

ON NONLINEAR TRANSFORMATIONS OF STOCHASTIC VARIABLES AND ITS APPLICATION TO NONLINEAR FILTERING

Fredrik Gustafsson and Gustaf Hendeby

Department of Electrical Engineering
Linköping University
SE-581 83 Linköping, Sweden
{fredrik, hendebey}@isy.liu.se

ABSTRACT

A class of nonlinear transformation-based filters (NLTF) for state estimation is proposed. The nonlinear transformations that can be used include first (TT1) and second (TT2) order *Taylor expansions*, the *unscented transformation* (UT), and the *Monte Carlo transformation* (MCT) approximation. The *unscented Kalman filter* (UKF) is by construction a special case, but also nonstandard implementations of the *Kalman filter* (KF) and the *extended Kalman filter* (EKF) are included, where there are no explicit Riccati equations.

The theoretical properties of these mappings are important for the performance of the NLTF. TT2 does by definition take care of the bias and covariance of the second order term that is neglected in the TT1 based EKF. The UT computes this bias term accurately, but the covariance is correct only for scalar state vectors. This result is demonstrated with a simple example and a general theorem, which explicitly shows the difference between TT1, TT2, UT, and MCT.

Index Terms— unscented transform, nonlinear transformation, extended Kalman filtering, nonlinear filtering, Kalman filter

1. INTRODUCTION

The *unscented transformation* (UT) [1–3] has received considerable attention during the last decade. Its main idea is to compute approximations of the first and second moment of a nonlinear transformation $z = g(x)$. Its main application is in nonlinear filtering to improve the *extended Kalman filter* (EKF) [4–6] leading to the *unscented Kalman filter* (UKF).

Many successful applications of the UKF have been reported. One important case occurs in target tracking, where the UT is successful in converting range and bearing in radar measurements into Cartesian coordinates, see Table 2 and Fig. 1. In this case, the UT provides a good estimate of the first two moments, and the measurement update in UKF shows an obvious improvement over EKF when the relative range and bearing uncertainty is large.

Four approximative transformations are compared in Tables 1 and 2, and will be discussed in the paper:

TT1: First order Taylor expansion leading to Gauss’ approximation formula. TT1 applied to nonlinear filtering leads to the standard EKF, in the sequel called EKF1.

TT2: Second order Taylor expansion, which compensates the mean and covariance with the quadratic second order term [4, 6, 7]. TT2 applied to nonlinear filtering leads to EKF2 [8].

The authors gratefully acknowledge fundings from SSF (Swedish Foundation for Strategic Research) Strategic Research Center MOVIII, and VR (the Swedish Research Council) project Sensor Informatics.

UT: The unscented transformation [1], with a standard ‘std’ and a modified formulation here denoted ‘mod’ with more degrees of freedom. This leads to UKF.

MCT: The *Monte Carlo transformation* (MCT) approach, which in the limit should compute correct moments.

Section 2 explains these approximations in some more detail. A first contribution is to show that UKF attains its claimed properties only for mappings $g(x)$ for a scalar x . Generally, the covariance will be biased compared to TT2.

It is frequently stated in literature that the UT computes correct first and second order moments when the transformed random variable, x , is Gaussian. The following quote is a typical statement from [3]:

“These sample points [used in the UT] completely capture the true mean and covariance of the GRV [Gaussian random variable], and when propagated through the *true* nonlinear system, captures the posterior mean and covariance accurately to the 3rd order (Taylor series expansion) for *any* non linearity.”

It is easy to show by example that this statement is false. Table 1 provides one revealing example, and it is more generally shown in Theorem 1.

A general advantage of UKF compared to EKF is that only function evaluations are needed, therefore Jacobians and Hessians of the nonlinear functions do not have to be available, or even exist. That is, general mappings such as smoothed table lookups or function calls are allowed. As a second, minor, contribution in Section 3, is to point out that both EKF1 and EKF2 can be implemented with only function evaluations. The numerical values in Tables 1 and 2 are computed this way with general-purpose transformation implementations.

A more important third contribution is the introduction of a new class of *nonlinear-transformation-based filters* (NLTF) in Section 5, where TT1, TT2, UT and MCT can be applied independently in the time and measurement update of the KF.

2. NONLINEAR TRANSFORMATIONS REVISITED

This section details the different transformations considered in this paper.

2.1. Taylor Expansion

Consider a general nonlinear transformation and its second order Taylor expansion

$$z = g(x) = g(\mu_x) + g'(\mu_x)(x - \mu_x) + \underbrace{\left[\frac{1}{2}(x - \mu_x)^T g''(\xi)(x - \mu_x) \right]}_{r(x; \mu_x, g''(\xi))}, \quad (1)$$

where n_x is the dimension of the vector $x \in \mathbb{R}^{n_x}$, and $z \in \mathbb{R}^{n_z}$. The notation $[v_i]_i$ is used to denote a vector in which element i is v_i . Analogously, the notation $[m_{ij}]_{ij}$ will be used to denote the matrix where the (i, j) element is m_{ij} .

2.2. Summary of Approximative Transformations

To summarize, the following options are available for the transformation of $x \sim \mathcal{N}(\mu_x, P_x)$ using g . The result is a Gaussian approximation $z \sim \mathcal{N}(\mu_z, P_z)$.

TT1: First order Taylor approximation:

$$\mu_z = g(\mu_x) \quad P_z = g'(\mu_x)P_x(g'(\mu_x))^T \quad (2)$$

TT2: Second order Taylor approximation:

$$\mu_z = g(\mu_x) + \frac{1}{2}[\text{tr}(g''(\mu_x)P_x)]_i \quad (3a)$$

$$P_z = g'(\mu_x)P_x(g'(\mu_x))^T + \frac{1}{2}[\text{tr}(P_x g''_i(\mu_x) P_x g''_j(\mu_x))]_{ij} \quad (3b)$$

UT: Unscented transform approximation: First define, u_i and σ_i from the *singular value decomposition* (SVD) of the covariance matrix P_x ,

$$P_x = U\Sigma U^T = \sum_{i=1}^{n_x} \sigma_i^2 u_i u_i^T,$$

and then let

$$\begin{aligned} x^{(0)} &= \mu_x, & x^{(\pm i)} &= \mu_x \pm \sqrt{n_x + \lambda} \sigma_i u_i, \\ \omega^{(0)} &= \frac{\lambda}{n_x + \lambda}, & \omega^{(\pm i)} &= \frac{1}{2(n_x + \lambda)}, \end{aligned}$$

where $i = 1, \dots, n_x$. Let $z^{(i)} = g(x^{(i)})$, and apply

$$\mu_z = \sum_{i=-n_x}^{n_x} \omega^{(i)} z^{(i)}, \quad (5a)$$

$$\begin{aligned} P_z &= \sum_{i=-n_x}^{n_x} \omega^{(i)} (z^{(i)} - \mu_z)(z^{(i)} - \mu_z)^T \\ &\quad + (1 - \alpha^2 + \beta)(z^{(0)} - \mu_z)(z^{(0)} - \mu_z)^T, \quad (5b) \end{aligned}$$

where $\omega^{(0)} + (1 - \alpha^2 + \beta)$ is often denoted $\omega_c^{(0)}$ and used to make the notation more compact for the covariance matrix expression. This is the ‘mod’ version of the UT, to get the ‘std’ version remove the last term in (5b), *i.e.*, $\omega^{(0)} = \omega_c^{(0)}$.

MCT: Monte Carlo Transformation:

$$\begin{aligned} x^{(i)} &\sim \mathcal{N}(\mu_x, P_x), \quad i = 1, \dots, N, \\ z^{(i)} &= g(x^{(i)}), \end{aligned}$$

$$\mu_z = \frac{1}{N} \sum_{i=1}^N z^{(i)}, \quad (6a)$$

$$P_z = \frac{1}{N-1} \sum_{i=1}^N (z^{(i)} - \mu_z)(z^{(i)} - \mu_z)^T. \quad (6b)$$

The design parameters of UT have the same notation as in UKF literature (*e.g.*, [3]):

- λ is defined by $\lambda = \alpha^2(n_x + \kappa) - n_x$.
- α controls the spread of the sigma points and is suggested to be approximately 10^{-3} .
- β compensates for the distribution, and should be chosen as $\beta = 2$ when x is Gaussian.
- κ is usually chosen to zero.
- $\omega^{(0)} = 1 - \frac{n_x}{3}$ for UT (std) when x is Gaussian.

Note that $n_x + \lambda = \alpha^2 n_x$ when $\kappa = 0$, and that for $n_x + \lambda \rightarrow 0^+$ the central weight $\omega^{(0)} \rightarrow -\infty$. Furthermore, $\sum_i \omega^{(i)} = 1$.

In summary, TT1 is a computationally cheap approximation, TT2 recovers the first two moments if the gradient and Hessian are available (for Gaussian distributions and quadratic functions TT2 is completely correct, otherwise often a good approximation), the MCT approach is asymptotically correct, and that the UT is a fairly good compromise between TT2 and MCT, that improves computational complexity to MCT and the need for prior knowledge to TT2.

2.3. Nonlinear Filtering

Basically, the nonlinear filters in literature are based on the following:

- EKF in its classical formulation is based on the Kalman filter recursions using the constant and linear terms in (1). This is the EKF1 algorithm. EKF1 works well as long as the rest term is small. Small here relates both to the state estimation error and the degree of nonlinearity of g . As a rule of thumb, the rest term is negligible if either
 - the model is almost linear,
 - the *signal-to-noise ratio* (SNR) is high, in which case the estimation error can be considered sufficiently small.
- The EKF1 is often still useful if *dithering* is used to mitigate the effect of linearization errors. That is, the noise covariances in the state-space model can be increased by the MSE contribution of the mean (bias) and covariance of the second order Taylor term. This is part of the inevitable tuning process of Kalman filters.
- The second order compensated EKF, referred to as EKF2, approximates the rest term $r(x; \mu_x, g''(\xi))$ with $r(x; \mu_x, g''(\mu_x))$, and compensates for the mean and variance of this term. This works well if g'' varies little over the principal support of x .
- UKF estimates the first moments of the nonlinear transformation in (1), without explicitly computing, or even assuming existence, of any derivatives of g .

There are several links and interpretations connecting UKF and EKF as will be pointed out in Section 4.

Table 1. Nonlinear approximations of $x^T x$ for $x \sim \mathcal{N}(0_n, I_{n \times n})$. The theoretical distribution is $\chi^2(n)$ with mean n and variance $2n$. The mean and variance are below summarized as a Gaussian distribution. 10000 Monte Carlo simulations. $\omega^{(0)} = 1 - \frac{n}{3}$ for UT (std), and $\alpha = 10^{-3}$, $\beta = 2$, and $\kappa = 0$ for UT (mod).

n	TT1	TT2	UT (std)	UT (mod)	MCT
1	$\mathcal{N}(0, 0)$	$\mathcal{N}(1, 2)$	$\mathcal{N}(1, 2)$	$\mathcal{N}(1, 2)$	$\mathcal{N}(1.0, 2.2)$
2	$\mathcal{N}(0, 0)$	$\mathcal{N}(2, 4)$	$\mathcal{N}(2, 2)$	$\mathcal{N}(2, 8)$	$\mathcal{N}(2.0, 4.1)$
3	$\mathcal{N}(0, 0)$	$\mathcal{N}(3, 6)$	$\mathcal{N}(3, 0)$	$\mathcal{N}(3, 18)$	$\mathcal{N}(3.0, 6.3)$
4	$\mathcal{N}(0, 0)$	$\mathcal{N}(4, 8)$	$\mathcal{N}(4, -4)$	$\mathcal{N}(4, 32)$	$\mathcal{N}(4.0, 8.4)$
5	$\mathcal{N}(0, 0)$	$\mathcal{N}(5, 10)$	$\mathcal{N}(5, -10)$	$\mathcal{N}(5, 50)$	$\mathcal{N}(5.18, 10.4)$
n	$\mathcal{N}(0, 0)$	$\mathcal{N}(n, 2n)$	$\mathcal{N}(n, (3-n)n)$	$\mathcal{N}(n, 2n^2)$	—

3. NUMERIC TAYLOR TRANSFORMATIONS

It is a trivial fact that the Jacobian and Hessian in TT1 and TT2, respectively, can both be computed using numerical methods. Nevertheless, this fact is seldom explicitly stated in literature. It is worth stressing that both $g'_i(x)$ and $g''_i(x)$ are computed using numerical methods in Tables 1 and 2. That is, only function evaluations of the nonlinear function $g(x)$ are assumed to be available.

4. ANALYSIS OF THE UNSCENTED TRANSFORM

In this section the UT will be analyzed and expressions for the resulting mean and covariance are given and interpreted in the limit as the sigma points approach the center point. The results are exemplified numerically.

4.1. Main Result

Theorem 1 (First and second moments of UT). *Consider a nonlinear mapping of the random stochastic variable x , with mean μ_x and covariance P_x , to z , $g : \mathbb{R}^{n_x} \mapsto \mathbb{R}^{n_z}$. The UT yields mean μ_z and covariance P_z asymptotically as the sigma points in UT tend to the mean, i.e., $\lambda \rightarrow -n_x^+$, given by*

$$\mu_z^{\text{UT}} = g(\mu_x) + \frac{1}{2} [\text{tr}(g''_i P_x)]_i, \quad (7a)$$

$$P_z^{\text{UT}} = g'(\mu_x) P_x (g'(\mu_x))^T + \frac{(\beta - \alpha^2)}{4} [\text{tr}(P_x g''_i(\mu_x)) \text{tr}(P_x g''_j(\mu_x))]_{ij} \quad (7b)$$

Furthermore, $\mu^{\text{UT}} = \mu^{\text{TT2}}$ for all n_x , α , β , and κ , whereas $P_z^{\text{TT2}} = P_z^{\text{UT}}$ is only guaranteed to be true if $\beta - \alpha^2 = 2$ and $n_x = 1$.

In general, the covariances of TT2 and UT are different. Note that the trace in (3b) turns into a product of two traces in (7b), and generally $\text{tr}(AB) \neq \text{tr}(A) \text{tr}(B)$ unless A or B is scalar. The reason for the difference is that the UT cannot express the mixed second order derivatives needed for the TT2 compensation term without increasing the number of sigma points. The result of this approximation depends on the transformation and must be analyzed for the case at hand.

Proof. A proof of this theorem can be found in the technical report [9]. \square

4.2. Numerical Illustrations

The first illustration of the transformation is to approximate the distribution of $x^T x$ when x is a white Gaussian stochastic vector. The results are given in Table 1. In this case, $P_x = I$, $g'(0) = 0$, and $g''(0) = I$, and the asymptotic UT result (given by Theorem 1) is

$$m_z = n \quad P_z = (\beta - \alpha^2)n^2.$$

Note especially how poor the TT1 and UT (std) approximations are, and that neither of the two UT versions gives the correct approximation for this quadratic transformation.

The second illustration is the transformation of range, x_1 , and bearing, x_2 , to Cartesian coordinates, z_1, z_2 . That is, $z_1 = x_1 \cos x_2$ and $z_2 = x_1 \sin x_2$. Numerical results for different bearings are presented in Table 2.

For this case (numeric values are for $x_1 = 20$, $x_2 = \frac{\pi}{4}$),

$$g' = \begin{pmatrix} \cos x_2 & -x_1 \sin x_2 \\ \sin x_2 & x_1 \cos x_2 \end{pmatrix} = \frac{\sqrt{2}}{2} \begin{pmatrix} 1 & -20 \\ 1 & 20 \end{pmatrix}$$

$$g''_1 = \begin{pmatrix} 0 & -\sin x_2 \\ -\sin x_2 & x_1 \cos x_2 \end{pmatrix} = \frac{\sqrt{2}}{2} \begin{pmatrix} 0 & -1 \\ -1 & -20 \end{pmatrix}$$

$$g''_2 = \begin{pmatrix} 0 & \cos x_2 \\ \cos x_2 & -x_1 \sin x_2 \end{pmatrix} = \frac{\sqrt{2}}{2} \begin{pmatrix} 0 & 1 \\ 1 & -20 \end{pmatrix}.$$

It follows from (3) and Theorem 1 that

$$\mu_z^{\text{UT}} = \begin{pmatrix} 13.4 \\ 13.4 \end{pmatrix} = \mu_z^{\text{TT2}}$$

$$P_z^{\text{UT}} = \frac{\beta - \alpha^2}{4} \begin{pmatrix} 43.0 & -37.0 \\ -37.0 & 43.0 \end{pmatrix} \neq \frac{1}{2} \begin{pmatrix} 43.1 & -37.1 \\ -37.1 & 43.1 \end{pmatrix} = P_z^{\text{TT2}},$$

for the third row in Table 2. The same result is illustrated in Figure 1. Note again how TT1 stands out with poor approximations.

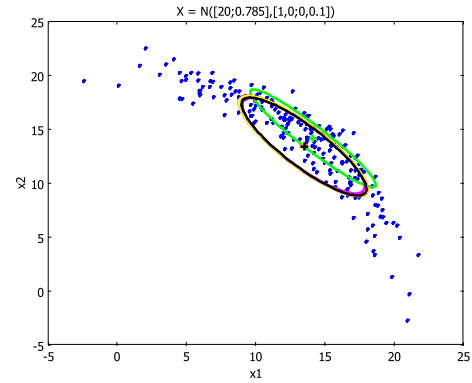


Fig. 1. Distributions on the last row in Table 2. The covariance ellipses for TT2, UT (std and mod), and MCT practically coincide, where as the light-gray (green) TT1 ellipse is clearly different.

5. NONLINEAR-TRANSFORMATION-BASED FILTERS

Consider the nonlinear state space model

$$x_{k+1} = f(x_k, u_k, w_k), \quad (8a)$$

$$y_k = h(x_k, u_k, e_k), \quad (8b)$$

where x_k is the state vector, u_k known input, w_k process noise, y_k measurements, and e_k the measurement noise. The Bayesian nonlinear filtering problem is to compute, or approximate, the posterior distribution $p(x_k | \mathbb{Y}_k)$, where $\mathbb{Y}_k = \{y_i\}_{i=1}^k$. A general algorithm that includes EKF and UKF as special cases is outlined in this section.

Table 2. Nonlinear approximations of the radar observations to Cartesian position. The mean and variance are below summarized as a Gaussian distribution. 10000 Monte Carlo simulations. $\omega^{(0)} = \frac{1}{3}$ for UT (std), and $\alpha = 10^{-3}$, $\beta = 2$, and $\kappa = 0$ for UT (mod).

x	TT1	TT2	UT (std)	UT (mod)	MCT
$\mathcal{N}\left(\begin{pmatrix} 20 \\ 0 \end{pmatrix}, \begin{pmatrix} 1 & 0 \\ 0 & 0.1 \end{pmatrix}\right)$	$\mathcal{N}\left(\begin{pmatrix} 20 \\ 0.0 \end{pmatrix}, \begin{pmatrix} 1.0 & 0.0 \\ 0.0 & 40.0 \end{pmatrix}\right)$	$\mathcal{N}\left(\begin{pmatrix} 19.0 \\ -0.0 \end{pmatrix}, \begin{pmatrix} 3.0 & 0.0 \\ 0.0 & 40.1 \end{pmatrix}\right)$	$\mathcal{N}\left(\begin{pmatrix} 19.0 \\ 0.0 \end{pmatrix}, \begin{pmatrix} 2.9 & 0.0 \\ 0.0 & 36.2 \end{pmatrix}\right)$	$\mathcal{N}\left(\begin{pmatrix} 19.0 \\ 0.0 \end{pmatrix}, \begin{pmatrix} 3.0 & 0.0 \\ 0.0 & 40.0 \end{pmatrix}\right)$	$\mathcal{N}\left(\begin{pmatrix} 19.0 \\ -0.1 \end{pmatrix}, \begin{pmatrix} 2.9 & 0.3 \\ 0.3 & 36.6 \end{pmatrix}\right)$
$\mathcal{N}\left(\begin{pmatrix} 20 \\ \pi/6 \end{pmatrix}, \begin{pmatrix} 1 & 0 \\ 0 & 0.1 \end{pmatrix}\right)$	$\mathcal{N}\left(\begin{pmatrix} 17.3 \\ 10.0 \end{pmatrix}, \begin{pmatrix} 10.7 & -16.9 \\ -16.9 & 30.3 \end{pmatrix}\right)$	$\mathcal{N}\left(\begin{pmatrix} 16.5 \\ 9.5 \end{pmatrix}, \begin{pmatrix} 12.3 & -16.1 \\ -16.1 & 30.8 \end{pmatrix}\right)$	$\mathcal{N}\left(\begin{pmatrix} 16.5 \\ 9.5 \end{pmatrix}, \begin{pmatrix} 11.2 & -14.4 \\ -14.4 & 27.8 \end{pmatrix}\right)$	$\mathcal{N}\left(\begin{pmatrix} 16.5 \\ 9.5 \end{pmatrix}, \begin{pmatrix} 12.3 & -16.0 \\ -16.0 & 30.7 \end{pmatrix}\right)$	$\mathcal{N}\left(\begin{pmatrix} 16.3 \\ 9.8 \end{pmatrix}, \begin{pmatrix} 12.2 & -15.4 \\ -15.4 & 27.9 \end{pmatrix}\right)$
$\mathcal{N}\left(\begin{pmatrix} 20 \\ \pi/4 \end{pmatrix}, \begin{pmatrix} 1 & 0 \\ 0 & 0.1 \end{pmatrix}\right)$	$\mathcal{N}\left(\begin{pmatrix} 14.1 \\ 14.1 \end{pmatrix}, \begin{pmatrix} 20.5 & -19.5 \\ -19.5 & 20.5 \end{pmatrix}\right)$	$\mathcal{N}\left(\begin{pmatrix} 13.4 \\ 13.4 \end{pmatrix}, \begin{pmatrix} 21.5 & -18.5 \\ -18.5 & 21.6 \end{pmatrix}\right)$	$\mathcal{N}\left(\begin{pmatrix} 13.5 \\ 13.5 \end{pmatrix}, \begin{pmatrix} 19.5 & -16.6 \\ -16.6 & 19.5 \end{pmatrix}\right)$	$\mathcal{N}\left(\begin{pmatrix} 13.4 \\ 13.4 \end{pmatrix}, \begin{pmatrix} 21.5 & -18.5 \\ -18.5 & 21.5 \end{pmatrix}\right)$	$\mathcal{N}\left(\begin{pmatrix} 13.3 \\ 13.6 \end{pmatrix}, \begin{pmatrix} 20.3 & -17.1 \\ -17.1 & 20.0 \end{pmatrix}\right)$

5.1. Algorithm

The class of *nonlinear transformation-based filters* (NLTF) discussed here is based on a general algorithm consisting of a time update

$$\begin{pmatrix} x_k \\ v_k \end{pmatrix} \sim \mathcal{N}\left(\begin{pmatrix} \hat{x}_{k|k} \\ 0_{n_v,1} \end{pmatrix}, \begin{pmatrix} P_{k|k} & 0 \\ 0 & Q \end{pmatrix}\right) \quad (9a)$$

$$x_{k+1} = f(x_k, u_k, v_k) \stackrel{\text{approx.}}{\sim} \mathcal{N}(\hat{x}_{k+1|k}, P_{k+1|k}), \quad (9b)$$

and a measurement update

$$\begin{pmatrix} x_k \\ e_k \end{pmatrix} \sim \mathcal{N}\left(\begin{pmatrix} \hat{x}_{k|k-1} \\ 0_{n_e,1} \end{pmatrix}, \begin{pmatrix} P_{k|k-1} & 0 \\ 0 & R \end{pmatrix}\right) \quad (9c)$$

$$\begin{pmatrix} x_k \\ y_k \end{pmatrix} = \begin{pmatrix} x_k \\ h(x_k, u_k, e_k) \end{pmatrix} \stackrel{\text{approx.}}{\sim} \mathcal{N}\left(\begin{pmatrix} \hat{x}_{k|k-1} \\ \hat{y}_{k|k-1} \end{pmatrix}, P_{k|k-1}^{(x,y)}\right). \quad (9d)$$

To complete the measurement update, the Kalman filter gain K_k and measurement update can be computed as

$$K_k = P_{k|k-1}^{xy} (P_{k|k-1}^{yy})^{-1}, \quad (9e)$$

$$\hat{x}_{k|k} = \hat{x}_{k|k-1} + K_k (y_k - \hat{y}_{k|k-1}), \quad (9f)$$

$$P_{k|k} = P_{k|k-1} - K_k P_{k|k-1}^{yy} K_k^T, \quad (9g)$$

where the mean and covariance of the joint distribution of $(x_k^T, y_k^T)^T$ is given by a block partitioning of $P_{k|k-1}^{(x,y)}$.

That is, the posterior distribution $p(x_k | \mathbb{Y}_k)$ is at each stage approximated with a Gaussian distribution. The two approximations above can be computed with TT1, TT2, UT, or MCT, respectively. This yields in total 16 different versions of UKF/EKF.

5.2. Discussion

Both the computational complexity and accuracy of the transformations increase in the same way for TT1, UT, TT2 and MC. As a user guideline, the choice of transform in the time and measurement update, respectively, of the NLTF is a compromise between accuracy and complexity, and it basically only depends on the degree of nonlinearity in the dynamics f and measurement relation h . We point out the following properties of the class of NLTF in (9):

- UKF is obtained using the UT in both time and measurement updates.
- A Riccati-free implementation of EKF1 is obtained using TT1 in both time and measurement updates. The algebraic equality of the EKF1 on standard form, using the Kalman filter Riccati equation for the covariance, can be shown using the (9b), (9d), (9e), and (9g), which implicitly implements the Riccati equation.
- A Riccati-free implementation of the standard KF is obtained in case the model (8) is linear. Again, the Riccati equation is implicitly updated in (9b), (9d), (9e), and (9g).

- If either the dynamics in (8a) or the measurement relation in (8b) is *linear*, then the TT1 update can be used in the corresponding update without approximation. This simple fact appears not to be mentioned in the UKF literature.
- Another fact not explicitly mentioned in literature is the possible inclusion of a Monte Carlo update. For very nonlinear mappings, this might be the most feasible alternative.

6. CONCLUSIONS

The *unscented transformation* (UT) has been analyzed in comparison with the comparable alternative based on a first (TT1) and second (TT2) order Taylor expansions. Both UT and TT2 aim at approximating the mean and covariance of a nonlinear mapping $g(x)$ of a stochastic variable x with second order accuracy. It was first shown by a counter-example that UT fails with its mission in a simple example, and then a general result was stated that the mean term of UT approaches the TT2 mean, but the covariances are only the same for a scalar x in general. A class of *nonlinear-transformation-based filters* (NLTF) was proposed for state estimation in nonlinear systems. This includes the *unscented Kalman filter* (UKF) as a special case, but also Riccati-free versions of the standard *Kalman filter* (KF) and extended Kalman filter (EKF). It also allows arbitrary combinations of UT, TT1, TT2 and a *Monte Carlo transformation* (MCT) step in the measurement and time updates, respectively. All these reported algorithms are available in version 1.1 of the Signals and Systems Lab of Comsol script (<http://www.comsol.se>).

7. REFERENCES

- [1] S. J. Julier and J. K. Uhlmann, "Unscented filtering and nonlinear estimation," *Proc. IEEE*, vol. 92, no. 3, pp. 401–422, Mar. 2004.
- [2] S. J. Julier, J. K. Uhlmann, and H. F. Durrant-Whyte, "A new approach for filtering nonlinear systems," in *Proc. American Contr. Conf.*, Seattle, WA, USA, June 1995, pp. 1628–1632.
- [3] E. A. Wan and R. van der Merwe, "The unscented Kalman filter for nonlinear estimation," in *Proc. IEEE Adapt. Syst. for Signal Processing*, Lake Louise, AB, Canada, Oct. 2000, pp. 153–158.
- [4] B. D. O. Anderson and J. B. Moore, *Optimal Filtering*, Prentice-Hall, Inc, Englewood Cliffs, NJ, 1979.
- [5] A. H. Jazwinski, *Stochastic Processes and Filtering Theory*, vol. 64 of *Mathematics in Science and Engineering*, Academic Press, Inc, 1970.
- [6] Y. Bar-Shalom and T. E. Fortmann, *Tracking and Data Association*, Academic Press, Inc, 1988.
- [7] F. Gustafsson, *Adaptive Filtering and Change Detection*, John Wiley & Sons, Ltd, 2000, 2 reprint.
- [8] M. Athans, R. P. Wishner, and A. Bertolini, "Suboptimal state estimation for continuous-time nonlinear systems from discrete noisy measurements," *IEEE Trans. Autom. Control*, vol. 13, no. 5, pp. 504514, Oct. 1968.
- [9] G. Hendebey and F. Gustafsson, "On nonlinear transformations of stochastic variables and its application to nonlinear filtering," Tech. Rep. LiTH-ISY-R-2828, Dept. Electr. Eng. Linköpings universitet, Sweden, Oct. 2007.