Tronarp, Filip; Hostettler, Roland; Särkkä, Simo

Sigma-Point Filtering for Nonlinear Systems with Non-Additive Heavy-Tailed Noise

Published in:
Proceedings of the 19th International Conference on Information Fusion, FUSION 2016

Published: 05/07/2016

Document Version
Peer reviewed version

Please cite the original version:
Sigma-Point Filtering for Nonlinear Systems with Non-Additive Heavy-Tailed Noise

Filip Tronarp, Roland Hostettler, and Simo Särkkä

This is a post-print of a paper published in *The 19th International Conference on Information Fusion (FUSION)*. When citing this work, you must always cite the original article:


DOI
n/a

Copyright
© 2016 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works.
Sigma-Point Filtering for Nonlinear Systems with Non-Additive Heavy-Tailed Noise

Filip Tronarp, Roland Hostettler, and Simo Särkkä
Department of Electrical Engineering and Automation
Aalto University, Espoo, Finland
E-Mail: {filip.tronarp, roland.hostettler, simo.sarkka}@aalto.fi

Abstract—This paper is concerned with sigma-point methods for filtering in nonlinear systems, where the process and measurement noise are heavy tailed and enter the system non-additively. The problem is approached within the framework of assumed density filtering and the necessary statistics are approximated using sigma-point methods developed for Student’s t-distribution. This leads to UKF/CKF-type of filters for Student’s t-distribution. Four different sigma-point methods are considered that compute exact expectations of polynomials for orders up to 3, 5, 7, and 9, respectively. The resulting algorithms are evaluated that compute exact expectations of polynomials for orders up to 3, 5, 7, and 9, respectively. The resulting algorithms are evaluated that compute exact expectations of polynomials for orders up to 3, 5, 7, and 9, respectively. The resulting algorithms are evaluated.

I. INTRODUCTION

The problem of state estimation is an ubiquitous problem in science and engineering. It arises in numerous sensor fusion applications in aerospace industry [1], smartphones [1], [2], health technology, as well as pedestrian dead-reckoning and indoor navigation [3], [4], among others.

In a mathematical sense, the state estimation problems arise in partially observed systems, where the measurements of a stochastic process are a function of the process and external noise. In this article, such state-space models of the following form are considered:

\[ X_t = f(X_{t-1}, U_t), \]
\[ Y_t = h(X_t, V_t), \tag{1} \]

where \( X_t \in \mathbb{R}^{d_x} \) is a latent process with initial distribution \( X_0 \sim p(x_0) \), \( Y_t \in \mathbb{R}^{d_y} \) is the measurement of \( X_t \), \( U_t \in \mathbb{R}^{d_u} \) is heavy-tailed process noise, \( f : \mathbb{R}^{d_x \times d_u} \rightarrow \mathbb{R}^{d_x} \) is the transition function of the state, \( V_t \in \mathbb{R}^{d_v} \) is a heavy-tailed measurement error, and \( h : \mathbb{R}^{d_x \times d_u} \rightarrow \mathbb{R}^{d_y} \) is the measurement function.

Furthermore it is assumed that \( C[X_{t-1}, U_t] = 0, C[V_t, U_t] = 0, \) and \( C[X_t, V_t] = 0 \), where \( C[\cdot, \cdot] \) denotes the covariance. The state estimation problem is then to find a good estimate (posterior mean) of \( x_t \) given the measurement sequence, \( y_{1:t} \), that is, \( \hat{x}_{t|t} = E[X_t | y_{1:t}] \), and the uncertainty (posterior covariance), \( V[X_t | y_{1:t}] \). Moreover, the notation \( X_{t|t} \triangleq X_t | y_{1:t} \) is used for brevity. Furthermore, if \( X \sim St(\mu, \Sigma, \vartheta) \) then

\[ \Sigma = \frac{\vartheta - 2}{\vartheta} V[X], \tag{2} \]

where it’s important to note that the covariance only exists for \( \vartheta > 2 \).

State estimation in systems with heavy-tailed process and measurement noise has already been studied to some degree. Student’s \( t \) filters are developed for linear systems in [5], where it is also mentioned that their framework can be extended to nonlinear systems in a manner similar to the development of the extended Kalman filter for Gaussian systems. Nonlinear systems with additive noise, Gaussian state transitions, and Student’s \( t \) observation are studied in [6], [7] where a variational Bayes approach is employed. In [8], an approximating smoother for the case with additive Student’s \( t \) noise in the process and measurement is developed using variational Bayes. In the context of Gaussian processes in machine learning, Student’s \( t \)-distributions are also sometimes used [9], [10], and state-space methods for that context have been developed, for example, in [11]. Other methods for robust filtering include, for example \( \ell_1 \) and \( H_\infty \) approaches [12], [13].

In contrast, this paper develops sigma-point methods for filtering in nonlinear systems where the process and measurement noise are heavy-tailed, more specifically Student’s \( t \)-distributed, and enter the system non-additively. The methods use numerical integration algorithms for integrating nonlinear functions with respect to Student’s \( t \)-distributions, which are instances of the fully symmetric integration formulas presented in [14]. In this scenario the distribution of the state can be matched to a Student’s \( t \)-density leading to a nonlinear extension of the Student’s \( t \) filters presented in [5].

II. SIGMA-POINT STUDENT’S T FILTERING

Here the filtering problem is approached from the point of view of statistical linear regression (SLR) [15]–[17], where we reformulate a sigma-point filtering approximation as a linearization of the state-space model with modified noise terms to account for the linearization error. Sigma-point filtering refers to numerical integration schemes that approximate the quantities needed to perform filtering by evaluating the function whose expectation is sought at a set of so called sigma-points; the expectation is then computed by a weighted sum [18]. This section proceeds with a discussion on Student’s \( t \) filtering in the linear case, followed by Student’s \( t \) statistically linearized
filter and ends with Student’s t sigma-point filtering where sigma-point methods for Student’s t-distribution are presented along with algorithms for performing approximate statistically linearized filtering in the Student’s t filtering framework.

A. Student’s t Filtering in Linear Systems

Methods for Student’s t filtering in linear systems have already been developed [5]. Though since it’s important for the development of Student’s t filtering in nonlinear systems the results are presented here as well.

Suppose the system is linear, that is \( f(X_{t-1}, U_t) = A(f)X_{t-1} + B(f)U_t \) and \( h(X_t, V_t) = A(h)X_t + B(h)V_t \), where \( A(f) \in \mathbb{R}^{d_X \times d_X}, B(f) \in \mathbb{R}^{d_X \times d_U}, A(h) \in \mathbb{R}^{d_Y \times d_X}, \) and \( B(h) \in \mathbb{R}^{d_Y \times d_Y} \). Furthermore, assume \( X_{t-1|t-1}, U_t, V_t \) are jointly Student’s t-distributed according to \( \text{St}(\mu_{t-1}, \Sigma_{t-1}, \vartheta_{t-1}) \), where

\[
\begin{align*}
\mu_{t-1} &= \begin{bmatrix} \hat{x}_{t-1|t-1} \\ 0 \\ 0 \end{bmatrix}, \quad \Sigma_{t-1} = \begin{bmatrix} \hat{\Sigma}_{t-1|t-1} & 0 & 0 \\ 0 & \Sigma_{X_t} & 0 \\ 0 & 0 & \Sigma_{V_t} \end{bmatrix}.
\end{align*}
\]

Then the random variables, \( X_{t|t-1}, V_t \) are jointly Student’s t distributed with the parameters

\[
\begin{align*}
\mu_t &= \begin{bmatrix} A(f)\hat{x}_{t-1|t-1} \\ 0 \\ 0 \end{bmatrix}, \\
\Sigma_t &= \begin{bmatrix} A(f)\hat{\Sigma}_{t-1|t-1}(A(f))^T + B(f)\Sigma_{U_t}(B(f))^T & 0 \\ 0 & \Sigma_{X_t} \\ 0 & \Sigma_{V_t} \end{bmatrix}.
\end{align*}
\]

and degrees of freedom parameter, \( \vartheta_t = \vartheta_{t-1} \), unchanged [19]. Going from (3) to (4) corresponds to the prediction step in a Student’s t filter [5]. Now, the joint distribution of \( X_{t|t-1}, Y_t \) is similarly given by a Student’s t-distribution with parameters

\[
\begin{align*}
\mu_t &= \begin{bmatrix} \hat{x}_{t|t-1} \\ 0 \\ 0 \end{bmatrix}, \\
\Sigma_t &= \begin{bmatrix} \hat{\Sigma}_{t|t-1} & 0 & 0 \\ 0 & \Sigma_{X_t} & 0 \\ 0 & 0 & \Sigma_{V_t} \end{bmatrix}.
\end{align*}
\]

Hence, the update step is then given by [19]

\[
\begin{align*}
\vartheta_t &= \vartheta_t + d_Y, \\
\hat{x}_{t|t} &= \hat{x}_{t|t-1} + \hat{\Sigma}_{t|t-1,Y:t-1}^{-1}(y_t - A(h)\hat{x}_{t|t-1}), \\
\hat{\Sigma}_{t|t} &= \vartheta_t + \Delta^2 y \begin{bmatrix} \vartheta_t \end{bmatrix}^{-1} (\hat{\Sigma}_{t|t-1,Y:t-1}^{-1} - \hat{\Sigma}_{t|t-1,Y:t-1}),
\end{align*}
\]

where

\[
\Delta^2 y = (y_t - A(h)\hat{x}_{t|t-1})^T \Sigma_{X_t}^{-1}(y_t - A(h)\hat{x}_{t|t-1}).
\]

This provides an exact filter that converges to the ordinary Kalman filter since the degrees of freedom parameter, \( \vartheta_t \to \infty \) as \( t \to \infty \). Though, it was proposed in [5] that assuming \( X_{t-1|t-1} \) and \( U_t \) are marginally Student’s t, the prediction step can be carried out by approximating their joint density by a Student’s t with degrees of freedom parameter \( \min(\vartheta_{t-1}, \vartheta_{U_t}) \) and rescaled matrices \( \hat{\Sigma}_{t-1|t-1} \) and \( \Sigma_{U_t} \) in order to preserve the covariances of the marginal distributions. The update is then carried out by assuming \( X_{t|t-1} \) and \( V_t \) are marginally Student’s t and their joint distribution is approximated in the same manner with degrees of freedom parameter \( \min(\vartheta_V, \min(\vartheta_{t-1}, \vartheta_{U_t})) \) and the matrices \( \hat{\Sigma}_{t|t-1} \) and \( \Sigma_{V_t} \) are rescaled to preserve covariances.

B. Statistically Linearized Student’s t Filter

Since both the Gaussian distribution and the Student’s t-distribution are closed under linear transformation the nonlinear filtering problem in either case is suitably solved by linearizing the transition function, \( f(X, U) \), as well as the measurement function, \( h(X, V) \). Under the assumption that \( X_{t-1}, U_t, V_t \mid y_{1:t-1} \) are jointly Gaussian, the ordinary Kalman update may be employed. Though one can also assume that \( X_{t-1}, U_t, V_t \mid y_{1:t-1} \) are jointly Student’s t which enables the Student’s t prediction and update.

Assume that \( X \) and \( W \) are uncorrelated, with \( \mathbb{E}[X] = \mu_X \), \( \mathbb{E}[W] = 0 \), \( \mathbb{C}[X, X] = \Sigma_X \), and \( \mathbb{C}[W, W] = \Sigma_W \). Then, for an arbitrary nonlinear function \( g(X, W) \), approximations of the following form are sought:

\[
g(X, W) \approx AX + BW + D + E,
\]

where \( A \in \mathbb{R}^{d_X \times d_X}, B \in \mathbb{R}^{d_X \times d_U}, D \in \mathbb{R}^d, E \in \mathbb{R}^d, \) and \( E \) is a zero-mean random variable with covariance matrix \( \Sigma_E \).

Note that if \( g = f \) then \( W = U \) and \( d = d_X \), and if \( g = h \) then \( W = V \) and \( d = d_Y \). The statistical linearization regression (SLR) solution to this problem is given as [15]–[17]

\[
\begin{align*}
A &= \mathbb{C}[g, X] \mathbb{V}[X]^{-1}, \\
B &= \mathbb{C}[g, W] \mathbb{V}[W]^{-1}, \\
D &= \mathbb{E}[g] - \mathbb{A} \mathbb{E}[X], \\
\Sigma_E &= \mathbb{V}[E] = \mathbb{V}[g] - \mathbb{A} \mathbb{V}[X] \mathbb{A}^T - \mathbb{B} \mathbb{V}[W] \mathbb{B}^T.
\end{align*}
\]

In the Gaussian case this can be interpreted as minimizing the Kullback-Leibler divergence of the approximate joint density of \( g(X, W) \) and \( X \) implied by Equations (8a) to (8d) and the true joint density (see [17]).

The linear Student’s t filter equations can be applied provided that the quantities in Equations (8a) and (8d) are available and the additional assumption that

\[
X_{t-1|t-1}, U_t, V_t, E_t(f), E_t(h) \mid y_{1:t-1} \sim \text{St}(\mu_{t-1}, \Sigma_{t-1}, \vartheta_{t-1}),
\]

where

\[
\begin{align*}
\mu_{t-1} &= \begin{bmatrix} \hat{x}_{t-1|t-1} \\ 0 \\ 0 \end{bmatrix}, \\
\Sigma_{t-1} &= \begin{bmatrix} \frac{\vartheta_{t-1} - 2}{\vartheta_{t-1}} \text{blkdiag} \left( \mathbb{V}[X_{t-1|t-1}], \mathbb{V}[U_t], \ldots, \mathbb{V}[V_t], \mathbb{V}[E_t(f)], \mathbb{V}[E_t(h)] \right) \end{bmatrix}.
\end{align*}
\]
The prediction step can then be carried out according to the following:

\[
\begin{bmatrix}
X_{t|t-1} \\
V_t \\
E_t^{(h)}
\end{bmatrix} =
\begin{bmatrix}
A_t(f) & B_t(f) & 0 & I & 0 \\
0 & 0 & I & 0 & 0 \\
0 & 0 & 0 & 0 & I
\end{bmatrix}
\begin{bmatrix}
X_{t-1|t-1} \\
U_t \\
V_t \\
E_t^{(f)} \\
E_t^{(h)}
\end{bmatrix} +
\begin{bmatrix}
D_t^{(f)} \\
0 \\
0 \\
0 \\
0
\end{bmatrix}.
\]

\[X_{t|t-1}, V_t \text{ and } E_t^{(h)} \text{ are now jointly Student's t-distributed with parameters}
\]

\[\mu_t^i = \begin{bmatrix} \bar{x}_{t|t-1} \\ 0 \\ 0 \end{bmatrix} \text{ and } \Sigma_t^i = \begin{bmatrix} \Sigma_{t|t-1} & 0 & 0 \\ 0 & \Sigma_U & 0 \\ 0 & 0 & \Sigma_{E_t^{(h)}} \end{bmatrix} \] (9)

The joint distribution of \(X_{t|t-1}, Y_t\) is then readily retrieved by the following linear transformation

\[
\begin{bmatrix}
X_{t|t-1} \\
Y_t
\end{bmatrix} =
\begin{bmatrix}
1 & 0 & 0 \\
A_t^{(h)} & B_t^{(h)} & 1
\end{bmatrix}
\begin{bmatrix}
X_{t|t-1} \\
V_t \\
E_t^{(h)}
\end{bmatrix} +
\begin{bmatrix}
0 \\
D_t^{(h)}
\end{bmatrix}. \] (10)

where the parameters of the joint distribution are given by

\[
\mu_t = \begin{bmatrix} \bar{x}_{t|t-1} \\ E[h(X_{t|t-1}, V_t)] \end{bmatrix}, \]

\[
\Sigma_t = \begin{bmatrix}
\Sigma_{t|t-1} & \Sigma_{t|t-1} \bar{A}_t^{(h)} \Sigma_{t|t-1}^{(h)} & \Sigma_{t|t-1} \bar{A}_t^{(h)} \Sigma_{t|t-1}^{(h)} \\
\Sigma_{t|t-1} \bar{A}_t^{(h)} \Sigma_{t|t-1}^{(h)} & \Sigma_{t|t-1}^{(h)} & \Sigma_{t|t-1}^{(h)} \\
\Sigma_{t|t-1} \bar{A}_t^{(h)} \Sigma_{t|t-1}^{(h)} & \Sigma_{t|t-1}^{(h)} & \Sigma_{t|t-1}^{(h)}
\end{bmatrix} \] (12)

Finally, the parameters of the distribution of the state are then readily updated by equation (6).

C. Student’s t Sigma-Point Filter

The expectations present in Equations (8a) to (8d) are generally not tractable. Though, in the framework of Gaussian filtering, they can be approximated by, for example, the use of numerical integration methods [14] that give the correct integrals for polynomials of up to order 3, 5, 7, or 9 [20, 21] or by using various other sigma-point methods [18]. However, the integration methods developed in [14] are not only applicable to Gaussian distributions, but for any fully symmetric distribution. As a matter of fact, the Student’s t-distribution is also fully symmetric, among other distributions.

The sigma-point method for computing the expectation of a non-linear function, \(\tilde{g}(X)\) with respect to Student’s t-distribution takes the following form

\[
E[\tilde{g}(X)] = \int \tilde{g}(x) \text{St}(x; \mu_X, \Sigma_X, \vartheta_X) dx
\]

\[
= \int \tilde{g}(\mu_X + (\Sigma_X)^{-\frac{1}{2}} x') \text{St}(x'; 0, 1, \vartheta_X) dx'
\]

\[
\approx \sum w_j \tilde{g}(\mu_X + (\Sigma_X)^{-\frac{1}{2}} \chi_j). \] (13)

Student’s t sigma-point method of degree 3 is given by the following weights [14]

\[w_j = \begin{cases}
\frac{\kappa}{d_X + \kappa} & j = 0 \\
\frac{d_X}{2(d_X + \kappa)} & j = 1, \ldots, 2d_X
\end{cases} \] (14)

and the sigma-points

\[\chi_j = \begin{cases}
0 & j = 0 \\
se_j & j = 1, \ldots, d_X \\
-\kappa \vartheta_j & j = d_X + 1, \ldots, 2d_X
\end{cases} \] (15)

where

\[s = \left(\frac{\vartheta_X}{\vartheta_X - 2(d_X + \kappa)}\right)^\frac{1}{2}. \] (16)

The unit vectors, \(e_j\), denote basis vectors from the canonical basis and the quantity, \(\kappa\), is a free parameter in the algorithm. Clearly, Student’s t sigma-point method of degree 3 is exactly the same as the unscaled unscented transform [22]. The factor \(\frac{\vartheta_X}{\vartheta_X - 2}\) in the coordinate scaling is due to the fact that the covariance is \(V[X'] = \frac{\vartheta_X}{\vartheta_X - 2} \). That is, if the integral in equation (13) would have been transformed into an expectation with regards to the variable

\[X'' = \left(\frac{\vartheta_X}{\vartheta_X - 2\Sigma_X}\right)^\frac{1}{2} (x - \mu_X), \] (17)

the unscaled unscented transform would have been recovered.

For Student’s sigma point method of degree 5 the weights are computed as [14]

\[w_j = \begin{cases}
1 - d_X \left(\frac{d_X}{4}\right)^2 & j = 0 \\
\frac{1}{2} \left(\frac{d_X}{4}\right)^2 (I_{4j} - (d_X - 1)I_{22}) & j = 1, \ldots, 2d_X \\
\frac{1}{4} \left(\frac{d_X}{4}\right)^2 I_{22} & j \in J_{[s,s]}
\end{cases} \] (18)

where

\[J_{[s,s]} = \left\{2d_X + 1, \ldots, 2d_X + \frac{2d_X!}{(d_X - 2)!}\right\} \]

and the sigma points are given by

\[\chi_j = \begin{cases}
0 & j = 0 \\
se_j & j = 1, \ldots, d_X \\
-\kappa \vartheta_j & j = d_X + 1, \ldots, 2d_X
\end{cases} \] (19)

and \(S_{[s,s]}^j\) is the \(j\)-th element of the set of points \(S_{[s,s]}\) generated by all possible combinations of assigning the pairs \((s, s), (-s, s), (s, -s), \) and \((-s, -s)\) to the coordinate indices.
of a vector of length $d_X$ such that no duplicate vectors are generated. The quantities $I_2$, $I_4$, $I_{22}$, and $s$ are given by

$$I_2 = \frac{\vartheta_X}{\vartheta_X - 2},$$

$$I_4 = \frac{3\vartheta_X^2}{(\vartheta_X - 2)(\vartheta_X - 4)},$$

$$I_{22} = \frac{\vartheta_X^2}{(\vartheta_X - 2)(\vartheta_X - 4)},$$

$$s = \left(\frac{I_2}{I_4}\right)^{\frac{1}{2}}.$$  

The formulas for generating sigma-points and their corresponding weights are a lot more involved for methods of higher degrees and will not be listed here, instead, refer to [14]. Some useful formulas for computing the integrals in [14] when the weight function is a Student’s t density are provided in [23].

Algorithm 1 Student’s t Sigma-Point Predictor I

**Input:** $\hat{x}_t|t, \hat{\Sigma}_t|t, \hat{\Sigma}_{U_{t+1}}, \vartheta_t$

**Output:** $\hat{x}_{t+1}|t, \hat{\Sigma}_{t+1|t}, \vartheta_{t+1}$

Approximate $E[f(X_{t+1|t}, U_{t+1})]$ and $V[f(X_{t+1|t}, U_{t+1})]$, with a Student’s t sigma-point method.

$$\hat{x}_{t+1|t} \leftarrow \mathcal{E}[f(X_{t+1|t}, U_{t+1})]$$

$$\hat{\Sigma}_{t+1|t} \leftarrow \frac{\vartheta_t - 2}{\vartheta_t} V[f(X_{t+1|t}, U_{t+1})]$$

Algorithm 2 Student’s t Sigma-Point Predictor II

**Input:** $\hat{x}_t|t, \hat{\Sigma}_t|t, \hat{\Sigma}_{U_{t+1}}, \vartheta_t, \vartheta_{U_{t+1}}$

**Output:** $\hat{x}_{t+1|t}, \hat{\Sigma}_{t+1|t}, \vartheta_{t+1}$

Rescale $\hat{\Sigma}_{t|t}, \hat{\Sigma}_{U_{t+1}}, \hat{\Sigma}_{V_{t+1}}$ so that the covariances are the same after degree of freedom change.

$$\hat{x}_{t+1|t} \leftarrow \mathcal{E}[f(X_{t+1|t}, U_{t+1})]$$

$$\hat{\Sigma}_{t+1|t} \leftarrow \frac{\vartheta_t - 2}{\vartheta_t} V[f(X_{t+1|t}, U_{t+1})]$$

Algorithm 3 Student’s t Sigma-Point Updater

**Input:** $y_{t+1}, \hat{x}_{t+1|t}, \hat{\Sigma}_{t+1|t}, \hat{\Sigma}_{V_{t+1}}, \vartheta_t$

**Output:** $\hat{x}_{t+1|t+1}, \hat{\Sigma}_{t+1|t+1}, \vartheta_{t+1}$

Approximate $E[Y_{t+1}], \mathcal{V}[Y_{t+1}]$, and $C[X_{t+1|t}, Y_{t+1}]$ with a Student’s t sigma-point method.

$$z_{t+1} \leftarrow y_{t+1} - \mathcal{E}[Y_{t+1}]$$

$$\Sigma_{Y_{t+1}} \leftarrow \frac{\vartheta_t - 2}{\vartheta_t} \mathcal{V}[Y_{t+1}]$$

$$K_{t+1} \leftarrow \frac{\vartheta_t - 2}{\vartheta_t} C[X_{t+1|t}, Y_{t+1}] \Sigma_{Y_{t+1}}^{-1}$$

$$\Delta_{y,t+1} \leftarrow z_{t+1} \Sigma_{V_{t+1}}^{-1} - z_{t+1}$$

$$\tilde{v}_{t+1} \leftarrow \vartheta_t + \Delta_{y,t+1}$$

$$\hat{x}_{t+1|t+1} \leftarrow \hat{x}_{t+1|t} + \tilde{v}_{t+1}$$

$$\hat{\Sigma}_{t+1|t+1} \leftarrow \frac{\vartheta_t + \Delta_{y,t+1}^2}{\vartheta_t + \Delta_{y,t+1}} (\hat{\Sigma}_{t+1|t} - K_{t+1} \Sigma_{Y_{t+1}} K_{t+1}^T)$$

The algorithm for the Student’s t filter in the case where it converges to the unscented Kalman filter is constructed by cycling through the predictor in Algorithm 1 and the updater in Algorithm 3. A Student’s t filter with preserved heavy tail behavior is then retrieved by employing Algorithm 2 as a predictor instead. The filter resulting from Algorithm 1 converges to the UKF hence its only advantage is better suppression of outliers prior to asymptotics kicking in. On the other hand the filter resulting from Algorithm 2 will preserve low degrees of freedom indefinitely and is therefore more suitable in the case where outliers are expected at any
time during the filtering procedure. Note that the approach for preserved heavy tail filtering here differs from the one in [5] though only when \( \dot{\vartheta}_{t_i} > \dot{\vartheta}_{V_i} \). Nevertheless if it is desired to do predictions with high degrees of freedom and updates with low degrees of freedom only small adjustments to Algorithm 2 and Algorithm 3 are required to achieve this. It is also important to note that since the covariance update in Algorithm 3 depends on the squared Mahalanobis distance between measurement and predicted measurement, the filter may behave differently than the UKF even for fairly high values of the degrees of freedom parameter. To clarify, the filters’ behaviors are only similar when \( \Delta t_{y,t+1} \approx d_y \). That is, if the squared Mahalanobis distance between measurement and measurement mean significantly deviates from this relationship, then there is either a downscaling or an upscaling of the filter covariance in comparison to the UKF.

III. EXPERIMENTAL RESULTS

The practical applicability of the proposed Student’s t sigma-point filter is evaluated using a simulated example first and then in a pedestrian dead-reckoning (PDR) application using an inertial measurement unit (IMU) where high peaks in the measured IMU signals are a common problem. In both examples, the sigma-point filter is compared to the standard UKF as well as the extendet Student’s t filter [5].

A. Simulated Example

The following system is considered
\[
\begin{bmatrix}
X_{t+1}^{(1)} \\
X_{t+1}^{(2)}
\end{bmatrix} = \begin{bmatrix}
1 - \frac{0.1}{1+||X_t||} & \frac{1}{1+||X_t||} \\
1 & 1
\end{bmatrix} \begin{bmatrix}
X_t^{(1)} \\
X_t^{(2)}
\end{bmatrix} + U_{t+1}
\]
\[Y_{t+1} = \begin{bmatrix}(1 + V_{t+1}^{(1)})X_t^{(1)} \\
(1 + V_{t+1}^{(2)})X_t^{(2)}
\end{bmatrix},
\]
where the noise processes are governed by the following probability laws,
\[U_{t+1} \sim 0.95\mathcal{N}(u; 0, 0.01I) + 0.05\mathcal{N}(u; 0, 5I)
\]
\[V_{t+1} \sim 0.9\mathcal{N}(v; 0, 0.01I) + 0.1\mathcal{N}(v; 0, 5I).
\]

The system is simulated 5000 times for 250 time steps. Student’s t sigma-point filter is compared against the unscented Kalman filter and the extended Student’s t filter. The sigma-point filters used the scheme of degree 3 and the free parameter was set to \( \kappa = -1 \). The degrees of freedom parameter was set to \( \vartheta = 4 \) for Student’s t filters. The performance of the filters are compared in terms of execution time (ET), mean norm error (MeNE) and maximum norm error (MaNE). Where mean norm error and max norm error refers to the time average and the maximum value of the sequence \( \{||\hat{x}_t - x_t||\}_{t=1}^{250} \), respectively. \( x_t \) denote the realised value of \( X_t \) and \( \hat{x}_t \) refers to the filter estimate of \( x_t \). In order to evaluate Student’s t filters for some moderately large degrees of freedom parameter the preceeding experiment is run again but with \( \vartheta = 50 \). The simulations were carried out in Matlab on a computer with an Intel Xeon E3-1231 v3 CPU 4 \( \times \) 3.4 GHz and 16 GB of RAM.

B. Pedestrian Dead-Reckoning Example

PDR has many different applications in localization and tracking, for example for tracking the movements of firefighters in and around accident sites. The predominant approach to solving this problem is to use strapdown inertial navigation using IMUs, attached to, among others, the users legs or embedded in their shoes [24]. Due to the nature of the IMU, an often encountered problem is high peaks in the measurement signals during the step-down phase of the gait cycle [4]. These spikes can cause errors in the PDR system as they violate the Gaussian assumption of the measurement and process noise. Hence, a filter based on heavy-tailed noise ought to be more robust in these situations.

1) Dynamic Model and Experiment Setup: PDR commonly makes use of indirect Kalman filtering [3], that is, the measured accelerations and rotational velocities are treated as control inputs to the system and then directly integrated. This, however, has the disadvantage that sensor biases and noise cause the integrator to drift. In order to mitigate that, known conditions of the gait cycle are used to generate pseudo-measurements of the integration error. One such state in particular is the stationary phase where the foot is still on the ground, that is, its velocity is zero. Comparing this known velocity to the integrated velocity yields the so called zero-velocity updates (ZUPT) that can be used to adjust the integrator. The discretized integrator is then given by
\[
\begin{bmatrix}
p_{t+1} \\
v_{t+1} \\
q_{t+1}
\end{bmatrix} = \begin{bmatrix}
I_3 & \Delta tI_3 & 0 \\
0 & I_3 & 0 \\
0 & 0 & \Omega(\omega_t^M)
\end{bmatrix} \begin{bmatrix}
p_t \\
v_t \\
q_t
\end{bmatrix} + \frac{(\Delta t)^2}{2} \begin{bmatrix}
\delta I_3 \\
\delta v_t \\
\delta q_t
\end{bmatrix} a_t
\]

TABLE I

<table>
<thead>
<tr>
<th>MeNE</th>
<th>UKF</th>
<th>ESTF</th>
<th>SPSTF</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.5%-th percentile</td>
<td>6.2890</td>
<td>4.1578</td>
<td>4.2166</td>
</tr>
<tr>
<td>50%-th percentile</td>
<td>12.7680</td>
<td>8.6331</td>
<td>8.7770</td>
</tr>
<tr>
<td>97.5%-th percentile</td>
<td>23.6030</td>
<td>39.9242</td>
<td>19.8164</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>MaNE</th>
<th>UKF</th>
<th>ESTF</th>
<th>SPSTF</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.5%-th percentile</td>
<td>41.1887</td>
<td>58.3000</td>
<td>67.9382</td>
</tr>
<tr>
<td>50%-th percentile</td>
<td>66.7930</td>
<td>189.7000</td>
<td>212.7570</td>
</tr>
<tr>
<td>97.5%-th percentile</td>
<td>162.7787</td>
<td>3569.4000</td>
<td>639.4494</td>
</tr>
</tbody>
</table>

TABLE II

<table>
<thead>
<tr>
<th>MeNE</th>
<th>UKF</th>
<th>ESTF</th>
<th>SPSTF</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.5%-th percentile</td>
<td>6.3708</td>
<td>4.7380</td>
<td>4.7355</td>
</tr>
<tr>
<td>50%-th percentile</td>
<td>12.8018</td>
<td>9.2261</td>
<td>9.2109</td>
</tr>
<tr>
<td>97.5%-th percentile</td>
<td>24.4848</td>
<td>26.2806</td>
<td>19.7414</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>MaNE</th>
<th>UKF</th>
<th>ESTF</th>
<th>SPSTF</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.5%-th percentile</td>
<td>40.7023</td>
<td>49.4582</td>
<td>49.4700</td>
</tr>
<tr>
<td>50%-th percentile</td>
<td>66.5971</td>
<td>133.3444</td>
<td>133.4799</td>
</tr>
<tr>
<td>97.5%-th percentile</td>
<td>153.1950</td>
<td>708.7380</td>
<td>452.3309</td>
</tr>
</tbody>
</table>
where $p_i$ is the position of the person in the reference coordinate frame at time instant $t$ and $v_i$ its velocity. $q_i$ is the rotation quaternon from the IMU coordinate frame to the reference coordinate frame and $\Delta t$ is the time since the last integrator update. $a_t^M$ and $\omega_t^M$ denote the acceleration and gyroscope measurements both in the IMU coordinate frame, respectively, while $a_t$ denotes the acceleration in the reference coordinate frame and is given by

$$a_t = q_t \circ (a_t^M - g) \circ q_t^*.$$

Here, $\circ$ denotes the quaternion product. Furthermore, the sub-matrix $\Omega(\omega_t^M)$ in (25) is given by

$$\Omega(\omega_t^M) = \cos \left( \frac{\omega_t^M |\Delta t|}{2} \right) I_4 + \begin{bmatrix}
0 & -\omega_t^{M,z} & -\omega_t^{M,y} & -\omega_t^{M,x} \\
-\omega_t^{M,z} & 0 & -\omega_t^{M,x} & \omega_t^{M,y} \\
-\omega_t^{M,y} & -\omega_t^{M,x} & 0 & \omega_t^{M,z} \\
\omega_t^{M,x} & \omega_t^{M,y} & -\omega_t^{M,z} & 0
\end{bmatrix} \sin \left( \frac{\omega_t^M |\Delta t|}{2} \right).$$

Finally, the error measurement can be described as

$$y_{tk} = v_{ref} - v_{tk} + r_{tk} = -v_{tk} + r_{tk}$$

(26)

since the reference velocity during the stationary phase is assumed to be $v_{ref} = 0$. Furthermore $t_k$ denotes the stationary phase’s detection time and $r_{tk}$ is additive measurement noise.

For a more thorough introduction to PDR dynamics and the indirect Kalman filtering methods commonly used therein, please refer to [3] or [4].

The experimental data was gathered as follows. An LG Nexus 5 smartphone was tightly strapped to a test subject’s lower leg just above the ankle. The test subject was then asked to first stand still for about 5 s and then casually walk along a predefined path (a loop starting and ending at the same position) while the IMU data was gathered at a sampling rate of 200 Hz for both, the accelerometer and gyroscope.

The parameters used for evaluation were

$$Q = \begin{bmatrix} 2I_3 & 0 \\ 0 & I_3 \end{bmatrix}$$

and

$$R = 1 \times 10^{-3} I_3$$

for the process noise and the pseudo measurement noise covariances, respectively. The degrees of freedom for the $t$ distributions were chosen to be 3 in order to preserve as heavy tails as possible. Furthermore, the initial position and velocity was set to zero (corresponding to standing still at the origin of the reference coordinate system) and the initial orientation was determined from the stationary data in the beginning of the measurement sequence. Finally, for the detection of the stationary phase, the generalized maximum likelihood ratio test shown in [25] is used, where the noise measurement noise
covariances were set to
\[ \Sigma_a = 1 \times 10^{-3} \text{diag} \left( \begin{bmatrix} 5 & 1 & 6 \end{bmatrix} \right), \]
\[ \Sigma_\omega = 1 \times 10^{-3} \text{diag} \left( \begin{bmatrix} 0.2 & 0.1 & 0.3 \end{bmatrix} \right) \]
for the accelerometer and gyroscope, respectively. The detection threshold was set to \( \gamma = 6 \times 10^3 \) and the test data length was set to \( N = 10 \) samples.

2) Results and Discussion: Fig. 4 illustrates the estimated trajectories for all three filters together with the predefined path. As it can be seen, the estimated trajectories match the predefined path in shape but are slightly off in the y-direction, especially after the third turn (the starting point is \((0, 0)\)). The complete trajectory length is 73.6 m while the estimated trajectory lengths are 84.3 m for the UKF, 93.7 m for the ESTF, and 78.5 m for the SPSTF. Furthermore, the offset at the end of the loop (loop closure error) is 4.6 m for the UKF, 5.4 m for the ESTF, and 3.9 m for the SPSTF. This indicates that the SPSTF is slightly more accurate in both, travelled distance and loop closure error. Also, it is interesting to note that the ESTF actually performs worse than the UKF, suggesting that the linearization approach is inferior to the sigma-point approach in general for this problem.

The individual components of the position and speed estimates are depicted in Fig. 5 and Fig. 6. No major difference between the UKF and the SPSTF can be observed in the position estimates except for the diverging behavior in \( \dot{p}^x(t) \) around \( t = 60 \) s (Fig. 5). Again, the ESTF exhibits the most deviating behavior, especially in the \( \dot{p}^x(t) \) component. More interestingly, an important difference can be seen in the plots for the speed estimates in Fig. 6. Comparing the estimates for the UKF and the SPSTF, it can be seen that the UKF suffers from larger peaks. This is very prominently visible in the plot for \( \dot{v}^y(t) \) around \( t = 20 \) s...40 s. Note that during that time, the direction of motion is primarily in the positive y-direction. Hence, the negative peaks estimated by the UKF are likely not true motions of the foot (moving the foot backwards) but rather caused by the spikes in the acceleration measurements when putting the foot on the ground. Hence, the results indicate that the SPSTF is more robust toward these kind of extreme value measurements since it uses a heavy-tailed distribution as the underlying noise model. Again, the behavior of the ESTF is different from the sigma-point based filters in the sense that it has significant peaks in the \( \dot{v}^x(t) \) and \( \dot{v}^y(t) \) components, which are not present in the sigma-point based filters.

IV. Conclusion

A sigma-point method for filtering nonlinear systems where Student’s t-distributed process and measurement noise enter the system non-additively was developed in an analogous manner to the UKF. Student’s t sigma-point method differs from the Gaussian one in that it prefers to place the sigma-points further away from the origin, reflecting the heavier tail.

The resulting algorithm was compared to an extended Student’s t filter and the UKF in a simulation experiment where Student’s t filters were found to be faster in suppressing large errors in the state estimates in comparison to the UKF though they may momentarily produce very large errors, the extended filter in particular. All the same, Student’s t filters were found to be superior to the UKF in sense of mean norm error. The sigma-point Student’s t filter was also compared to the UKF in a PDR experiment where Student’s t filter was found to outperform the UKF in terms of error in the estimated...
trajectory length, loop-closure error, and it giving more realistic velocity estimates.

Future work will include a more in-depth study of the stability properties of the Student’s t filter in relation to the Kalman filters. That is, to investigate the question whether it is possible to use a Student’s t filter in cases where Kalman filters fail to converge. Another open aspect is the choice of degrees of freedom and how it affects the behavior of the proposed filter. A possible extension is to use an ensemble of filters using different degrees of freedom or considering the degrees of freedom as a random variable itself. Furthermore, investigating marginalized particle filters using Student’s t filters in place of Kalman filters for the marginalized states.

ACKNOWLEDGMENT

Financial support by the Academy of Finland under the grant no. #295080 (CrowdSLAM) is hereby gratefully acknowledged.

REFERENCES


