{y}}_\pi$, and covariance, $\boldsymbol{P}_{\mathbf{y},\pi}$, of $\mathbf{y}$.

The mean, expressed as a functional of the transformation, g, is denoted by

$$\bar{\mathbf{y}}(g) \triangleq \int \mathcal{N}(\mathbf{x}\,; \boldsymbol{\mu}_\mathbf{x}, \boldsymbol{P}_\mathbf{x}) g(\mathbf{x}) d\mathbf{x}, \tag{9}$$

and the corresponding covariance matrix by

$$P_\mathbf{y}(g) \triangleq \int \mathcal{N}(\mathbf{x}\,; \boldsymbol{\mu}_\mathbf{x}, \boldsymbol{P}_\mathbf{x}) [g(\mathbf{x}) - \bar{\mathbf{y}}(g)][g(\mathbf{x}) - \bar{\mathbf{y}}(g)]^T d\mathbf{x}. \tag{10}$$

The expressions for the desired mean and covariance of $\mathbf{y}$, given $\mathbf{z}$ and χ, are given by marginalization over g:

$$\begin{aligned} \bar{\mathbf{y}}_\pi &= \mathbb{E}[\mathbf{y}|\mathbf{z}, \chi] \\ &= \int \bar{\mathbf{y}}(g) p(g|\mathbf{z}, \chi) dg \end{aligned} \tag{11}$$

$$\begin{aligned} \boldsymbol{P}_{\mathbf{y},\pi} &= \mathbb{E}\left[\boldsymbol{P}_\mathbf{y}(g) \,\middle|\, \mathbf{z}, \chi\right] \\ &= \int \boldsymbol{P}_\mathbf{y}(g) p(g|\mathbf{z}, \chi) dg. \end{aligned} \tag{12}$$

The idea is to use a prior $\pi(g)$ for which the integrals in (11) and (12) have closed-form solutions. Although it is possible to find solutions for infinite-dimensional integrals, it is more practical to consider a finite parameterization of g. In this paper we focus on one such prior, presented in Section IV, where g is assumed to belong to the family of Hermite polynomials. An interpretation of using this prior is that the mean (11) and covariance (12) are averaged over polynomials that pass through the points $(\mathbf{x}^i, g(\mathbf{x}^i))$, for all integers $i \in [0, 2n]$, as illustrated in Fig. 1.

IV. Using a Hermite polynomial to model the transforming function

Hermite polynomials are used to model g for three main reasons. First, using *polynomials* facilitate comparisons with other sigma-point methods, which calculate (7) exactly for certain polynomials. Second, *Hermite polynomials* yield particularly simple expressions when $p(x)$ is Gaussian, and third,

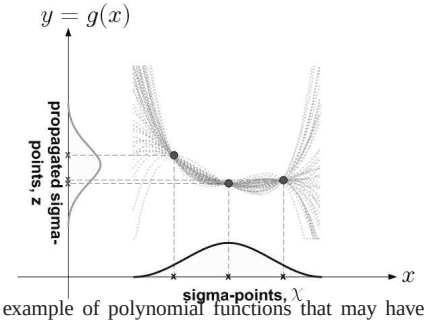

Fig. 1. An example of polynomial functions that may have performed the transformation of the sigma-points. The mean and covariance of $y = g(x)$ is calculated by a weighted average of all such functions.

polynomials are well known for their ability to approximate arbitrary continuous functions [21].

To illustrate some fundamental properties, we study a scalar transformation. Any function g, for which $\mathbb{E}[g(x)^2] < \infty$, can be expressed in terms of a series of weighted Hermite polynomials [10]:

$$g(x) = \sum_{k=0}^{\infty} \frac{1}{k!} \mathbb{E}[g(x) H_k(x)] H_k(x), \tag{13}$$

for $x \sim \mathcal{N}(0, 1)$. To have a tractable solution, we assume that the transforming function can be approximated using a finite series, fully described by a weight vector $\boldsymbol{\theta} = [\theta_0, \ldots, \theta_p]^T$:

$$g(x) \approx \sum_{k=0}^{p} \theta_k H_k(x). \tag{14}$$

The k^{th} order hermite polynomial, H_k, is given by (86) in Appendix B, which contains a summary of useful properties of Hermite polynomials. For instance, using Hermite polynomials leads to very simple expressions for the mean, $\bar{y}(g)$, and covariance, $P_y(g)$, or as they now can be expressed, $\bar{y}(\boldsymbol{\theta})$ and $\sigma_y^2(\boldsymbol{\theta})$:

$$\begin{aligned} \bar{y}(\boldsymbol{\theta}) &= \theta_0 \\ \sigma_y^2(\boldsymbol{\theta}) &= \sum_{k=1}^{p} \theta_k^2 k! \,. \end{aligned}$$

For example, if $y = x + x^2$ and $x \sim \mathcal{N}(0, 1)$, then $y = H_0 + H_1 + H_2$, (i.e., $\boldsymbol{\theta} = [1\,1\,1]^T$). Consequently, the expected value is $\theta_0 = 1$ and the variance is $\theta_1^2 + \theta_2^2 2! = 3$.

A. Multidimensional transformation

A transformation $g : \mathbb{R}^n \to \mathbb{R}^m$, performed by a linear combination of base functions can be written as

$$g(\mathbf{x}; \boldsymbol{\theta}) = \boldsymbol{\theta}^T \mathbf{h}(\mathbf{x}), \tag{15}$$

where the base functions enter the equation through

$$\begin{aligned} \mathbf{h}(\mathbf{x}) = [&H_0,\ H_1(x_1), \ldots, H_p(x_1),\ H_1(x_2), \\ &\ldots, H_p(x_2), \ldots, H_1(x_n), \ldots, H_p(x_n)]^T. \end{aligned} \tag{16}$$

In the following sections we assume $\mathbf{x} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_{n \times n})$, a simplification justified in Section IV-D. We construct the weight matrix from the p-dimensional vectors $\boldsymbol{\theta}^{i,j}$, each describing

the transformation from x_i to y_j, and the scalars θ_0^j, for $i = 1 \dots n$ and $j = 1 \dots m$:

$$\boldsymbol{\theta} = \begin{bmatrix} \theta_0^1 & \dots & \theta_0^j & \dots & \theta_0^m \\ \boldsymbol{\theta}^{1,1} & \dots & \boldsymbol{\theta}^{1,j} & \dots & \boldsymbol{\theta}^{1,m} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ \boldsymbol{\theta}^{n,1} & \dots & \boldsymbol{\theta}^{n,j} & \dots & \boldsymbol{\theta}^{n,m} \end{bmatrix}. \tag{17}$$

Consequently, $\boldsymbol{\theta}^j$, the j^{th} column of $\boldsymbol{\theta}$, defines the mapping from $\mathbf{x} \in \mathbb{R}^n$ to y_j over the base functions in $\mathbf{h}(\mathbf{x})$:

$$y_j = (\boldsymbol{\theta}^j)^T \mathbf{h}(\mathbf{x}). \tag{18}$$

The function g is completely described by $\boldsymbol{\theta}$ through equation (15), and we turn our attention to the expressions for $\bar{\mathbf{y}}(\boldsymbol{\theta})$ and $\boldsymbol{P}_{\mathbf{y}}(\boldsymbol{\theta})$. For a given polynomial, i.e., one realization of $\boldsymbol{\theta}$, $\mathbf{y}$ has the mean

$$\begin{aligned} \bar{\mathbf{y}}(\boldsymbol{\theta}) &= \mathbb{E}[\boldsymbol{\theta}^T \mathbf{h}(\mathbf{x}) | \boldsymbol{\theta}] \\ &= \left[\theta_0^1, \dots, \theta_0^m\right]^T, \end{aligned} \tag{19}$$

where $\mathbb{E}[\mathbf{h}(\mathbf{x})]$ is given by equation (84). To simplify notation, we introduce the vector

$$\mathbf{w} \triangleq \mathbb{E}[\mathbf{h}(\mathbf{x})] = [1,\ 0, \dots, 0]^T, \tag{20}$$

and write the covariance matrix for $\mathbf{y}$:

$$\begin{aligned} \boldsymbol{P}_{\mathbf{y}}(\boldsymbol{\theta}) &= \mathbb{E}\left[[\boldsymbol{\theta}^T \mathbf{h}(\mathbf{x}) - \boldsymbol{\theta}^T \mathbf{w}][\boldsymbol{\theta}^T \mathbf{h}(\mathbf{x}) - \boldsymbol{\theta}^T \mathbf{w}]^T | \boldsymbol{\theta} \right] \\ &= \boldsymbol{\theta}^T \mathbb{E}\left[[\mathbf{h}(\mathbf{x}) - \mathbf{w}][\mathbf{h}(\mathbf{x}) - \mathbf{w}]^T \right] \boldsymbol{\theta} \\ &= \boldsymbol{\theta}^T \mathbf{C} \boldsymbol{\theta}. \end{aligned} \tag{21}$$

All off-diagonal elements of $\mathbf{C} \triangleq \mathbb{E}[[\mathbf{h}(\mathbf{x}) - \mathbf{w}][\mathbf{h}(\mathbf{x}) - \mathbf{w}]^T]$ are zero, and the $pn + 1$ diagonal elements are:

$$\text{diag}(\mathbf{C}) = [0,\ 1!,\ 2!, \dots,\ p!, \dots,\ 1!,\ 2!, \dots,\ p!]^T, \tag{22}$$

see equations (84) and (85) in Appendix B. The relation between the mean (19) and covariance (21) of $\mathbf{y}$, and the parameter vector $\boldsymbol{\theta}$, is now clear. Before we attempt to marginalize $\boldsymbol{\theta}$ from these expressions, we attend to the prior.

B. Designing the prior distribution

Using Hermitian polynomials, designing the prior $\pi(g)$ is now equivalent to designing $\pi(\boldsymbol{\theta})$, and there is an intuitive interpretation: the number of elements in $\boldsymbol{\theta}$ determines the maximum order of the transforming polynomial. Similarly, the variance determines which coefficients are updated with the information provided in the propagated sigma points.

The proposed prior assumes the vectors $\boldsymbol{\theta}^{i,j}$ to be independently generated from a hierarchical model:

$$\boldsymbol{\theta}^{i,j} \sim \mathcal{N}(0, \alpha_j \boldsymbol{P}_\theta^{i,j}). \tag{23}$$

It is shown in Section VI-B that the sigma-points can be selected such that the prior on θ_0 does not affect the posterior distribution, $p(\boldsymbol{\theta}|\mathbf{z}, \chi)$, but for completeness let it be assumed that all scalars θ_0^j are independently drawn from $\mathcal{N}(0, \sigma_{\theta_0}^2)$. The covariance matrix $\text{Cov}(\boldsymbol{\theta}^j) = \alpha_j \boldsymbol{P}_\theta^j$ is therefore block-diagonal, with:

$$\boldsymbol{P}_\theta^j = \begin{bmatrix} \sigma_{\theta_0}^2/\alpha_j & \mathbf{0} & \dots & \mathbf{0} \\ \mathbf{0} & \boldsymbol{P}_\theta^{1,j} & \ddots & \vdots \\ \vdots & \ddots & \ddots & \mathbf{0} \\ \mathbf{0} & \dots & \mathbf{0} & \boldsymbol{P}_\theta^{n,j} \end{bmatrix}. \tag{24}$$

Note that the hyperparameter, α_j, is common for all the parameters $\theta^{1,j}, \dots, \theta^{n,j}$, in order to share information about the scale of the problem across dimensions. Techniques for estimating α_j are discussed in Section V-B.

C. Estimates of mean and covariance

Expressions (19) and (21) are derived for a given weight matrix, $\boldsymbol{\theta}$. However, since $\boldsymbol{\theta}$ is modeled as a stochastic variable, the marginalization in (11) and (12) gives the final estimators:

$$\begin{aligned} \bar{\mathbf{y}}_\pi &= \mathbb{E}[\boldsymbol{\theta}^T | \mathbf{z}, \chi] \mathbf{w} \\ &= \boldsymbol{\mu}_{\theta|\mathbf{z}}^T \mathbf{w} \\ &= \mathbb{E}\left[[\theta_0^1, \dots, \theta_0^m]^T | \mathbf{z}, \chi \right] \end{aligned} \tag{25}$$

$$\begin{aligned} \boldsymbol{P}_{\mathbf{y},\pi} &= \mathbb{E}[\boldsymbol{\theta}^T \mathbf{C}\, \boldsymbol{\theta} | \mathbf{z}, \chi] \\ &= \boldsymbol{\mu}_{\theta|\mathbf{z}}^T \mathbf{C}\, \boldsymbol{\mu}_{\theta|\mathbf{z}} + \mathbb{E}\left[[\boldsymbol{\theta} - \boldsymbol{\mu}_{\theta|\mathbf{z}}]^T \mathbf{C} [\boldsymbol{\theta} - \boldsymbol{\mu}_{\theta|\mathbf{z}}] | \mathbf{z}, \chi \right] \\ &= \boldsymbol{\mu}_{\theta|\mathbf{z}}^T \mathbf{C}\, \boldsymbol{\mu}_{\theta|\mathbf{z}} + \begin{bmatrix} \alpha_1 \text{Tr}\left\{ \boldsymbol{P}_{\theta|\mathbf{z}}^1 \mathbf{C} \right\} & & 0 \\ & \ddots & \\ 0 & & \alpha_m \text{Tr}\left\{ \boldsymbol{P}_{\theta|\mathbf{z}}^m \mathbf{C} \right\} \end{bmatrix}, \end{aligned} \tag{26}$$

where we introduce the notation $\boldsymbol{\mu}_{\theta|\mathbf{z}}$ for the conditional mean, $\mathbb{E}[\boldsymbol{\theta}^T | \mathbf{z}, \chi]$, and $\boldsymbol{P}_{\theta|\mathbf{z}}^j$ ($j = 1 \dots m$) for the conditional posterior covariance. Expressions for $\boldsymbol{\mu}_{\theta|\mathbf{z}}$ and $\boldsymbol{P}_{\theta|\mathbf{z}}^j$ given observations $\mathbf{z}, \chi$ are derived in Section V.

D. Stochastic decoupling

The simple forms for $\mathbf{w}$ in (20) and $\mathbf{C}$ in (22) are expressed for vector arguments, $\mathbf{x}$, whose elements are uncorrelated with unit variance. Rather than expressing $\mathbf{w}$ and $\mathbf{C}$ for any mean and covariance of $\mathbf{x}$, a stochastic decoupling procedure similar to the approach in [6] is proposed, such that $\mathbf{w}$ and $\mathbf{C}$ are constant. Instead of studying

$$\mathbf{y} = g(\mathbf{x}), \quad \mathbf{x} \sim \mathcal{N}(\boldsymbol{\mu}_{\mathbf{x}}, \boldsymbol{P}_{\mathbf{x}}), \tag{27}$$

we introduce $\tilde{\mathbf{x}} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_{n \times n})$, where $\mathbf{I}_{n \times n}$ is the $n \times n$ identity matrix, and set

$$\mathbf{y} = \tilde{g}(\tilde{\mathbf{x}}) \triangleq g(\boldsymbol{\mu}_{\mathbf{x}} + \sqrt{\boldsymbol{P}_{\mathbf{x}}}\, \tilde{\mathbf{x}}), \tag{28}$$

which has the same distribution as the original $\mathbf{y}$ in (27). *Therefore, rather than recalculating* $\mathbf{w}$ *and* $\mathbf{C}$*, we assume the transformation is performed by* $\tilde{g}$ *in* (28)*. This adaptation is built in to the algorithm described in Section V-C.*

V. Calculating the posterior distribution

Our objective is now to calculate the posterior distribution $p(\boldsymbol{\theta}|\mathbf{z},\chi)$ and its first two moments, which are needed in the expressions for the mean and covariance of $\mathbf{y}$, given by equations (25)–(26). An exact expression of the distribution is obtained by marginalizing the hyperparameter, α, from the hierarchical model:

$$p(\boldsymbol{\theta}^j|\mathbf{z},\chi) = \int p(\boldsymbol{\theta}^j|\alpha_j,\mathbf{z},\chi)p(\alpha_j|\mathbf{z},\chi)d\alpha_j. \tag{29}$$

Finding a closed-form solution to (29) is usually difficult. A simple yet useful substitute is to use a point estimate of α_j. In other words, we set

$$p(\boldsymbol{\theta}^j|\mathbf{z},\chi) \approx p(\boldsymbol{\theta}^j|\hat{\alpha}_j,\mathbf{z},\chi). \tag{30}$$

In the following section, the first two moments of $p(\boldsymbol{\theta}^j|\hat{\alpha}_j,\mathbf{z},\chi)$ are calculated for a given estimate, $\hat{\alpha}_j$, which is then derived in Section V-B.

A. Mean and covariance of $\boldsymbol{\theta}$

The linear relation between observations $\mathbf{z}$ and parameter vector $\boldsymbol{\theta}$ was established in equation (15):

$$\mathbf{z} = \boldsymbol{\theta}^T\mathbf{H}^T(\chi), \tag{31}$$

where the observation matrix is given by:

$$\mathbf{H}(\chi) = \begin{bmatrix} \mathbf{h}^T(\mathbf{x}^0) \\ \vdots \\ \mathbf{h}^T(\mathbf{x}^{2n}) \end{bmatrix}. \tag{32}$$

For notational convenience, we omit the reference to χ from now on. Given a zero-mean Gaussian prior distribution on $\boldsymbol{\theta}^j$, with $\mathrm{Cov}(\boldsymbol{\theta}^j) = \alpha_j\boldsymbol{P}_\theta^j$, the posterior distribution is also Gaussian with mean and covariance [22]:

$$\boldsymbol{\mu}_{\theta|\mathbf{z}}^j = \boldsymbol{P}_\theta^j\mathbf{H}^T\left[\mathbf{H}\boldsymbol{P}_\theta^j\mathbf{H}^T\right]^{-1}\mathbf{z}^j \tag{33}$$

$$\alpha_j\boldsymbol{P}_{\theta|\mathbf{z}}^j = \left(\mathbf{I} - \boldsymbol{P}_\theta^j\mathbf{H}^T\left[\mathbf{H}\boldsymbol{P}_\theta^j\mathbf{H}^T\right]^{-1}\mathbf{H}\right)\alpha_j\boldsymbol{P}_\theta^j, \tag{34}$$

where $\mathbf{z}^j$ is the j^{th} column in $\mathbf{z}^T$. The conditional mean of $\boldsymbol{\theta}$ is $\boldsymbol{\mu}_{\theta|\mathbf{z}} = [\boldsymbol{\mu}_{\theta|\mathbf{z}}^1, \boldsymbol{\mu}_{\theta|\mathbf{z}}^2, \ldots, \boldsymbol{\mu}_{\theta|\mathbf{z}}^m]$. Estimates $\bar{\mathbf{y}}_\pi$ and $\boldsymbol{P}_{\mathbf{y},\pi}$ in (25) and (26) can thus readily be calculated.

If all transformations are treated the same way a priori, i.e., if the covariance matrices $\boldsymbol{P}_\theta^{i,j}$ in (23) do not depend on j, the elements $\mathrm{Tr}\{\boldsymbol{P}_\theta^j\mathbf{C}\}$ are also independent of j. Hence, the superscript j can be dropped and the expression for $\boldsymbol{P}_{\mathbf{y},\pi}$ can be simplified to

$$\boldsymbol{P}_{\mathbf{y},\pi} = \boldsymbol{\mu}_{\theta|\mathbf{z}}^T\mathbf{C}\boldsymbol{\mu}_{\theta|\mathbf{z}} + \begin{bmatrix} \alpha_1 & & 0 \\ & \ddots & \\ 0 & & \alpha_m \end{bmatrix}\mathrm{Tr}\big\{\boldsymbol{P}_{\theta|\mathbf{z}}\mathbf{C}\big\}. \tag{35}$$

To simplify notation in the remaining part of the paper, it is assumed that $\boldsymbol{P}_\theta$ and $\boldsymbol{P}_\theta^j$ can be used interchangeably. Furthermore, according to equation (34), $\mathrm{Tr}\{\boldsymbol{P}_{\theta|\mathbf{z}}\mathbf{C}\}$ does not depend on $\mathbf{z}$ and can therefore be calculated in advance.

B. The hyperparameter α

Estimates of α_j, which were assumed known in the previous section, are preferably derived from the posterior distribution conditioned on the propagated sigma-points $\mathbf{z}$:

$$p(\alpha_j|\mathbf{z}) \propto p(\mathbf{z}|\alpha_j)p(\alpha_j). \tag{36}$$

The posterior, on the other hand, relies on expressions for the likelihood $p(\mathbf{z}\big|\alpha_j)$ and the prior $p(\alpha_j)$.

1) The likelihood function: In our setting, $\boldsymbol{\theta}^j$ is a zero-mean Gaussian random variable, conditioned on α_j, and so is the linearly dependent observations $\mathbf{z}^j$. However, from the results in Appendix C it follows that the mean is known for the cases we study and, consequently, is independent of the hyperparameter prior. The observation vector of interest, $\tilde{\mathbf{z}}^j$, is therefore the j^{th} column in $\big[g(\mathbf{x}^0)-\boldsymbol{\theta}_0, \ldots, g(\mathbf{x}^{2n})-\boldsymbol{\theta}_0\big]^T$, and the likelihood function takes the following simple form:

$$p(\tilde{\mathbf{z}}^j|\alpha_j) = \frac{1}{(2\pi)^{\frac{\rho}{2}}(\alpha_j)^{\frac{\rho}{2}}\sqrt{|\mathbf{H}\boldsymbol{P}_\theta^j\mathbf{H}^T|}}\; e^{-\frac{1}{2\alpha_j}\tilde{\mathbf{z}}^{jT}(\mathbf{H}\boldsymbol{P}_\theta^j\mathbf{H}^T)^{-1}\tilde{\mathbf{z}}^j}, \tag{37}$$

in which ρ is the number of observations, in this case $2n+1$.

2) The prior: In the absence of prior knowledge of α_j, we want the prior to be noninformative to ensure a weak influence on the posterior distribution. It is argued in [23] that

$$p(\alpha_j) \propto 1/\alpha_j, \tag{38}$$

is a sensibly vague prior with respect to the likelihood (37).

3) The posterior distribution: The expression for the posterior distribution, using the likelihood (37) and prior (38), is:

$$p(\tilde{\mathbf{z}}^j|\alpha_j)\,p(\alpha_j) \propto \frac{1}{\alpha_j^{\frac{\rho}{2}+1}}\; e^{-\frac{1}{2\alpha_j}d^2}, \tag{39}$$

where $d^2 = \tilde{\mathbf{z}}^{j\,T}(\mathbf{H}\boldsymbol{P}_\theta^j\mathbf{H}^T)^{-1}\tilde{\mathbf{z}}^j$. The above expression is proportional to the scaled inverse chi-square distribution, so

$$\alpha_{j|\mathbf{z}} \sim \text{inv-}\chi^2(\nu, s^2), \tag{40}$$

with parameters $\nu = \rho$ and $s^2 = d^2/\rho$. The mean and mode of the scaled inverse chi-square distribution are:

$$\mathbb{E}(\alpha_j) = \frac{\nu}{\nu-2}s^2, \tag{41}$$

$$\text{mode}(\alpha_j) = \frac{\nu}{\nu+2}s^2, \tag{42}$$

and can be used as point estimates of α_j in the posterior covariance matrix expression (35). Note that the conditional mean (33) is unaffected by the hyperparameter. The algorithm presented in Section V-C employs the mode (42) as a point estimate of α_j.

C. The marginalized transform (MT) estimator

We have now reached the point where the MT estimation algorithm can be summarized, and somewhat simplified, in a few easy steps. There are two design decisions that can be made independently: the order of the transforming polynomial, p, and the sigma-point selection scheme. Using $2 \le p \le 3$ for

the cubature points and $2 \leq p \leq 5$ for the UT points assures a fully known mean (further explained in Section VI-B).

For $\mathbf{x} \in \mathbb{R}^n$, $\mathbf{y} = g(\mathbf{x}) \in \mathbb{R}^m$, $\mathbf{x} \sim \mathcal{N}(\boldsymbol{\mu}_\mathbf{x}, \boldsymbol{P}_\mathbf{x})$

1) Select a prior covariance matrix $\boldsymbol{\Sigma}$, a diagonal $p \times p$ matrix, $p \leq 5$, with at least two nonzero elements. See, e.g., the priors used in Section VIII.

2) Generate sigma-points using $w_0 = 1 - \frac{n}{3}$:

$$\begin{aligned}\mathbf{x}^0 &= \mathbf{0}_{n\times 1}\\ \mathbf{x}^k &= \begin{cases} +\left(\sqrt{\frac{n}{(1-w_0)}\mathbf{I}_{n\times n}}\right)_k & ,\ 1 < k \leq n \\ -\left(\sqrt{\frac{n}{(1-w_0)}\mathbf{I}_{n\times n}}\right)_{k-n} & ,\ n < k \leq 2n \end{cases}\\ \chi &= \left[\mathbf{x}^0, \mathbf{x}^1, \ldots, \mathbf{x}^{2n}\right].\end{aligned}$$

(although $w_0 = 0$ *can* be used if $p \leq 3$, and for $p = 2$, *any* $2n + 1$ points can be used).

3) Set $\boldsymbol{P}_\theta^{i,j} = \boldsymbol{\Sigma}$ in equation (24) to form $\boldsymbol{P}_\theta$. The value for $\sigma_{\theta_0}^2/\alpha_j$ will not matter. Calculate $\mathbf{w}, \mathbf{C}, \mathbf{H}(\chi)$ and $\boldsymbol{P}_{\theta|\mathbf{z}}$ using equations (20), (22), (32) and (34) respectively.

4) Propagate the sigma-points:

$$\mathbf{z} = \left[g(\boldsymbol{\mu}_\mathbf{x} + \sqrt{\boldsymbol{P}_\mathbf{x}}\mathbf{x}^0), \ldots, g(\boldsymbol{\mu}_\mathbf{x} + \sqrt{\boldsymbol{P}_\mathbf{x}}\mathbf{x}^{2n})\right].$$

5) Compute the mean, $\bar{\mathbf{y}}_\pi$, using equation (25) and (33):

$$\begin{aligned}\boldsymbol{\mu}_{\theta|\mathbf{z}} &= \boldsymbol{P}_\theta \mathbf{H}^T \left[\mathbf{H}\boldsymbol{P}_\theta\mathbf{H}^T\right]^{-1}\mathbf{z}^T\\ \bar{\mathbf{y}}_\pi &= \boldsymbol{\mu}_{\theta|\mathbf{z}}\mathbf{w}\end{aligned}$$

6) Estimate the modes of the hyperparameters:

$$\hat{\alpha}_j = \frac{1}{(2n+1)+2}\tilde{\mathbf{z}}^{j\,T}\left[\mathbf{H}\boldsymbol{P}_\theta\mathbf{H}^T\right]^{-1}\tilde{\mathbf{z}}^{j\,T},$$

where $\tilde{\mathbf{z}}^{j\,T}$ is the j^{th} row in the observation matrix with subtracted mean, $\left[g(\mathbf{x}^0) - \bar{\mathbf{y}}_\pi, \ldots, g(\mathbf{x}^{2n}) - \bar{\mathbf{y}}_\pi\right]$.

7) Calculate the covariance matrix, $\boldsymbol{P}_{\mathbf{y},\pi}$, using equation (35).

Steps $1-3$ can be done in advance, as well as computing $\boldsymbol{P}_\theta\mathbf{H}^T\left[\mathbf{H}\boldsymbol{P}_\theta\mathbf{H}^T\right]^{-1}$, $\left[\mathbf{H}\boldsymbol{P}_\theta\mathbf{H}^T\right]^{-1}$ and $\mathrm{Tr}\{\boldsymbol{P}_{\theta|\mathbf{z}}\mathbf{C}\}$, in that way simplifying the algorithm significantly. For example, the calculation of the mean can be identical to the UT, cubature rule or to the DD2, for which also the covariance matrix estimator can be the same — all depending on the design of the prior, see the discussion in Section VI.

D. Calculating the posterior cross-covariance matrix

It is sometimes required to know the cross-covariance between the state, $\mathbf{x}$, and the transformed state, $\mathbf{y} = g(\mathbf{x})$. In the filtering algorithm that will be presented in Section VII-C, it is a necessity, and is in fact already known from estimating $\boldsymbol{\mu}_{\theta|\mathbf{z}}$. The cross-covariance matrix is:

$$\begin{aligned}\boldsymbol{P}_{\mathbf{xy}}(\boldsymbol{\theta}) &= \int_{R^n} \mathcal{N}(\mathbf{x}\,; \mathbf{0}, \mathbf{I}_n)\left[\mathbf{x} - \mathbb{E}[\mathbf{x}]\right]\left[g(\mathbf{x};\boldsymbol{\theta}) - \bar{\mathbf{y}}(\boldsymbol{\theta})\right]^T d\mathbf{x}\\ &= \mathbb{E}[\mathbf{x}[\boldsymbol{\theta}^T\mathbf{h}(\tilde{\mathbf{x}}) - \boldsymbol{\theta}^T\mathbf{w}]^T]\\ &= \mathbb{E}[\mathbf{x}[\mathbf{h}(\mathbf{x}) - \mathbf{w}]^T]\boldsymbol{\theta}\\ &= \mathbf{D}\boldsymbol{\theta}. \end{aligned} \tag{43}$$

The sparse matrix $\mathbf{D} \triangleq \mathbb{E}[\tilde{\mathbf{x}}[\mathbf{h}(\tilde{\mathbf{x}}) - \mathbf{w}]^T]$ is constant and can be written:

$$\mathbf{D} = \begin{bmatrix} 0 & [1,0,\ldots,0] & \mathbf{0}^T & \ldots \\ 0 & \mathbf{0}^T & \ddots & \ddots \\ \vdots & \vdots & \ddots & [1,0,\ldots,0] \end{bmatrix}, \tag{44}$$

which follows from the orthogonality property (83) of Hermite polynomials described in Appendix B (recall that $x = H_1(x)$). In other words, $\boldsymbol{P}_{\mathbf{xy}}(\boldsymbol{\theta})$ is the $n \times m$ matrix of all first order weights:

$$\boldsymbol{P}_{\mathbf{xy}}(\boldsymbol{\theta}) = \begin{bmatrix} \theta^{1,1}(1) & \ldots & \theta^{1,m}(1) \\ \theta^{2,1}(1) & \ldots & \theta^{2,m}(1) \\ \vdots & & \vdots \\ \theta^{n,1}(1) & \ldots & \theta^{n,m}(1) \end{bmatrix}. \tag{45}$$

The above cross-covariance matrix describes the relation to $\mathbf{x} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_{n\times n})$, whereas the relation to a correlated state is established by multiplication with $\sqrt{\boldsymbol{P}_\mathbf{x}}$. Including the square-root matrix and carrying out the marginalization of $\boldsymbol{\theta}$ in (43) yields

$$\begin{aligned}\boldsymbol{P}_{\mathbf{xy},\pi} &= \sqrt{\boldsymbol{P}_\mathbf{x}}\ \mathbf{D}\mathbb{E}\left[\boldsymbol{\theta}\middle|\mathbf{z}, \chi\right]\\ &= \sqrt{\boldsymbol{P}_\mathbf{x}}\ \mathbf{D}\boldsymbol{\mu}_{\theta|\mathbf{z}},\end{aligned} \tag{46}$$

which is the estimate of the cross covariance matrix.

VI. Analysis and comparison

In this section, we further explain the behavior of the proposed estimator, and clarify the relationship with other sigma-point estimators.

A. Posterior uncertainties in mean and covariance

First, we analyze our estimates in terms of their distributions. Conditioned on α, the mean, $\bar{\mathbf{y}}(\boldsymbol{\theta})$, is a Gaussian random variable with covariance

$$\mathbb{E}\left[[\bar{\mathbf{y}}(\boldsymbol{\theta}) - \bar{\mathbf{y}}_\pi][\bar{\mathbf{y}}(\boldsymbol{\theta}) - \bar{\mathbf{y}}_\pi]^T|\alpha\right] = \mathbf{I}_{m\times m}\mathbf{w}\boldsymbol{P}_{\theta|\mathbf{z}}\mathbf{w}^T. \tag{47}$$

The distribution of the elements in the covariance matrix, $\boldsymbol{P}_\mathbf{y}(\boldsymbol{\theta})$, is less trivial; diagonal elements are weighted sums of chi-square distributed variables, whereas the off-diagonal elements are created from products between independent Gaussian random variables. This could be looked upon as a weighted sum of Wishart distributed matrices created from the rows, $\boldsymbol{\theta}_i$, of $\boldsymbol{\theta}$:

$$\boldsymbol{P}_\mathbf{y}(\boldsymbol{\theta}) = \sum_{k=0}^{pn} \boldsymbol{\theta}_k^T\boldsymbol{\theta}_k c_{k+1}, \tag{48}$$

where c_k is the k^{th} diagonal element in $\mathbf{C}$, defined in equation (22).

Equation (47) illustrates how uncertainties in $\boldsymbol{\theta}$ affect $\bar{\mathbf{y}}(\boldsymbol{\theta})$, and it is desirable to design an estimator such that this variance equals zero. Inserting the expression for $\boldsymbol{P}_{\theta|\mathbf{z}}$, from equation (34), into (47), we see that the covariance of $\bar{\mathbf{y}}(\boldsymbol{\theta})$ is

$$\text{Cov}(\bar{\mathbf{y}}(\boldsymbol{\theta})) = \mathbf{w}\left(\mathbf{I} - \boldsymbol{P}_\theta \mathbf{H}^T \left[\mathbf{H}\boldsymbol{P}_\theta \mathbf{H}^T\right]^{-1} \mathbf{H}\right) \boldsymbol{P}_\theta \mathbf{w}^T. \tag{49}$$

One of the arguments for sigma-point approaches has been that it is easier to approximate the probability distribution than the transforming function [7], [24]. However, it is not required for $\boldsymbol{\theta}$ to be fully known ($\boldsymbol{P}_{\theta|\mathbf{z}} = 0$) in order for the estimate to be exact; we see from equation (49) that it is enough to project the uncertainties in $\boldsymbol{\theta}$ onto the plane orthogonal to the vector $\mathbf{w}$. In Appendix C it is shown that the selection scheme (4)–(5) attains this projection, which means that $\bar{\mathbf{y}}_\pi = \bar{\mathbf{y}}(\boldsymbol{\theta})$ with probability one. In other words, $\bar{\mathbf{y}}(\boldsymbol{\theta})$ is identical for all polynomials passing through the sigma-points.

The result follows from using an integration rule, well-known from the literature, [12], [14], which integrates these functions correctly. However, the new derivation provided here is conceptually different and may be more intuitive to some readers. Furthermore, the type of uncertainty analysis performed in this paper can provide an important tool for designing new sigma-point selection schemes in the future.

B. Comparison with the UT and the cubature rule

Contrary to the UT and the cubature rule, the presented method suggests to calculate the covariance matrix using a model of the transformation, and the estimates are therefore conceptually different. The estimates of the mean, however, are easier to compare; the UT and the cubature rule employ known integration rules, and the proposed method can yield these rules under certain conditions. To show the similarities, we write the MT estimator of the mean (25) on the same form as the UT estimator (7):

$$\bar{\mathbf{y}}_\pi = \mathbf{z}\left[\boldsymbol{P}_\theta \mathbf{H}^T \left[\mathbf{H}\boldsymbol{P}_\theta \mathbf{H}^T\right]^{-1}\right]^T \mathbf{w}. \tag{50}$$

This is clearly a weighted sum, $\bar{\mathbf{y}}_\pi = \mathbf{z}\boldsymbol{\lambda}$, of the evaluated sigma-points, with a column weight vector

$$\boldsymbol{\lambda} = \left[\mathbf{H}\boldsymbol{P}_\theta \mathbf{H}^{\mathbf{T}}\right]^{-1} \mathbf{H}\boldsymbol{P}_\theta \mathbf{w}. \tag{51}$$

The MT and UT estimators are the same when the elements of $\boldsymbol{\lambda}$ are identical to the UT weights.

The definition of the precision of an integration rule is [14]: *'A rule is said to have precision p if it integrates monomials up to degree p exactly, that is, monomials $\Pi_{i=1}^d x_i^{k_i}$ with $k_i \geq 0$ and $\sum_{i=1}^d k_i \leq p$, but not exactly for some monomials of degree $\sum_{i=1}^d k_i = p+1$'.*

For the presented method, this definition is equivalent to having no uncertainties in $\bar{\mathbf{y}}(\boldsymbol{\theta})$, when the prior includes all monomials up to degree p. It is shown in Appendix C that the sigma-point selection scheme (4) – (5) satisfies exactly this — the MT and UT estimators for the mean are then identical. The explicit model assumptions in the proposed method coincide with the implicit assumptions in the sigma-point filter, and the actual values in the prior covariance matrix, $\boldsymbol{P}_\theta$, no longer affect the result.

The integration rule used by the UT and the cubature rule have precision 3, which can be quite limiting. A simple example serves as illustration:

$$y = x_1 x_2, \quad x \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_{2\times 2}). \tag{52}$$

The variance of y is $\mathbb{E}[x_1^2 x_2^2] = 1$, but the sigma-point methods discussed in this paper all fail to calculate the variance correctly. However, the prior used in the presented method explicitly excludes cross-terms in the model, so the result should come as no surprise. Moreover, the solution is straightforward: modify the model to include also cross-terms and add sigma-points to observe them. It should be mentioned here that the MT and the UT, with $w_0 = 1 - n/3$, would have precision 5 if it weren't for these cross-terms, i.e., single-element monomials, x_i^p, are correctly integrated up to $p = 5$.

Contrary to the UT, the MT can be tuned without moving the sigma-points. The cubature rule, on the other hand, cannot be tuned at all, and the position of the sigma-points varies in a predetermined manner with the dimensionality, n. For instance, in a tracking system where targets are tracked using a joint state vector, the performance of the cubature estimator depends on the number of targets, even if the targets are well separated with independent measurements (with respect to other targets).

C. Comparison with the divided difference filter

The DD2 is based on a second-order polynomial approximation of the transforming function, with cross-terms excluded. The MT assumes that the underlying distribution is Gaussian, which corresponds to setting the DD2 design parameter $h = \sqrt{3}$. It is possible to design an MT-prior to correspond to this estimator. More specifically, assuming a second order polynomial and using the UT sigma-points yields equally many unknowns as observations. The second order polynomial is therefore fully known, i.e. there are no posterior uncertainties in the parameter vector $\boldsymbol{\theta}$, and the estimators are, for this particular prior, identical.

D. Sigma-point selection and non-linear transformations

The effects of employing a particular set of sigma points with the MT can be evaluated in terms of the posterior uncertainties of the estimates. However, our focus here is to evaluate the MT performance when using the $2n+1$ UT points, and the $2n$ cubature points, where the main difference between these sets is that the cubature rule does not employ a weight in the distribution mean.

It is foreseeable that there will be functions for which the integral of a polynomial passing through the evaluated sigma-points, may constitute a worse approximation of the actual integral, than the integral over a lower order polynomial passing through fewer points. For instance, in [9], it was shown that the cubature rule performed better than the DD2

in estimating the mean of the function

$$g(\mathbf{x}) = \frac{1}{(\sqrt{1+\mathbf{x}^T\mathbf{x}})^q}, \tag{53}$$

when the integer q and the dimensionality of $\mathbf{x}$ was increased. Under these circumstances, the function (53) does not resemble a polynomial, and including a sigma-point in the mean $\mathbb{E}[\mathbf{x}]$ degrades performance. It cannot, however, be argued that it is generally sound to exclude that particular sigma-point — it has to be judged depending on the function. Including the point provides information of the function, which obviously sometimes is helpful, especially when calculating the covariance matrix. For example, the covariance matrix for functions symmetric over the covariance contour will be zero when calculated using the cubature rule, e.g.:

$$y = x^2, \quad x \sim \mathcal{N}(0,1). \tag{54}$$

If all propagated points have the same value this will also be the estimate of the mean, i.e., $g(\mathbf{x}^i) = \bar{\mathbf{y}}$ for all sigma-points. The variance estimate is then:

$$\sum_{i=0}^{2n} w_i[g(\mathbf{x}^i) - \bar{\mathbf{y}}][g(\mathbf{x}^i) - \bar{\mathbf{y}}]^T = \mathbf{0}.$$

This would be the case also for (53), if $\mathbf{x} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$. In real situations this is rarely the case, but nevertheless illustrates an undesired behavior.

The transformation (53) also serves to illustrate that sigma-point methods can perform well also for non-polynomial transformations, since a polynomial approximation need not resemble the transforming function in order to approximate its integral.

VII. Application example: Recursive filtering

Robust recursive filters, e.g., for tracking a continuous process measured at discrete time instances, are arguably very valuable. A famous solution is the Kalman filter (KF) [4], although the KF is applicable only when models are linear. Several filters intended for usage with non-linear models share a similar structure, differing only in how they estimate moments, e.g., the UKF, CKF, and EKF. By applying the marginalization technique presented in this paper in a similar fashion, the marginalized Kalman filter is created — the MKF.

A. System model

A discrete-time non-linear system, described by the state vector, $\mathbf{x}_k$, is assumed to evolve according to the model:

$$\mathbf{x}_k = f(\mathbf{x}_{k-1}, \mathbf{w}_{k-1}). \tag{55}$$

Observations, $\mathbf{y}_k$, are provided at discrete time instances:

$$\mathbf{y}_k = h(\mathbf{x}_k, \mathbf{v}_k). \tag{56}$$

The noise terms $\mathbf{w}_k, \mathbf{v}_k$ are modeled as zero mean independent white Gaussian noise. The goal is to calculate the posterior distribution $p(\mathbf{x}_k|\mathbf{Y}_k)$, where $\mathbf{Y}_k$ is the collection of all available measurements, $[\mathbf{y}_1, \ldots, \mathbf{y}_k]$. Estimates of the state vector are often denoted $\hat{\mathbf{x}}_{k|k}$, where the first subscript refers to the time index of the state and the latter to the time index of the last measurement used to update the state.

B. The one-step linear estimation algorithm

An accustomed approach for calculating the posterior distribution, used for example by the EKF, UKF, CKF and DD2 filters, is to apply the LMMSE estimator for each new observation. The filter performs two operations:

1) Prediction: Given $p(\mathbf{x}_{k-1}|\mathbf{Y}_{k-1})$, calculate the first two moments of the state distribution at the time of the next unused measurement:

$$\begin{aligned}\hat{\mathbf{x}}_{k|k-1} &= \mathbb{E}[\mathbf{x}_k|\mathbf{Y}_{k-1}] \\ &= \mathbb{E}[f(\mathbf{x}_{k-1},\mathbf{w}_{k-1})|\mathbf{Y}_{k-1}] \qquad (57)\\ \boldsymbol{P}_{k|k-1} &= \mathrm{Cov}(\mathbf{x}_k|\mathbf{Y}_{k-1}) \\ &= \mathbb{E}[f(\mathbf{x}_{k-1},\mathbf{w}_{k-1})f(\mathbf{x}_{k-1},\mathbf{w}_{k-1})^T|\mathbf{Y}_{k-1}] \qquad (58)\\ &\quad - \hat{\mathbf{x}}_{k|k-1}\hat{\mathbf{x}}_{k|k-1}^T\end{aligned}$$

2) Update: Correct the prediction, $\hat{\mathbf{x}}_{k|k-1}$, using the measurement, $\mathbf{y}_k$. The best update that is linear in $\mathbf{y}_k$, is given by the LMMSE estimator [25]:

$$\hat{\mathbf{x}}_{k|k} = \hat{\mathbf{x}}_{k|k-1} + \boldsymbol{P}_{\mathbf{xy}}\mathbf{S}_{k|k-1}^{-1}\left(\mathbf{y}_k - \hat{\mathbf{y}}_{k|k-1}\right). \tag{59}$$

The estimator (59) requires knowledge of the mean, $\hat{\mathbf{y}}_{k|k-1}$, and covariance, $\mathbf{S}_{k|k-1}$, of the measurement distribution, as well as the cross-covariance matrix $\boldsymbol{P}_{\mathbf{xy}}$:

$$\begin{aligned}\hat{\mathbf{y}}_{k|k-1} &= \mathbb{E}[\mathbf{y}_k|\mathbf{Y}_{k-1}] \\ &= \mathbb{E}[h(\mathbf{x}_k,\mathbf{v}_k)|\mathbf{Y}_{k-1}] \qquad (60)\\ \mathbf{S}_{k|k-1} &= \mathrm{Cov}(\mathbf{y}_k|\mathbf{Y}_{k-1}) \qquad (61)\\ &= \mathbb{E}[h(\mathbf{x}_k,\mathbf{v}_k)h(\mathbf{x}_k,\mathbf{v}_k)^T|\mathbf{Y}_{k-1}] - \hat{\mathbf{y}}_{k|k-1}\hat{\mathbf{y}}_{k|k-1}^T. \\ \boldsymbol{P}_{\mathbf{xy}} &= \mathrm{Cov}(\mathbf{x}_k,\mathbf{y}_k|\mathbf{Y}_{k-1}) \\ &= \mathbb{E}[\mathbf{x}_k h(\mathbf{x}_k,\mathbf{v}_k)^T|\mathbf{Y}_{k-1}] - \hat{\mathbf{x}}_{k|k-1}\hat{\mathbf{y}}_{k|k-1}^T \qquad (62)\end{aligned}$$

The matrix mean squared error (MSE) of the estimate (59) is used as an approximation of the posterior covariance matrix, $\boldsymbol{P}_{k|k}$. The matrix MSE is:

$$\begin{aligned}&\mathbb{E}\left[[\mathbf{x}_k - \hat{\mathbf{x}}_{k|k}][\mathbf{x}_k - \hat{\mathbf{x}}_{k|k}]^T|\mathbf{y}_k\right] \qquad (63)\\ &\qquad = \boldsymbol{P}_{k|k-1} - \boldsymbol{P}_{\mathbf{xy}}\mathbf{S}_{k|k-1}^{-1}\boldsymbol{P}_{\mathbf{yx}},\end{aligned}$$

and is a reasonable approximation to a posterior covariance matrix which does not depend on the observation $\mathbf{y}_k$. Expressed in terms of the so called gain matrix, $\mathbf{K}_k = \boldsymbol{P}_{\mathbf{xy}}\mathbf{S}_{k|k-1}^{-1}$, the expressions for the state update are:

$$\hat{\mathbf{x}}_{k|k} = \hat{\mathbf{x}}_{k|k-1} + \mathbf{K}_k\left(\mathbf{y}_k - \hat{\mathbf{y}}_{k|k-1}\right) \tag{64}$$

$$\boldsymbol{P}_{k|k} = \boldsymbol{P}_{k|k-1} - \mathbf{K}_k\mathbf{S}_{k|k-1}\mathbf{K}_k^T. \tag{65}$$

To sum up, the filter approximates the first two moments of the posterior distribution, $p(\mathbf{x}_k|\mathbf{Y}_k)$, with the estimate of the mean (64) and the matrix MSE (65), concluding the recursion.

C. The marginalized Kalman filter (MKF)

The MKF is the recursive filter following the application of the MT to steps 1–2 in the previous section. The state vector can be augmented to include noise terms, described, e.g., in [7].

1) MKF *prediction:* Assume the state vector is Gaussian, i.e.,

$$p(\mathbf{x}_{k-1}|\mathbf{Y}_{k-1}) = \mathcal{N}(\mathbf{x}_{k-1}; \hat{\mathbf{x}}_{k-1|k-1}, \boldsymbol{P}_{k-1|k-1}).$$

Use the algorithm in Section V-C to calculate the mean (57) and covariance (58) of the predictive distribution,

$$p(\mathbf{x}_k|\mathbf{Y}_{k-1}) \approx \mathcal{N}(\mathbf{x}_k; \hat{\mathbf{x}}_{k|k-1}, \boldsymbol{P}_{k|k-1}).$$

2) MKF *update:* Apply the algorithm a second time to calculate the mean (60) and covariance (61) of the measurement distribution. The cross-covariance matrix (62) is given by equation (46). Calculate the gain matrix, $\mathbf{K}_k = \boldsymbol{P}_{\mathbf{xy}}\mathbf{S}_{k|k-1}^{-1}$, and approximate the posterior distribution

$$p(\mathbf{x}_k|\mathbf{Y}_k) \approx \mathcal{N}(\mathbf{x}_k; \hat{\mathbf{x}}_{k|k}, \boldsymbol{P}_{k|k}),$$

using the LMMSE estimate (64) and the matrix MSE (65).

VIII. Simulation examples

The cubature rule is a special case of the unscented transform with the benefit that the estimated covariance matrix is always positive-definite — a property shared also by the proposed method. Further, the results in [9] indicate that the cubature rule performs better than the divided difference filter. Therefore, our main goal is to show how the presented method performs compared to the cubature transform. Two examples are examined: the transformation from polar to Cartesian coordinates, which is also commonly used to illustrate the performance of the unscented transform, and the bearings-only tracking problem [26].

In the first evaluation we use the Kullback-Leibler (KL) discrimination[1] to measure how much a distribution $q(\mathbf{y})$ differs from a reference distribution $p(\mathbf{y})$ [28]:

$$d_{\mathrm{KL}}(p, q) = \int p(\mathbf{y}) \log \frac{p(\mathbf{y})}{q(\mathbf{y})} d\mathbf{y}. \tag{66}$$

This measure was also used in [9] to evaluate the cubature rule, which further motivates using the same approach here. The distributions p and q are approximated as Gaussians, for which $d_{KL}(p, q)$ can be calculated analytically. The first two moments of the reference distribution, p, are estimated using Monte Carlo integration:

$$\int p(x)g(x)dx \approx \sum_{n=1}^{N} g(x_n). \tag{67}$$

Two slightly different versions, the MT^5 and the MT^3, of the presented method are evaluated. The MT^5 is implemented according to the algorithm in Section V-C, with $p = 5$, using the $2n+1$ UT sigma-points. However, in order to compare the method fairly to the cubature rule, the MT^3 is introduced, using $p = 3$ and the $2n$ cubature sigma-points. This is not the same as setting $w_0 = 0$ in the second step of the algorithm, which in practice would exclude the point $\mathbf{x}^0$ in the calculation of the mean but not in the calculation of the covariance matrix.

[1]Usually referred to as the Kullback-Leibler divergence, although when introduced in [27], the authors used the term "divergence" for the symmetric measure $d_{\mathrm{KL}}(p, q) + d_{\mathrm{KL}}(q, p)$.

A. Polar to Cartesian transformation

In this section the MT^3, using two slightly different priors, is compared to the cubature rule. Let $\mathbf{y} = g(\mathbf{x})$ be the transformation from a polar coordinate system defined in terms of range, r, and azimuth, ψ, to a Cartesian coordinate system:

$$\mathbf{x} = \begin{bmatrix} r \\ \psi \end{bmatrix}, \quad \mathbf{y} = \begin{bmatrix} x_1 \cos x_2 \\ x_1 \sin x_2 \end{bmatrix}. \tag{68}$$

By modifying the prior, the presented method can be optimized to yield excellent results for a narrow family of transformations. However, this is not a fair comparison and often not a realistic approach. Instead we use the same prior for the 11 positions in Fig. 2, and for each position we evaluate 8 different azimuth measurement noise variances, σ_ψ^2:

$$\sigma_\psi^2 = [5^2, 10^2, 15^2, 20^2, 25^2, 30^2, 35^2, 40^2](\frac{\pi}{180})^2 \quad [\mathrm{rad}^2]. \tag{69}$$

The range measurement noise variance is constant throughout all evaluations, $\sigma_r^2 = 0.5$ [m^2].

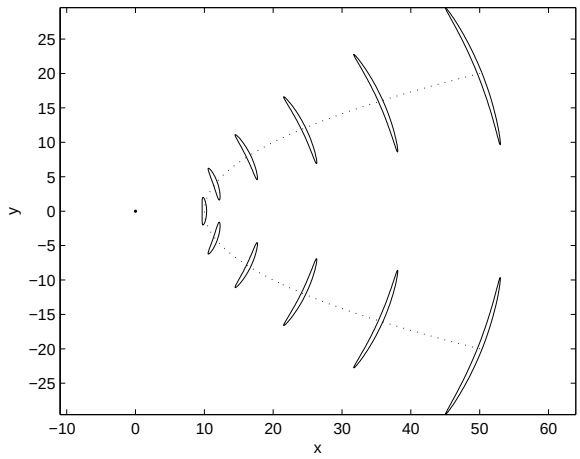

Fig. 2. A sensor, situated in the origin, with uncertainties in range and angle measurements observes a target at eleven positions. The "banana-shaped" contours are measurement space covariance contours, transformed to the Cartesian coordinate system.

To illustrate the influence of the prior, we present results for two different priors, both assuming a zero-mean Gaussian distribution of $\boldsymbol{\theta}$. The first one is created using the simple assumption that the function is a 2$^{\text{nd}}$ order polynomial where the higher order term is relatively small, whereas the second one has been numerically derived to perform well in this scenario:

$$\boldsymbol{\Sigma}_1 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \frac{1}{100} & 0 \\ 0 & 0 & 0 \end{bmatrix}, \; \boldsymbol{\Sigma}_2 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0.036 & 0 \\ 0 & 0 & 0.0007 \end{bmatrix}. \tag{70}$$

The cubature evaluation points are used by all three methods and, as argued in Section VI-B, the prior variance for the mean, θ_0, does not influence the estimate.

The average Kullback-Leibler discrimination is presented in Table I and the mean for each position and noise variance is displayed in Fig. 3. The reference density was calculated using 10^5 samples. The results show that, although all methods

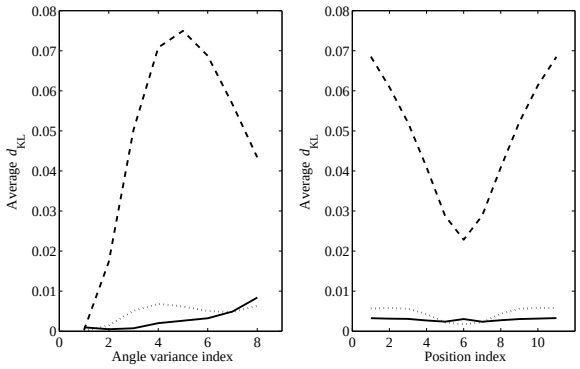

Fig. 3. The left figure shows the average Kullback-Leibler discrimination for the different azimuth noise variances, whereas the right figure shows the average Kullback-Leibler discrimination for the positions. The dashed line illustrates the Cubature rule, the dotted line represents the use of $\mathbf{\Sigma}_1$, and the solid line the use of $\mathbf{\Sigma}_2$.

TABLE I
AVERAGE KULLBACK-LEIBLER DISCRIMINATION

	Average KL-discrimination $[\times 10^{-4}]$
Cubature rule	478
MT^3, $\mathbf{\Sigma}_1$	45
MT^3, $\mathbf{\Sigma}_2$	29

perform very well in absolute numbers, the marginalized sigma-point estimator outperforms the Cubature rule using the same points χ. It can also be seen that Σ_1 is the better description for some noise models, and for position 6, but that Σ_2 performs better on average.

B. Bearings only tracking

The bearings only tracking problem is well-studied and arises in passive sensor applications such as sonar tracking. Several filters have been designed for this particular task, such as the range-parameterized EKF [26], but since we are interested in comparing sigma-point filters, those filters are not included in the comparison. Two MKF versions, based on the MT^5 and the MT^3, are compared to the CKF, the UKF and the DD2-filter.

The scenario we consider here, tracking of a non-maneuvering submarine, is illustrated in Fig. 4. Most parameter values are taken from [26]. The state vector contains the Cartesian position and velocity, $\mathbf{x} = [x\ y\ \dot{x}\ \dot{y}]^T$, and bearing observations are non-linear transformations of $\mathbf{x}$, with additive Gaussian noise:

$$\theta = \tan^{-1}\left(\frac{y}{x}\right) + w. \tag{71}$$

The variance of the measurement noise, $w_k \sim \mathcal{N}(0, \sigma_w^2)$, is known to the tracking algorithms, which are also given perfect knowledge of the prior distribution; for each simulation, the initial position of the target is generated from the prior. The process model is linear:

$$\mathbf{x}_k = \begin{bmatrix} 1 & 0 & T & 0 \\ 0 & 1 & 0 & T \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \mathbf{x}_{k-1} + \begin{bmatrix} \frac{T^2}{2} & 0 \\ 0 & \frac{T^2}{2} \\ T & 0 \\ 0 & T \end{bmatrix} \mathbf{v}_k, \tag{72}$$

with process noise, $\mathbf{v}_k \sim \mathcal{N}(0, \sigma_{\mathbf{v}}^2 \mathbf{I}_{2\times 2})$. The state distribution is assumed Gaussian, and the predicted distribution, which is consequently also Gaussian, is correctly calculated by all five filters. Hence, the methods differ only in the calculation of the measurement distribution and the cross-covariance matrix. The parameter values are:

$$\sigma_{\mathbf{v}} = \sqrt{10^{-5}}\,\frac{m}{s^2}, \quad \sigma_w = 1.5\,^\circ, \quad T = 60\ s, \quad N = 30,$$

where N is the length of a trajectory. The filter is initiated using the scheme in [26], at

$$\mathbf{x}_0 = \begin{bmatrix} 3000 \\ 4000 \\ -0.6 \\ -0.8 \end{bmatrix}, \quad \boldsymbol{P}_0 \approx \begin{bmatrix} 592^2 & 682^2 & 0 & 0 \\ 682^2 & 816^2 & 0 & 0 \\ 0 & 0 & 0.57 & -0.35 \\ 0 & 0 & -0.35 & 0.34 \end{bmatrix},$$

which corresponds to a target at a range of 5 km, traveling towards the sensor at a speed of 1 m/s with uncertainties in range ($\sigma_r = 1000$ m), speed ($\sigma_s = 0.3$ m/s) and course ($\sigma_c = \frac{\pi}{\sqrt{12}}$ rad).

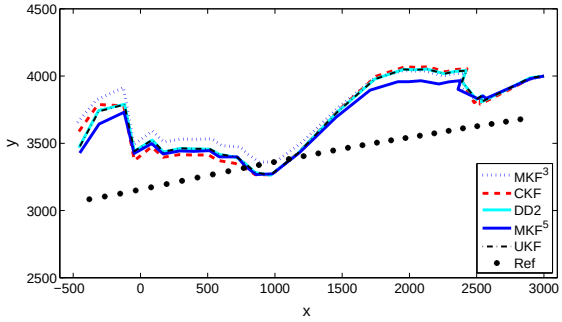

Fig. 4. Five different filters are applied to the tracking problem where a bearings-only sensor, situated in the origin, makes 30 observations of a moving target. In this particular example the target process noise is near-zero.

Two performance measures are averaged over 10^4 simulations: The MSE, ξ, and the average normalized estimation error squared (NEES), ζ:

$$\xi = \frac{1}{N}\sum_{k=1}^{N} [\hat{\mathbf{x}}_k^p - \mathbf{x}_k^p]^T [\hat{\mathbf{x}}_k^p - \mathbf{x}_k^p] \tag{73}$$

$$\zeta = \frac{1}{N}\sum_{k=1}^{N} [\hat{\mathbf{x}}_k^p - \mathbf{x}_k^p]^T \left(\hat{\boldsymbol{P}}_{\mathbf{k}}^{\mathbf{p}}\right)^{-1} [\hat{\mathbf{x}}_k^p - \mathbf{x}_k^p]. \tag{74}$$

Both are calculated for the position states, $\mathbf{x}^p = [x\ y]^T$, and its covariance matrix, $\boldsymbol{P}^p$. The results are summarized in Table II. When the posterior covariance matrix correctly describes the estimation error, the NEES is equal to the number of dimensions of the evaluated state vector, i.e., 2 in this example. Consequently, $\zeta > 2$ indicates that the covariance is underestimated and vice versa.

In this evaluation, the UT mean weight, w_0, is $1 - \frac{n}{3}$, the DD2 parameter, h, is $\sqrt{3}$, and the MKF^3 and MKF^5 are based

on the MT^3 and MT^5, respectively[2], with priors:

$$\mathbf{\Sigma}_{\text{MT}^3} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \frac{1}{10} & 0 \\ 0 & 0 & \frac{5}{100} \end{bmatrix}, \mathbf{\Sigma}_{\text{MT}^5} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & \frac{1}{10} & 0 & 0 & 0 \\ 0 & 0 & \frac{5}{100} & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{1000} & 0 \\ 0 & 0 & 0 & 0 & \frac{1}{1000} \end{bmatrix}.$$

TABLE II
RMSE, NEES, AND CPU TIME REQUIRED TO PROCESS A TRAJECTORY, AVERAGED OVER 10^4 SIMULATIONS

	RMSE, $\sqrt{\bar{\xi}}$	NEES, $\bar{\zeta}$	No. of σ-points	CPU time $[ms]$
MKF^3	1074	1.97	$2n$	28
CKF	1083	2.46	$2n$	24
DD2	1077	2.40	$2n+1$	23
MKF^5	1064	2.01	$2n+1$	29
UKF	1076	2.40	$2n+1$	25

From Table II we conclude that the choice of filters does not, on average, affect the MSE in particular and that the CKF, UKF and DD2 underestimate the size of the error. The MKF, however, performs very well in the NEES sense. Though the NEES should be used with care [29], this indicates that the MKF filters are better at self-assessing their accuracies. This can be explained in terms of the posterior uncertainties in $\boldsymbol{\theta}$, which contribute to the covariance matrix estimate through the additive diagonal matrix in equation (35). An accurate approximation of the posterior covariance matrix is important, e.g., in a Bayesian decision-making scheme.

A standard laptop with an Intel core I5 CPU, running at 2.4GHz, was used to run the filters in MATLAB. There is a slight increase in processing time for the MKF that originates from the calculation of the hyperparameter, $\boldsymbol{\alpha}$, which has no counterpart in the other filters.

IX. Conclusions

We have presented a derivative-free method, the marginalized transform (MT), for estimating the mean and covariance of a transformed Gaussian-distributed random variable, which has several beneficial properties. In summary, the method:

- performs better than well-known sigma-point methods, such as the UT, DD2, or cubature rule, in the evaluated estimation task and the bearings-only tracking scenario.
- is easy to apply, as the simplicity of derivative-free filters is maintained.
- has tuning-parameters that can be intuitively understood in terms of the model of the transforming function.

In a more general sense, we present a method for designing sigma-point estimators, based on explicit model assumptions. For example, it has been shown which assumptions lead to the integration rules of the DD2, UT, and the cubature rule.

Sigma-point filters have previously been analyzed in terms of the precision of the applied integral approximation. Still, as the non-linear functions encountered in most applications are not polynomial, we argue that it is relevant to ask what the estimates represent when they are *not* exact. A description of the latter is precisely what the MT gives; the family of functions contributing to the estimates.

[2]In other words, the MKF^3 and the MKF^5 estimates of the mean are calculated using the same rules as the CKF and the UKF/DD2, respectively.

Appendix A
UT covariance matrix estimates

The UT covariance matrix estimate (8) is on the form

$$\hat{\boldsymbol{P}}_y = \sum_{i=0}^{2n} w_i \mathbf{d}_i \mathbf{d}_i^T, \text{ with } \mathbf{d}_i = [g(\mathbf{x}^i) - \bar{\mathbf{y}}]. \tag{75}$$

Lemma 1: The covariance matrix estimate calculated by the UT is guaranteed to be positive-semidefinite when all weights are positive.

Proof: $\hat{\boldsymbol{P}}_y$ is positive-semidefinite if $\mathbf{x}^T \hat{\boldsymbol{P}}_y \mathbf{x} \geq 0$, and

$$\mathbf{x}^T \hat{\boldsymbol{P}}_y \mathbf{x} = \sum_{i=0}^{2n} w_i (\mathbf{x}^T \mathbf{d}_i)^2 \geq 0, \text{ if } w_i \geq 0 \, \forall i \tag{76}$$

Lemma 2: When $w_0 \notin [0,1]$, there are functions for which $\hat{\boldsymbol{P}}_y$ is not positive-semidefinite.

Proof: For example, there exists a function $g : \mathbb{R}^n \to \mathbb{R}^1$, symmetric such that

$$a = g(\mathbf{x}^i), i \in \{1, \dots 2n\} \tag{77}$$
$$b = g(\mathbf{x}^0). \tag{78}$$

The UT weights sum to one and $\bar{\mathbf{y}}$ is assumed zero,

$$w_0 b + 2n w_i a = 0, \tag{79}$$

leading to the following two relations:

$$w_i = \frac{1 - w_0}{2n} \triangleq w, \text{ and } a = b \frac{-w_0}{1 - w_0}. \tag{80}$$

The variance is negative if,

$$\begin{aligned} \sigma_y^2 = w_0 b^2 + 2nwa^2 &< 0 \\ \Leftrightarrow w_0 b^2 + (1 - w_0) b^2 \frac{(-w_0)^2}{(1 - w_0)^2} &< 0 \\ \Leftrightarrow w_0 (1 - w_0)^2 + (1 - w_0) w_0^2 &< 0. \end{aligned} \tag{81}$$

The left hand side on the last row is a second order polynomial with roots $w_0 = 0$ and $w_0 = 1$, and a maximum in $w_0 = 1/2$. In other words:

$$w_0 \notin [0,1] \Rightarrow \sigma_y^2 < 0. \tag{82}$$

Each diagonal element in the $m \times m$ covariance matrix, corresponding to $g : \mathbb{R}^n \to \mathbb{R}^m$, is calculated analogous to σ_y^2. The proof is therefore valid for any dimensionality.

Appendix B
Properties of Hermite polynomials

The univariate Hermite polynomials are orthogonal under integration under the Gaussian pdf, i.e., for $x \sim \mathcal{N}(0,1)$,

$$\mathbb{E}[H_i(x) H_j(x)] = \int p(x) H_i(x) H_j(x) dx = \begin{cases} 0 & i \neq j \\ i! & i = j \end{cases}. \tag{83}$$

It follows that the expected value is zero for all but the 0^{th} polynomial:

$$\mathbb{E}[H_i(x)] = \int p(x)H_i(x)H_0(x)dx = \begin{cases} 0 & ,i \neq j \\ 1 & ,i = 0 \end{cases}. \quad (84)$$

Further, we conclude that, for $[x_1, \ldots, x_n]^T \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_{n\times n})$,

$$\begin{aligned} \mathbb{E}[H_i(x_k)H_j(x_l)] &= \int p(\mathbf{x})H_i(x_k)H_j(x_l)d\mathbf{x} \\ &= \begin{cases} 0 & ,i \neq j \cup k \neq l \\ 1 & ,i = j = 0 \; \forall k, l \\ i! & ,i = j \cap k = l \end{cases}, \end{aligned} \quad (85)$$

which follows from (83), (84). A simple formula expressing the Hermite polynomials in terms of a random variable $\nu \sim \mathcal{N}(0, 1)$ was given in [30]:

$$H_n(x) = \mathbb{E}\left[(x + \nu\sqrt{-1})^n | x\right]. \quad (86)$$

The first six Hermite polynomials are

$$\begin{aligned} &H_0(x) = 1, \;\; H_2(x) = x^2 - 1, \quad H_4(x) = x^4 - 6x^2 + 3 \\ &H_1(x) = x, \;\; H_3(x) = x^3 - 3x, \;\; H_5(x) = x^5 - 10x^3 + 15x. \end{aligned}$$

Scaling the Hermite polynomials to achieve orthogonality when $\sigma_x \neq 1$ is achieved by dividing the argument with the standard deviation: $H_i(x/\sigma_x)$. Expressions for multivariate Hermitian polynomials are described in [30], offering the possibility to extend the framework to model also terms not represented by the univariate Hermite polynomials, i.e., products on the form $y = \prod_{i=1}^{n} x_i^{\kappa_i}$, for $\kappa_i \in \{0, 1, 2, \ldots\}$.

APPENDIX C
THE SIGMA-POINT SELECTION SCHEME

The uncertainties in the estimate of the mean are described by equation (49). It is zero if $\mathbf{H}\boldsymbol{P}_\theta\mathbf{H}^T$ is invertible and there exists a vector $\boldsymbol{\lambda}$ such that

$$\mathbf{H}^T(\chi)\boldsymbol{\lambda} = \mathbf{w}, \quad (87)$$

with $\mathbf{w} = [1,\ 0, \ldots,\ 0]^T$. As we shall see, the sigma-point selection scheme (4) - (5) always attains the relation (87).

For $x \sim \mathcal{N}(0, 1)$ the sigma-points are $\chi = [0, \sqrt{3}, -\sqrt{3}]$ and the observation matrix for Hermite polynomials up to order 5 is:

$$\begin{aligned} \mathbf{H}^T(\chi) &= [\mathbf{h}(0),\ \mathbf{h}(\sqrt{3}),\ \mathbf{h}(-\sqrt{3})] \\ &= \begin{bmatrix} 1 & 1 & 1 \\ 0 & \sqrt{3} & -\sqrt{3} \\ -1 & 2 & 2 \\ 0 & 0 & 0 \\ 3 & -6 & -6 \\ 0 & -6\sqrt{3} & 6\sqrt{3} \end{bmatrix}. \end{aligned} \quad (88)$$

For $\boldsymbol{\lambda} = [\lambda_0, \lambda_1, \ldots]^T$ to solve equation (87) we see that:

$$\begin{array}{lll} 1: & \sum_{i=0}^{2n} \lambda_i = 1 & \text{(from row one)} \\ 2: & \lambda_i = \lambda_j,\ \forall i, j \neq 0 & \text{(from row two and six)} \\ 3: & \lambda_0 = 4\lambda_i,\ i > 0 & \text{(from row three and five)} \end{array} \quad (89)$$

When the dimensionality of $\mathbf{x}$ increases, no unique elements are added to $\mathbf{H}^T$. When $\mathbf{x} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_{n\times n})$:

$$\mathbf{H}^T(\chi) = \begin{bmatrix} \mathbf{h}(0) & \mathbf{h}(\sqrt{3}) & \mathbf{h}(-\sqrt{3}) & \mathbf{h}(0) & \mathbf{h}(0) & \ldots \\ \mathbf{h}(0) & \mathbf{h}(0) & \mathbf{h}(0) & \mathbf{h}(\sqrt{3}) & \mathbf{h}(-\sqrt{3}) & \ddots \\ \vdots & \vdots & \vdots & \ddots & \ddots & \ddots \end{bmatrix}.$$

The third requirement is therefore adjusted to suit the multi-dimensional case: $\lambda_0 = (6 - 2n)\lambda_i$. Substituting λ_i with w_i, these are exactly the criterions (4) - (5), with $w_0 = 1 - n/3$. The observation matrix associated with the cubature sigma-point selection scheme enjoys the same properties (for $p \leq 3$).

Fredrik Sandblom was born in Mölndal, Sweden in 1979. He received the M.Sc. and Ph.D. degrees from Chalmers University of Technology in Gothenburg, Sweden, in 2004 and 2011 respectively.

Since 2005 he has been with the Volvo group, working with active safety systems, and now holds a position as senior technology specialist. His interests concern object tracking and sensor data fusion; particularly methods for estimating statistical moments and their application to recursive filtering.

Lennart Svensson was born in Älvängen, Sweden in 1976. He received the M.S. degree in electrical engineering in 1999 and the Ph.D. degree in 2004, both from Chalmers University of Technology, Gothenburg, Sweden.

He is currently Associate Professor at the Signal Processing group, again at Chalmers University of Technology. His research interests include Bayesian inference in general, and nonlinear filtering and tracking in particular.

REFERENCES

[1] L. de Menezes, A. Soares, F. Silva, M. Terada, and D. Correia, "A new procedure for assessing the sensitivity of antennas using the unscented transform," *IEEE Trans. Antennas Propag.*, vol. 58, no. 3, pp. 988 –993, March 2010.

[2] C.-L. Su, "Probabilistic load-flow computation using point estimate method," *IEEE Trans. Power Syst.*, vol. 20, no. 4, pp. 1843 – 1851, Nov. 2005.

[3] G. Steiner, H. Zangl, and D. Watzenig, "Generic statistical circuit design based on the unscented transformation and its application to capacitive sensor instrumentation," in *IEEE Int. Conf. on Ind. Technology*, Dec. 2005, pp. 108 –113.

[4] R. E. Kalman, "A new approach to linear filtering and prediction problems," *Trans. of the ASME – Journal of Basic Eng.*, vol. 82, no. Series D, pp. 35–45, 1960.

[5] K. Ito and K. Xiong, "Gaussian filters for non linear filtering problems," *IEEE Trans. Autom. Control*, vol. 45, pp. 910–927, 2000.

[6] M. Norgaard, N. K. Poulsen, and O. Ravn, "New developments in state estimation for nonlinear systems," *Automatica*, vol. 36, no. 11, pp. 1627–38, 2000.

[7] S. Julier and J. Uhlmann, "Unscented filtering and nonlinear estimation," *Proc. IEEE*, vol. 92, no. 3, pp. 401 – 22, 2004.

[8] I. Arasaratnam, S. Haykin, and R. J. Elliott, "Discrete-time nonlinear filtering algorithms using Gauss-Hermite quadrature," *Proc. IEEE*, vol. 95, no. 5, pp. 953–977, 2007.

[9] I. Arasaratnam and S. Haykin, "Cubature Kalman filters," *IEEE Trans, Autom. Control*, vol. 54, no. 6, pp. 1254–1269, 2009.

[10] J. Sarmavuori and S. Sarkka, "Fourier-hermite kalman filter," *IEEE Trans. Autom. Control*, vol. 57, no. 6, pp. 1511 –1515, June 2012.

[11] S. J. Julier, J. K. Uhlmann, and H. F. Durrant-Whyte, "A new approach for filtering nonlinear systems," in *Proc. American Control Conference*, vol. 3, 1995, pp. 1628–1632 vol.3.

[12] U. Lerner, "Hybrid Bayesian Networks for Reasoning About Complex Systems," Ph.D. dissertation, Stanford Univ., 2002.

[13] J. McNamee and F. Stenger, "Construction of fully symmetric numerical integration formulas," *Numerische Mathematik*, vol. 10, no. 4, pp. 327–344, 1967.

[14] Y. Wu, D. Hu, M. Wu, and X. Hu, "A numerical-integration perspective on Gaussian filters," *IEEE Trans. Signal Processing*, vol. 54, no. 8, pp. 2910–21, 2006.

[15] C. Fernández-Prades and J. Vilà-Valls, "Bayesian Nonlinear Filtering Using Quadrature and Cubature Rules Applied to Sensor Data Fusion for Positioning," in *IEEE Int. Conf. on Communications*, 2010, pp. 2–6.

[16] K. Pakki, B. Chandra, G. Da-Wei, and I. Postlethwaite, "Cubature Kalman Filter based Localization and Mapping," in *Preprints of the 18th IFAC World Congress, Milano, Italy*, 2011, pp. 2121–2125.

[17] M. P. Deisenroth, M. F. Huber, and U. D. Hanebeck, "Analytic moment-based Gaussian process filtering," in *Proc. of the 26th Annual Int. Conf. on Machine Learning*, ser. ICML '09. New York, NY, USA: ACM, 2009, pp. 225–232.

[18] J. Ko and D. Fox, "GP-BayesFilters: Bayesian filtering using Gaussian process prediction and observation models," *Autonomous Robots*, vol. 27, pp. 75–90, 2009.

[19] A. O'Hagan, "Bayes-Hermite quadrature," *J. Statist. Plann. Inference*, vol. 29, no. 3, pp. 245–260, Nov. 1991.

[20] C. P. Robert, *The Bayesian Choice*. Springer, 2007.

[21] E. Meijering, "A chronology of interpolation: from ancient astronomy to modern signal and image processing," *Proc. IEEE*, vol. 90, no. 3, pp. 319 –342, Mar 2002.

[22] S. M. Kay, *Fundamentals of statistical signal processing - estimation theory*. Prentice-Hall, Inc., 1993.

[23] A. Gelman, J. B. Carlin, H. S. Stern, and D. B. Rubin, *Bayesian Data Analysis, Second Edition*, 2nd ed. Chapman and Hall/CRC, Jul. 2003, ch. 2.9.

[24] J. Uhlmann, "Simultaneous map building and localization for real time applications," Transfer thesis, Univ. Oxford, Oxford, U.K., 1994.

[25] Y. Bar-Shalom, X. Rong Li, and T. Kirubarajan, *Estimation with Applications to Tracking and Navigation*. John Wiley & Sons, Inc., 2001.

[26] B. Ristic, S. Arulampalam, and N. Gordon, *Beyond the Kalman filter: particle filters for tracking applications*. Artech House, 2004, ch. 6.

[27] S. Kullback and R. Leibler, "On information and sufficiency," *The Annals of Mathematical Statistics*, pp. 79–86, 1951.

[28] A. R. Runnalls, "Kullback-Leibler Approach to Gaussian Mixture Reduction," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 43, no. 3, pp. 989–999, Jul. 2007.

[29] X. R. Li, Z. Zhao, and V. P. Jilkov, "Practical measures and test for credibility of an estimator," in *Proc. Workshop on Estimation, Tracking, and Fusion*, May 2001, pp. 481–495.

[30] C. Withers, "A simple expression for the multivariate Hermite polynomials," *Statistics & Probability Letters*, vol. 47, no. 2, pp. 165–169, Apr. 2000.