Available online at www.sciencedirect.com

ScienceDirect

IFAC PapersOnLine 56-2 (2023) 5951–5956

The Simple Solution for Nonlinear State Estimation of Ill-Conditioned Systems: The Normalized Unscented Kalman Filter [1]

Halvor Aarnes Krog*. Johannes Jäschke.*

**Department of Chemical Engineering, Norwegian University of Science and Technology (NTNU), Sem Sælandsvei 4, Kjemiblokk 5, 7491 Trondheim, Norway*
(e-mail: halvor.a.krog@ntnu.no, johannes.jaschke@ntnu.no).

Abstract: An easy-to-implement method for nonlinear state estimation for ill-conditioned systems is proposed. By propagating standard deviations and correlations instead of the covariance in the unscented Kalman filter (UKF), the condition numbers of relevant matrices are reduced. The reduction in the condition number is related to the scaling of the problem. Hence, what we propose is a normalization method that acts as an "auto-scaler". Compared to other methods in state estimation for ill-conditioned systems, our proposed method factors the covariance matrix into physically meaningful statistics which can be used to check for filter divergence online. The method is compared to a standard UKF in a case study and shows a significant reduction in the condition number.

Keywords: State estimation; Normalized UKF; SigmaRho UKF; Ill-conditioned system; square-root filter.

1 INTRODUCTION

The task of a state estimator is to infer the true state of a system, $\boldsymbol{x}(t_k) = \boldsymbol{x}_k$, by combining measurements with a process model. The Kalman filter (KF) is the minimum variance estimator for linear systems. It estimates the mean and covariance of the state distribution for every timestep. For nonlinear systems, approximate solutions such as the Extended KF (EKF) or Unscented KF (UKF) can be used. The UKF is theoretically more accurate than the EKF but can have stability issues when applied to high-dimensional systems (Yuanxin et al., 2006).

It can be difficult to make the KF to work in practice. A standard assumption in the theory is infinite numerical accuracy (no round-off errors in computer implementations). However, the presence of round-off errors in computers can have a significant effect, as reported by Grewal and Andrews (2014, p. 360), who note that "the effects of roundoff may thought to be minor, but overlooking them could be a major blunder". This paper focuses on issues related to numerical accuracy for nonlinear systems, specifically the UKF.

For linear systems, this issue has been addressed by Simon (2006, p. 140) who recommends to use "some form of square-root filtering" to improve the numerical properties of the KF. The idea behind the different forms of square-root filtering is to decompose the covariance matrix into factors which are less sensitive to round-off errors. A square-root filter typically decomposes the covariance matrix into triangular Cholesky factors. See Grewal and Andrews (2014, ch. 7) for a good overview of numerically stable methods for KF implementations for linear systems.

The first square-root filter for the nonlinear UKF (SR-UKF) was of van der Merwe and Wan (2001). It is based on the one-rank Cholesky update procedure. The negative update (downdate) might however destroy the positive definiteness of the Cholesky factor. Kulikova and Kulikov (2020) avoid this issue in the Cholesky downdate by using a J-orthogonal transformation. De Vivo et al. (2017) take another angle: they developed a SR-UKF based on the Joseph stabilized version of the covariance matrix update formula. The Joseph stabilized form is known from linear KF theory to be more stable and robust than the normal covariance update formula (Simon, 2006, p. 129).

All the above-mentioned forms of square-root filters use matrix decompositions which have no immediate physical meaning. Grewal and Kain (2010) noted that one can propagate standard deviations, $\boldsymbol{\sigma}$, and correlation matrices, $\boldsymbol{\rho}$, instead of covariance matrices. This gives desirable numerical properties since the coefficients in the correlation matrix are bounded between $(-1,1]$, and the standard deviation is the square-root of variance. They named their filter the *sigmaRho KF*, and it is restricted to linear systems. Grewal and Andrews (2014, p. 416) briefly mention that the sigmaRho may be well suited for the UKF, but we have not found any mentions or reports of application about it in the literature.

This paper reports the first implementation of such a filter to nonlinear systems. To avoid confusion with *sigma-points* in the UKF literature, we call the filter the *Normalized UKF* (NUKF), as the correlation matrix is a normalized covariance matrix. Our proposed method has the following desirable properties:

[1] The authors acknowledge financial support from the Norwegian Research Council, SUBPRO, grant number: 237893. Code for the case study is available on: https://github.com/Process-Optimization-and-Control/normalizedUKF

10.1016/j.ifacol.2023.10.626

1. Condition numbers of relevant matrices are reduced. The reduction is case dependent as the method works as an "auto-scaler".

2. The method is easy to implement and require only standard numerical libraries. E.g. *cholupdate()* required for the SR-UKF in van der Merwe and Wan (2001) exists in Matlab, but not in Scipy (as of version 1.9.1).

3. Any matrix square-root can be used. We are not restricted to use Cholesky decompositions.

From a user perspective, our proposed method has further advantages, namely that i) it is simple to understand decomposition of covariance into standard deviation and correlation and ii) standard deviations and correlations have physical interpretation, while e.g. Cholesky factors do not.

The article is structured as follows: In section 2 we present the background theory about unscented transformation (UT), the standard UKF and a discussion about the choice of matrix square-roots for the UT. Section 3 contains the proposed methods: normalized matrix square-roots, normalized UT (NUT) and the NUKF. Section 4 contains two case studies: one where we show the increased numerical accuracy of the normalized Cholesky factorization, and the second a state estimation problem from the literature where we demonstrate the NUKF.

2 BACKGROUND

2.1 Unscented transformation (UT)

The UT approximates the mean and covariance of a random variable $\boldsymbol{y} = \boldsymbol{g}(\boldsymbol{x})$. The random variable $\boldsymbol{x} \in \mathbb{R}^{n_x}$ has a known mean and covariance, $\hat{\boldsymbol{x}}, \boldsymbol{P}_{\boldsymbol{x}}$, the function $\boldsymbol{g}: \mathbb{R}^{n_x} \to \mathbb{R}^{n_y}$ and the random variable $\boldsymbol{y} \in \mathbb{R}^{n_y}$. The UT is based on deterministically sampling a minimal set of points with corresponding weights, $\boldsymbol{\chi}^{(i)}, W_m^{(i)}, W_c^{(i)}$, such that the weighted points capture the mean and covariance of $\boldsymbol{x}$. The sample-points $\boldsymbol{\chi}^{(i)} \in \mathbb{R}^{n_x}$ are called sigma-points and the number of sigma points are typically $2n_x + 1$. There are various methods to generate sigma-points; here we consider the scaled UT described in Merwe and Wan (2004, ch. 3.2.2). The sigma-points and weights are calculated as below, where the indices i goes from $i = 1, \ldots, n_x$.

$$\begin{aligned} &\tilde{\boldsymbol{\chi}} = \sqrt{(n_x + \lambda)\boldsymbol{P}_{\boldsymbol{x}}} = \sqrt{n_x + \lambda}\sqrt{\boldsymbol{P}_{\boldsymbol{x}}} \\ &\boldsymbol{\chi}^{(0)} = \hat{\boldsymbol{x}} \qquad\qquad W_m^{(0)} = \lambda/(n_x + \lambda) \\ &\boldsymbol{\chi}^{(i)} = \hat{\boldsymbol{x}} + (\tilde{\boldsymbol{\chi}})_i \qquad W_m^{(i)} = 1/2(n_x + \lambda) \\ &\boldsymbol{\chi}^{(i+n_x)} = \hat{\boldsymbol{x}} - (\tilde{\boldsymbol{\chi}})_i \qquad W_m^{(i+n_x)} = 1/2(n_x + \lambda) \end{aligned} \tag{1}$$

Here, $(\tilde{\boldsymbol{\chi}})_i$ means the i-th column in $\tilde{\boldsymbol{\chi}}$, $W_c^{(j)} = W_m^{(j)}, j = 1, \ldots, 2n_x$ and $W_c^{(0)} = W_m^{(0)} + 1 - \alpha^2 + \beta$ for this algorithm. Here, $\lambda = \alpha^2(n_x + \kappa) - n_x$ and α, β, κ are tuning/scaling parameters. See Merwe and Wan (2004, ch. 3.2.2) for a more detailed discussion of these parameters.

The mean and covariance of $\boldsymbol{x}$ is retained by the sigma-points and weights as shown below:

$$\mathbb{E}[\boldsymbol{x}] = \hat{\boldsymbol{x}} = \sum_{i=0}^{2n_x} W_m^{(i)} \boldsymbol{\chi}^{(i)} \tag{2}$$

$$\begin{aligned} \boldsymbol{P}_{\boldsymbol{x}} &= \sum_{i=0}^{2n_x} W_c^{(i)} \left(\boldsymbol{\chi}^{(i)} - \hat{\boldsymbol{x}}\right)(\ldots)^T \\ &= W_c^{(1)} \sum_{i=1}^{n_x} \left(\sqrt{n_x + \lambda}\left(\sqrt{\boldsymbol{P}_{\boldsymbol{x}}}\right)_i\right)(\ldots)^T \\ &\qquad + \left(-\sqrt{n_x + \lambda}\left(\sqrt{\boldsymbol{P}_{\boldsymbol{x}}}\right)_i\right)(\ldots)^T \\ &= 2(n_x + \lambda) W_c^{(1)} \sum_{i=1}^{n_x} \left(\sqrt{\boldsymbol{P}_{\boldsymbol{x}}}\right)_i \left(\sqrt{\boldsymbol{P}_{\boldsymbol{x}}}\right)_i^T \\ &= \sum_{i=1}^{n_x} \left(\sqrt{\boldsymbol{P}_{\boldsymbol{x}}}\right)_i \left(\sqrt{\boldsymbol{P}_{\boldsymbol{x}}}\right)_i^T = \sqrt{\boldsymbol{P}_{\boldsymbol{x}}}\left(\sqrt{\boldsymbol{P}_{\boldsymbol{x}}}\right)^T, \end{aligned} \tag{3}$$

where we have used that i) $\boldsymbol{\chi}^{(0)} = \hat{\boldsymbol{x}}$ ii) $W_c^{(1)} = W_c^{(2)} = \cdots = W_c^{(2n_x)} = 1/(2(n_x + \lambda))$ iii) scalar-matrix multiplication is commutative and iv) computed matrix-matrix multiplication as a sum of outer products. The notation $(\boldsymbol{x})(\ldots)^T$ means $(\boldsymbol{x})(\boldsymbol{x})^T$. The matrix-square root in (3) is further discussed in section 2.3.

The UT approximates the mean and covariance of the output, $\hat{\boldsymbol{y}}, \boldsymbol{P}_{\boldsymbol{y}}$, by a 3rd order Taylor series accuracy given that the distribution $\boldsymbol{x}$ is symmetrical (Simon, 2006). Also, the cross-covariance between $\boldsymbol{x}$ and $\boldsymbol{y}, \boldsymbol{P}_{\boldsymbol{xy}}$, are given by:

$$\boldsymbol{\mathcal{Y}}^{(i)} = \boldsymbol{g}(\boldsymbol{\chi}^{(i)}) \tag{4}$$

$$\mathbb{E}[\boldsymbol{y}] = \hat{\boldsymbol{y}} = \sum_{i=0}^{2n_x} W_m^{(i)} \boldsymbol{\mathcal{Y}}^{(i)} \tag{5}$$

$$\boldsymbol{P}_{\boldsymbol{y}} = \sum_{i=0}^{2n_x} W_c^{(i)} \left(\boldsymbol{\mathcal{Y}}^{(i)} - \hat{\boldsymbol{y}}\right)(\ldots)^T \tag{6}$$

$$\boldsymbol{P}_{\boldsymbol{xy}} = \sum_{i=0}^{2n_x} W_c^{(i)} \left(\boldsymbol{\chi}_k^{(i)} - \hat{\boldsymbol{x}}\right)\left(\boldsymbol{\mathcal{Y}}^{(i)} - \hat{\boldsymbol{y}}\right)^T \tag{7}$$

2.2 The unscented Kalman filter

We briefly present some UKF concepts used in this work. For thorough background theory we refer to Simon (2006). Consider the nonlinear discrete system with additive noise given by:

$$\boldsymbol{x}_{k+1} = \boldsymbol{f}(\boldsymbol{x}_k) + \boldsymbol{w}_k \tag{8}$$

$$\boldsymbol{y}_k = \boldsymbol{h}(\boldsymbol{x}_k) + \boldsymbol{v}_k, \tag{9}$$

where $\boldsymbol{w}_k \sim (\boldsymbol{0}, \boldsymbol{Q}_k)$ is the zero-mean process noise with covariance $\boldsymbol{Q}_k$, and $\boldsymbol{v}_k \sim (\boldsymbol{0}, \boldsymbol{R}_k)$ is the measurement noise. The subscript k denotes the discrete time. The states are given by $\boldsymbol{x} \in \mathbb{R}^{n_x}$ and the measurements are given by $\boldsymbol{y} \in \mathbb{R}^{n_y}$. Noise which enters $\boldsymbol{f}(\cdot)$ or $\boldsymbol{h}(\cdot)$ nonlinearly can be approximated as additive noise by the framework of parametric uncertainty described in Krog and Jäschke (2022).

Let $\hat{\boldsymbol{x}}_k^+, \boldsymbol{P}_k^+$ be the *a posteriori* estimates of the mean and covariance from the UKF, and $\hat{\boldsymbol{x}}_k^-, \boldsymbol{P}_k^-$ be the *a priori* estimates. *A priori* estimates have only used the state equations for predicting the state at time k, while *a posteriori* estimates have also processed the measurements at time k. At every time step k, the estimate of the state and covariance are propagated through (8) by the UT, and the noise covariance $\boldsymbol{Q}_k$ is added to obtain the *a priori* estimates $\hat{\boldsymbol{x}}_k^-$ and $\boldsymbol{P}_k^-$. A new set of sigma points and weights based on $\hat{\boldsymbol{x}}_k^-, \boldsymbol{P}_k^-$ are then made. These points are propagated through (9) to obtain the estimate of the predicted measurement $\hat{\boldsymbol{y}}_k$, its covariance $\boldsymbol{P}_y$ and the cross-covariance $\boldsymbol{P}_{xy}$ between $\boldsymbol{x}_k^-$ and $\boldsymbol{y}_k$, see (4)-(7). The measurement noise $\boldsymbol{R}_k$ is added to $\boldsymbol{P}_y$. The Kalman gain $\boldsymbol{K}_k$ and the posterior estimates $\hat{\boldsymbol{x}}_k^+, \boldsymbol{P}_k^+$ are calculated based on the actual measurement $\boldsymbol{y}_k$:

$$\boldsymbol{K}_k = \boldsymbol{P}_{xy}\boldsymbol{P}_y^{-1} \tag{10}$$

$$\hat{\boldsymbol{x}}_k^+ = \hat{\boldsymbol{x}}_k^- + \boldsymbol{K}_k(\boldsymbol{y}_k - \hat{\boldsymbol{y}}_k) \tag{11}$$

$$\begin{aligned}\boldsymbol{P}_k^+ &= \boldsymbol{P}_k^- - \boldsymbol{K}_k\boldsymbol{P}_y\boldsymbol{K}_k^T \\ &= \boldsymbol{P}_k^- - \boldsymbol{P}_{xy}\boldsymbol{P}_y^{-1}\boldsymbol{P}_{xy}^T,\end{aligned} \tag{12}$$

where in the last equality we have inserted (10) and used that covariance matrices are symmetric; since $\boldsymbol{P_y} = \boldsymbol{P_y^T}$ then also $\boldsymbol{P_y^{-1}} = \left(\boldsymbol{P_y^{-1}}\right)^T = \boldsymbol{P_y^{-T}}$.

2.3 *Discussion of the matrix square-root in the UT*

In (3) we showed how the covariance matrix $\boldsymbol{P_x}$ is retained by the sigma-points and that the matrix square-root $\sqrt{\boldsymbol{P_x}}$ plays a role. One important observation in (3) is the transpose in last equality: we are *not* required to use an actual matrix square-root $\boldsymbol{S_P}$ of $\boldsymbol{P_x} = \boldsymbol{S_P}\boldsymbol{S_P} = \boldsymbol{S_P^2}$, such as the principal matrix square-root. The principal matrix square-root $\boldsymbol{S_P}$ is unique, symmetric and all eigenvalues are positive. Due to the transpose, the Cholesky decomposition $\boldsymbol{P_x} = \boldsymbol{L_P}\boldsymbol{L_P^T}$ is applicable, where $\boldsymbol{L_P}$ is the lower triangular Cholesky factor of the covariance matrix $\boldsymbol{P}$.

In the original article about the UT, Julier et al. (2000) proposed to use Cholesky decomposition since it is a numerically effective and stable method. The developed SR-UKF in literature updates the Cholesky factor in different ways, see e.g. van der Merwe and Wan (2001) and Kulikova and Kulikov (2020). However, Daid et al. (2021) remark from theoretical considerations that the Cholesky decomposition can lead to divergence issues in the Unscented Kalman observer while this is not the case for the principal square-root. Nevertheless, it seems for most practical applications that the UT is quite robust when it comes to the form of square-root, as there are many examples of the success of Cholesky decomposition (Grewal and Andrews, 2014, p. 404).

For flexibility purposes, it is desirable that methods for ill-conditioned systems can use both the Cholesky decomposition and the principal matrix square-root. As we will see in section 3.1, our proposed method accomplishes this.

3 PROPOSED APPROACH FOR ILL-CONDITIONED SYSTEMS – THE NORMALIZED UKF (NUKF)

The aim for this paper is to report on the first implementation of a Normalized UKF. We propagate the standard deviation and correlation matrix of the states instead of the covariance matrix. The covariance matrix can be decomposed as:

$$\boldsymbol{P_x} = \boldsymbol{\sigma_x}\boldsymbol{\rho_x}\boldsymbol{\sigma_x}, \tag{13}$$

where the standard deviation matrix $\boldsymbol{\sigma_x} \in \mathbb{R}^{n_x \times n_x}$ is a diagonal matrix. The correlation matrix $\boldsymbol{\rho_x} \in \mathbb{R}^{n_x \times n_x}$ is the normalized covariance matrix, i.e. the variables in $\boldsymbol{x}$ have been centered and scaled to have a variance of 1. In some sense, the correlation matrix contains the directions of the distribution while the standard deviation matrix contains the scale. Hence, $diag(\boldsymbol{\rho_x}) = \boldsymbol{I}$ and $\boldsymbol{\rho_x}[i,j] \in (-1,1]$. The condition number of the correlation matrix is therefore usually lower than the condition number of the corresponding covariance matrix. This is the reason why it makes sense from a numerical point of view to propagate the correlation- and standard deviation matrix. Note that both matrices are symmetric; $\boldsymbol{\sigma_x} = \boldsymbol{\sigma_x^T}$ and $\boldsymbol{\rho_x} = \boldsymbol{\rho_x^T}$. Since $\boldsymbol{\sigma_x}$ is a diagonal matrix, its inverse is diagonal and cheap to compute: $\boldsymbol{\sigma_x^{-1}}[i,i] = 1/\boldsymbol{\sigma_x}[i,i]$. The standard deviations and correlations can be obtained from the covariance matrix by:

$$\boldsymbol{\sigma_x} = \sqrt{(diag(\boldsymbol{P_x})} \tag{14}$$

$$\boldsymbol{\rho_x} = \boldsymbol{\sigma_x^{-1}}\boldsymbol{P_x}\boldsymbol{\sigma_x^{-1}} \tag{15}$$

The main benefit of the approach is that we obtain lower condition numbers when solving for the matrix square-root and the Kalman gain.

3.1 *The matrix square-root for ill-conditioned covariance matrices*

3.1.1 *Normalized Cholesky decomposition*

Applying the decomposition of covariance into correlation- and standard deviation matrices as in (13), the Cholesky decomposition is equal to:

$$\begin{aligned}\boldsymbol{P_x} &= \boldsymbol{\sigma_x}\boldsymbol{\rho_x}\boldsymbol{\sigma_x} = \boldsymbol{\sigma_x}\boldsymbol{L_\rho}\boldsymbol{L_\rho^T}\boldsymbol{\sigma_x} \\ &= \left(\boldsymbol{\sigma_x}\boldsymbol{L_\rho}\right)\left(\boldsymbol{\sigma_x}\boldsymbol{L_\rho}\right)^T = \boldsymbol{L_P}\boldsymbol{L_P^T},\end{aligned} \tag{16}$$

where we have used that $\boldsymbol{\sigma_x} = \boldsymbol{\sigma_x^T}$ and $(\boldsymbol{AB})^T = \boldsymbol{B}^T\boldsymbol{A}^T$. Hence, the Cholesky factor of a covariance matrix is equal to the Cholesky factor of the correlation matrix multiplied with the diagonal standard deviation matrix. The benefit is the reduced condition number in the correlation matrix compared to the covariance matrix. This scaling-technique is also applied in Grewal and Andrews (2014, p. 414).

3.1.2 *The principal square-root matrix in the UT*

We apply the same methodology for the principal square-root matrix:

$$\begin{aligned}\boldsymbol{P_x} &= \boldsymbol{\sigma_x}\boldsymbol{\rho_x}\boldsymbol{\sigma_x} = (\boldsymbol{\sigma_x}\boldsymbol{S_\rho})(\boldsymbol{\sigma_x}\boldsymbol{S_\rho}) \\ &= \left(\boldsymbol{\sigma_x}\boldsymbol{S_\rho}\right)\left(\boldsymbol{S_\rho^T}\boldsymbol{\sigma_x^T}\right)^T = \left(\boldsymbol{\sigma_x}\boldsymbol{S_\rho}\right)\left(\boldsymbol{S_\rho}\boldsymbol{\sigma_x}\right)^T\end{aligned} \tag{17}$$

These equalities are valid since $\boldsymbol{\sigma_x} = \boldsymbol{\sigma_x^T}$ and $\boldsymbol{S_\rho} = \boldsymbol{S_\rho^T}$. However, we have not obtained the true principal square-root matrix $\boldsymbol{P_x} = \boldsymbol{S_P}\boldsymbol{S_P} = \boldsymbol{S_P^2}$, as that would require that the matrix product $\left(\boldsymbol{\sigma_x}\boldsymbol{S_\rho}\right)$ is symmetric, and that is not the case. However, for the UT we know from (3) that the requirement is $\boldsymbol{P_x} = \sqrt{\boldsymbol{P_x}}\left(\sqrt{\boldsymbol{P_x}}\right)^T$ and not $\boldsymbol{P_x} = \boldsymbol{S_P^2}$. Hence, for the UT we can use $\sqrt{\boldsymbol{P_x}} = \boldsymbol{\sigma_x}\boldsymbol{S_\rho}$ where $\boldsymbol{S_\rho}$ is the principal square-root matrix of the correlation matrix, $\boldsymbol{\rho_x} = \boldsymbol{S_\rho^2}$. We stress that the factor

$(\boldsymbol{\sigma_x S_\rho}) \neq \boldsymbol{S_P}$; it is not a true principal matrix square-root. To the authors knowledge, this scaling-technique has not been used in the literature before.

3.2 The Normalized UT (NUT)

We show how to obtain standard deviations and correlations directly from the UT when there is presence of additive noise. This situation is shown in equation (9), but we omit the subscript k in this section for convenience. The sigma-points of $\boldsymbol{x}$ are generated by the algorithm in e.g. (1), propagated as in (4) and the mean $\hat{\boldsymbol{y}}$ is estimated by (5). We center all the sigma-points around the mean, $\tilde{\boldsymbol{\mathcal{Y}}}^{(i)} = \boldsymbol{\mathcal{Y}}^{(i)} - \hat{\boldsymbol{y}}$, and collect the centralized sigma-points in the matrix $\tilde{\boldsymbol{\mathcal{Y}}}$, defined as:

$$\tilde{\boldsymbol{\mathcal{Y}}} = \left[\tilde{\boldsymbol{\mathcal{Y}}}^{(0)}, \tilde{\boldsymbol{\mathcal{Y}}}^{(1)}, \dots, \tilde{\boldsymbol{\mathcal{Y}}}^{(2n_x)}\right] \in \mathbb{R}^{n_y \times (2n_x+1)} \tag{18}$$

Since matrix-matrix multiplication can be computed as a sum of outer products, the following is true:

$$\begin{aligned}\boldsymbol{P_y} &= \sum_{i=0}^{2n_x} W_c^{(i)} \left(\boldsymbol{\mathcal{Y}}^{(i)} - \hat{\boldsymbol{y}}\right)(\dots)^T + \boldsymbol{R} \\ &= \left(\boldsymbol{W}_{c,n_x} \otimes \tilde{\boldsymbol{\mathcal{Y}}}\right)\tilde{\boldsymbol{\mathcal{Y}}}^T + \boldsymbol{R} \\ &= \tilde{\boldsymbol{\mathcal{Y}}}_{\boldsymbol{W}} \tilde{\boldsymbol{\mathcal{Y}}}^T + \boldsymbol{R},\end{aligned} \tag{19}$$

Where $\boldsymbol{W}_{c,n_x} = \left[W_c^{(0)} \mathbf{1}_{n_y}, \dots, W_c^{(2n_x)} \mathbf{1}_{n_y}\right] \in \mathbb{R}^{n_y \times (2n_x+1)}$. That is, the first column in $\boldsymbol{W}_{c,n_x}$ is filled with $W_c^{(0)}$, the next is filled with $W_c^{(1)}$ etc. The symbol $\otimes$ denotes element-wise multiplication. Implementing (19) as matrix-matrix multiplication instead of a summation as in (6) significantly reduces computation time in numerical libraries such as Numpy (Harris et al., 2020). Although this is straight-forward usage of linear algebra, the authors have not seen (19) being written as matrix-matrix multiplication in literature before.

The standard deviations are the square-root of the diagonal elements of the covariance matrix. Each $\sigma_{y,i}, i = 1, \dots, n_y$, is therefore given by:

$$\sigma_{y,i} = \sqrt{\tilde{\boldsymbol{\mathcal{Y}}}_{\boldsymbol{W}}[i,:]\left(\tilde{\boldsymbol{\mathcal{Y}}}[i,:]\right)^T + \boldsymbol{R}[i,i]}, \tag{20}$$

where $\tilde{\boldsymbol{\mathcal{Y}}}_{\boldsymbol{W}}[i,:] \in \mathbb{R}^{1 \times (2n_x+1)}$ is the i-th row in the matrix $\tilde{\boldsymbol{\mathcal{Y}}}_{\boldsymbol{W}}$. Thus, we can directly obtain the standard deviations this way without calculating the entire covariance matrix. To obtain the correlations, we must first normalize the centered sigma-points and the noise matrix:

$$\tilde{\boldsymbol{\mathcal{Y}}}^{(i)\prime} = \boldsymbol{\sigma_y}^{-1} \tilde{\boldsymbol{\mathcal{Y}}}^{(i)}, \qquad i = 0, \dots, 2n_x \tag{21}$$

$$\boldsymbol{R}' = \boldsymbol{\sigma_y}^{-1} \boldsymbol{R} \boldsymbol{\sigma_y}^{-1}, \tag{22}$$

where $\boldsymbol{\sigma_y}$ is the diagonal matrix of standard deviations, and its inverse is therefore cheap to compute. One potential problem here is if $\boldsymbol{\sigma_y}[i,i] \gg 1$ and $\boldsymbol{R}[i,i] \ll 1$, then $\boldsymbol{R}'[i,i] \ll \boldsymbol{R}[i,i]$ and $\boldsymbol{R}'[i,i] \approx \boldsymbol{0}$. The same issue would however exist in the normal UT as well, and is not worsened by the NUT.

The remaining unique elements of the correlation matrix can now be computed fast and efficiently by exploiting that the diagonal is filled with ones and that the matrix is symmetric:

$$\boldsymbol{\rho_y}[i,j] = \boldsymbol{\rho_y}[j,i] = \begin{cases} 1 \; if \; i = j \\ \tilde{\boldsymbol{\mathcal{Y}}}'_{\boldsymbol{W}}[i,:]\left(\tilde{\boldsymbol{\mathcal{Y}}}'[j,:]\right)^T + \boldsymbol{R}'[i,j] \end{cases} \tag{23}$$

Alternatively, it can be obtained as the matrix-matrix product:

$$\boldsymbol{\rho_y} = \tilde{\boldsymbol{\mathcal{Y}}}'_{\boldsymbol{W}} \tilde{\boldsymbol{\mathcal{Y}}}'^T + \boldsymbol{R}' \tag{24}$$

3.3 The Normalized UKF (NUKF) Algorithm

Our NUKF propagates mean, standard deviations and correlation matrices to remedy ill-conditioned systems. While square-root filters double the numerical precision by propagating $\sqrt{\boldsymbol{P_x}}$, the numerical precision for the normalized UKF is case dependent. We note however that $\sqrt{\boldsymbol{P_x}} = \boldsymbol{\sigma_x}\sqrt{\boldsymbol{\rho_x}}$, so the condition numbers only depend on the structure of $\sqrt{\boldsymbol{\rho_x}}$.

3.3.1 Propagation step

Generate sigma-points $\boldsymbol{\chi}_{k-1}^{+(i)}$ based on the posterior distribution $\boldsymbol{x}_{k-1}^{+} \sim (\hat{\boldsymbol{x}}^{+}, \boldsymbol{\sigma}_{\boldsymbol{x}}^{+}\boldsymbol{\rho}_{\boldsymbol{x}}^{+}\boldsymbol{\sigma}_{\boldsymbol{x}}^{+})_{k-1}$, where the subscript $k-1$ means that all variables inside the parenthesis have that subscript. The matrix square-root is found by the methods of section 3.1. Propagate each sigma-point:

$$\boldsymbol{\chi}_k^{-(i)} = \boldsymbol{f}\left(\boldsymbol{\chi}_{k-1}^{+(i)}\right) \tag{25}$$

The mean, standard deviation and correlation for the prior, $\hat{\boldsymbol{x}}_k^-, \boldsymbol{\sigma}_{\boldsymbol{x}_k}^-, \boldsymbol{\rho}_{\boldsymbol{x}_k}^-$, is found by straight forward application of the NUT in section 3.2 where the additive noise term is the process noise covariance $\boldsymbol{Q}_{k-1}$.

3.3.2 Measurement update step

The predicted measurement, its standard deviation and correlation $\hat{\boldsymbol{y}}_k, \boldsymbol{\sigma}_{\boldsymbol{y}_k}, \boldsymbol{\rho}_{\boldsymbol{y}_k}$ is found by repeating section 3.3.1 but use instead i) the prior distribution $\boldsymbol{x}_k^- \sim (\hat{\boldsymbol{x}}^-, \boldsymbol{\sigma}_{\boldsymbol{x}}^-\boldsymbol{\rho}_{\boldsymbol{x}}^-\boldsymbol{\sigma}_{\boldsymbol{x}}^-)_k$ to generate sigma-points ii) propagate each sigma-point through the measurement equation $\boldsymbol{h}(\cdot)$ and iii) the measurement noise $\boldsymbol{v}_k \sim (\boldsymbol{0}, \boldsymbol{R}_k)$ as additive noise.

The cross-correlation between the states and the measurement are given by:

$$\boldsymbol{\rho}_{\boldsymbol{xy},k} = \sum_{i=0}^{2n_x} W_c^{(i)} \tilde{\boldsymbol{\chi}}_k^{-\prime(i)} \left(\tilde{\boldsymbol{\mathcal{Y}}}_k^{\prime(i)}\right)^T, \tag{26}$$

where the centered and normalized sigma-points are used, see (21). This term can be computed as a matrix product in the spirit of (19). The Kalman gain can be simplified by inserting the decomposition of covariance to standard deviations and correlations:

$$\begin{aligned}\boldsymbol{K}_k &= \left(\boldsymbol{\sigma}_k^- \boldsymbol{\rho}_{\boldsymbol{xy},k} \boldsymbol{\sigma}_{\boldsymbol{y}_k}\right)\left(\boldsymbol{\sigma}_{\boldsymbol{y}_k} \boldsymbol{\rho}_{\boldsymbol{y}_k} \boldsymbol{\sigma}_{\boldsymbol{y}_k}\right)^{-1} \\ &= \boldsymbol{\sigma}_k^- \boldsymbol{\rho}_{\boldsymbol{xy},k} \boldsymbol{\rho}_{\boldsymbol{y}_k}^{-1} \boldsymbol{\sigma}_{\boldsymbol{y}_k}^{-1} \\ &= \boldsymbol{\sigma}_k^- \boldsymbol{K}'_k \boldsymbol{\sigma}_{\boldsymbol{y}_k}^{-1},\end{aligned} \tag{27}$$

Where we have defined the normalized Kalman gain $\boldsymbol{K}'_k = \boldsymbol{\rho}_{\boldsymbol{xy},k} \boldsymbol{\rho}_{\boldsymbol{y}_k}^{-1}$. Inserting into the measurement update equation for the mean, equation (11):

$$\begin{aligned}\hat{\boldsymbol{x}}_k^+ &= \hat{\boldsymbol{x}}_k^- + \boldsymbol{\sigma}_k^- \boldsymbol{\rho}_{\boldsymbol{xy},k} \boldsymbol{\rho}_{\boldsymbol{y}_k}^{-1} \boldsymbol{\sigma}_{\boldsymbol{y}_k}^{-1} (\boldsymbol{y}_k - \hat{\boldsymbol{y}}_k) \\ &= \hat{\boldsymbol{x}}_k^- + \boldsymbol{\sigma}_k^- \boldsymbol{K}'_k \boldsymbol{\sigma}_{\boldsymbol{y}_k}^{-1} (\boldsymbol{y}_k - \hat{\boldsymbol{y}}_k)\end{aligned} \tag{28}$$

From the covariance update formula (12), we know that we need an expression for the term $\boldsymbol{P}_{xy}\boldsymbol{P}_y^{-1}\boldsymbol{P}_{xy}^T$:

$$\begin{aligned}\boldsymbol{P}_{xy}\boldsymbol{P}_y^{-1}\boldsymbol{P}_{xy}^T &= \left(\boldsymbol{\sigma}_k^-\boldsymbol{\rho}_{xy,k}\boldsymbol{\sigma}_{y_k}\right)\left(\boldsymbol{\sigma}_{y_k}\boldsymbol{\rho}_{y_k}\boldsymbol{\sigma}_{y_k}\right)^{-1}\boldsymbol{P}_{xy}^T \\ &= \boldsymbol{\sigma}_k^-\boldsymbol{\rho}_{xy,k}\boldsymbol{\sigma}_{y_k}\boldsymbol{\sigma}_{y_k}^{-1}\boldsymbol{\rho}_{y_k}^{-1}\boldsymbol{\sigma}_{y_k}^{-1}\boldsymbol{P}_{xy}^T \\ &= \boldsymbol{\sigma}_k^-\boldsymbol{\rho}_{xy,k}\boldsymbol{\rho}_{y_k}^{-1}\boldsymbol{\sigma}_{y_k}^{-1}\boldsymbol{P}_{xy}^T \\ &= \left(\boldsymbol{\sigma}_k^-\boldsymbol{\rho}_{xy,k}\boldsymbol{\rho}_{y_k}^{-1}\boldsymbol{\sigma}_{y_k}^{-1}\right)\left(\boldsymbol{\sigma}_k^-\boldsymbol{\rho}_{xy,k}\boldsymbol{\sigma}_{y_k}\right)^T \\ &= \boldsymbol{\sigma}_k^-\boldsymbol{\rho}_{xy,k}\boldsymbol{\rho}_{y_k}^{-1}\boldsymbol{\rho}_{xy,k}^T\boldsymbol{\sigma}_k^-\end{aligned} \tag{29}$$

Inserting into (12) and decomposing the covariance terms gives:

$$\begin{aligned}\boldsymbol{\sigma}_k^+\boldsymbol{\rho}_k^+\boldsymbol{\sigma}_k^+ &= \boldsymbol{P}_k^- - \boldsymbol{\sigma}_k^-\boldsymbol{\rho}_{xy,k}\boldsymbol{\rho}_{y_k}^{-1}\boldsymbol{\rho}_{xy,k}^T\boldsymbol{\sigma}_k^- \\ &= \boldsymbol{\sigma}_k^-\left(\boldsymbol{\rho}_k^- - \boldsymbol{\rho}_{xy,k}\boldsymbol{\rho}_{y_k}^{-1}\boldsymbol{\rho}_{xy,k}^T\right)\boldsymbol{\sigma}_k^- \\ &= \boldsymbol{\sigma}_k^-\left(\boldsymbol{\rho}_k^- - \boldsymbol{K}'_k\boldsymbol{\rho}_{xy,k}^T\right)\boldsymbol{\sigma}_k^- \\ &= \boldsymbol{\sigma}_k^-\left(\boldsymbol{\rho}_k^- - \boldsymbol{K}'_k\boldsymbol{\rho}_{y_k}\boldsymbol{K}_k'^T\right)\boldsymbol{\sigma}_k^- \\ &= \boldsymbol{\sigma}_k^-\breve{\boldsymbol{\rho}}_k^+\boldsymbol{\sigma}_k^-,\end{aligned} \tag{30}$$

which is a measurement update formula similar to the standard UKF, equation (12).

However, the estimated posterior correlation matrix $\breve{\boldsymbol{\rho}}_k^+$ is not a true correlation matrix, as the elements are not in the range -1 to 1. We must therefore normalize the estimated correlation matrix $\breve{\boldsymbol{\rho}}_k^-$ to find the true normalized correlation matrix $\boldsymbol{\sigma}_k^+$. Then, we multiply the "update-factor" $\breve{\boldsymbol{\sigma}}_k$ to the prior standard deviation matrix to obtain the posterior standard deviation.

$$\breve{\boldsymbol{\sigma}}_k[i,i] = \sqrt{\breve{\boldsymbol{\rho}}_k^+[i,i]}, \qquad i = 1,\dots,n_x \tag{31}$$

$$\boldsymbol{\sigma}_k^+ = \boldsymbol{\sigma}_k^-\breve{\boldsymbol{\sigma}}_k \tag{32}$$

$$\boldsymbol{\rho}_k^+ = (\breve{\boldsymbol{\sigma}}_k)^{-1}\breve{\boldsymbol{\rho}}_k^+(\breve{\boldsymbol{\sigma}}_k)^{-1}, \tag{33}$$

where all standard deviation matrices $\boldsymbol{\sigma}_k^-, \boldsymbol{\sigma}_k^+, \breve{\boldsymbol{\sigma}}_k$ are diagonal matrices.

4 CASE STUDIES

We demonstrate first the numerical benefit of the normalized Cholesky decomposition. Then, the benefit of the NUKF is highlighted in a state estimation problem from the literature.

4.1 Normalized Cholesky decomposition

Let $\boldsymbol{\sigma} = diag(10^7, 10^{-7}, 10^{-1})$ and the correlation matrix be:

$$\boldsymbol{\rho} = \begin{bmatrix} 1 & 10^{-1} & 10^{-1} \\ 10^{-1} & 1 & 0 \\ 10^{-1} & 0 & 1 \end{bmatrix} \tag{34}$$

The covariance matrix is $\boldsymbol{P} = \boldsymbol{\sigma\rho\sigma}$. From (16), we know that the Cholesky factor of the covariance matrix can be computed as $\boldsymbol{L_P} = \boldsymbol{\sigma L_\rho}$. Numerically, we see that applying the Cholesky decomposition directly on $\boldsymbol{P}$ gives a slightly different result than calculating the product $\boldsymbol{\sigma L_\rho}$ for this example. The condition number for $\boldsymbol{P}$ is $\kappa(\boldsymbol{P}) \approx 10^{28}$, the condition number for the correlation matrix is $\kappa(\boldsymbol{\rho}) \approx 1.32$ and $\kappa(\boldsymbol{\sigma L_\rho}) \approx 10^{14}$. If we set instead $\boldsymbol{\sigma}[1,1] = 10^2$ and $\boldsymbol{\sigma}[2,2] = 10^{-2}$, the two methods generate the same result and the condition number of the covariance matrix is then $\kappa(\boldsymbol{P}) \approx 10^8$. The reason for the difference is presumably due to round-off errors. The condition number of the normalized Cholesky decomposition indicates that this is the most accurate method.

The Cholesky decomposition was performed by *scipy.linalg.cholesky()* with a scipy version of 1.9.1. The datatype was float64.

4.2 Normalized UKF: falling body case study

We investigate the tracking problem from Julier et al. (2000). A body falls into the atmosphere from a very high altitude and velocity. We want to estimate the altitude $x_1(t)$, velocity $x_2(t)$ and a constant ballistic coefficient $x_3(t)$. As a measurement device, we have a radar which is located at a height a and a horizontal distance M from the falling body.

The dynamics of the system and the measurements are described by:

$$\dot{x}_1(t) = x_2(t) + w_1 \tag{35}$$

$$\dot{x}_2(t) = \frac{1}{2}\rho_0 e^{-\frac{x_1}{k}} x_2^2 x_3 - g + w_2 \tag{36}$$

$$\dot{x}_3(t) = w_3 \tag{37}$$

$$y_1(t_k) = \sqrt{M^2 + (x_1(t_k) - a)^2} + v_1 \tag{38}$$

To highlight the benefits of the proposed approach, we need $n_y > 1$ for condition numbers to make sense. Hence, we assume there is a pressure transmitter on the falling body which measures the atmospheric pressure according to:

$$y_2(t_k) = P_b\left[\frac{T_b + (x_1(t_k) - h_b)L_b}{T_b}\right]^{-\frac{gM}{RL_b}} + v_2 \tag{39}$$

Here, ρ_0 is the air density at sea level, k is a constant relating the air density and altitude, g is the gravitational acceleration, R the universal gas constant and P_b, T_b, L_b are reference parameters for pressure, temperature, and temperature lapse rate at a selected reference height h_b above sea level. Parameter values are $[\rho_0, k, g, M, a, P_b, T_b, h_b, L_b, R] = [105.1\frac{kg\ s^2}{m^4}, 6096m,$ $9.81\frac{m}{s^2}, 30\,480m, 30\,480m, 3.96Pa, 214.65K, 7 \times 10^4 m,$ $-0.002\frac{K}{m}, 8.314\frac{J}{mol\ K}]$.

Process noise realizations are sampled at every time step from the distribution $\boldsymbol{w}_k \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{Q}_k)$ where $\boldsymbol{Q}_k = diag(10^2, 10^2, 10^{-8})$. The measurement noise is coming from $\boldsymbol{v}_k \sim \boldsymbol{\mathcal{N}}(\boldsymbol{0}, \boldsymbol{R}_k)$ where $\boldsymbol{R}_k = diag(10^3, 50)$. A measurement is obtained every 0.5 second and a Runge-Kutta method of order 4 with adaptive step size is used to integrate the process model between every measurement. Initial conditions are $\boldsymbol{x}_0^T = [9.1 \times 10^4 m, -6 \times 10^3 \frac{m}{s}, 6.24 \times 10^{-5}\frac{m^3}{kg \cdot s^2}]$ and the UKF were initialized with a standard deviation of $\boldsymbol{\sigma}_0^+ = \left[10^4 m, 10^3 \frac{m}{s}, 10^{-5} \frac{m^3}{kg\cdot s^2}\right]^T$ and correlation $\boldsymbol{\rho}_0^+ = \boldsymbol{I}_3$.

At every time step, the condition number for relevant matrices were calculated. The condition number for matrices which needs to be inverted gives information about how sensitive the solution $\boldsymbol{x} = \boldsymbol{A}^{-1}\boldsymbol{b}$ for small errors in the value $\boldsymbol{b}$. For the UKF, the relevant matrices are the posterior, prior and measurement correlation (covariance) matrix. We repeated the simulation 100 times to see the effect of different starting point for the filters, $\hat{\boldsymbol{x}}_0^+$, and different realizations of process- and measurement noise. The 100 trajectories are shown in Figure 1. Note the logarithmic scale on the y-axis.

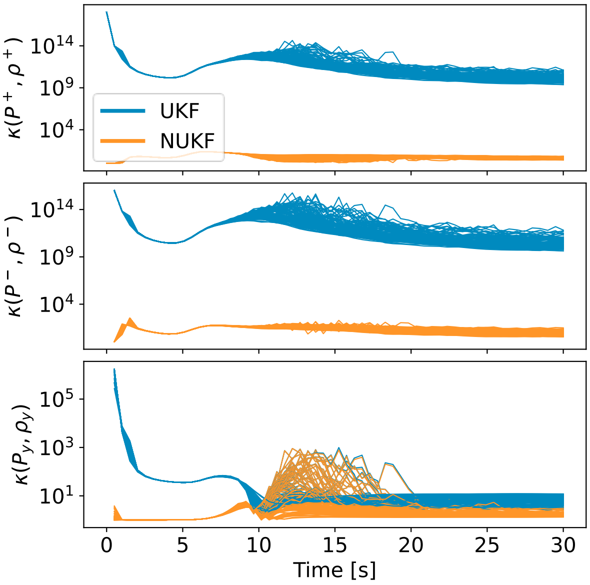

Figure 1: Condition numbers for relevant matrices through time for 100 simulations. The mean value for the condition number (averaged over time and 100 simulations) was $[1.6 \times 10^{16}, 1.8 \times 10^{14}, 2.5 \times 10^4]$ for the standard UKF and $[6.97, 27.1, 8.0]$ for the NUKF.

As expected, the condition numbers for the NUKF are significantly smaller than the standard UKF. The condition numbers imply that numerical issues may be encountered for the UKF but not for the NUKF. This documents that numerical issues due to scaling are automatically and efficiently handled by the proposed method.

The choice of matrix square-root when generating sigma-points did not matter in this case study. The Cholesky decomposition and the principal matrix square-root gave equal results. A figure of the state trajectories and the code is available on the Github repository mentioned on the front page.

5 CONCLUSION AND FUTURE WORK

The proposed NUKF was shown to have improved numerical properties compared to the standard UKF filter as the condition number of relevant matrices were significantly lower. We obtained very similar results for the NUKF and the UKF in our case study, but the condition numbers for the UKF can give a cause for concern. Compared to square-root filters in the literature, the NUKF has the benefit that it propagates physically meaningful factors which are typically used to check for filter divergence. This capability will be subject for future work of the authors, as well as to investigate under which circumstances the difference in condition numbers have a large practical effect.

6 REFERENCES

DAID, A., BUSVELLE, E. & AIDENE, M. 2021. On the convergence of the unscented Kalman filter. *European journal of control,* 57**,** 125-134.

DE VIVO, F., BRANDL, A., BATTIPEDE, M. & GILI, P. 2017. Joseph covariance formula adaptation to Square-Root Sigma-Point Kalman filters. *Nonlinear dynamics,* 88**,** 1969-1986.

GREWAL, M. S. & ANDREWS, A. P. 2014. *Kalman Filtering: Theory and Practice with MATLAB®: Fourth Edition*.

GREWAL, M. S. & KAIN, J. 2010. Kalman Filter Implementation With Improved Numerical Properties. *IEEE transactions on automatic control,* 55**,** 2058-2068.

HARRIS, C. R., MILLMAN, K. J., VAN DER WALT, S. J., GOMMERS, R., VIRTANEN, P., COURNAPEAU, D., WIESER, E., TAYLOR, J., BERG, S., SMITH, N. J., KERN, R., PICUS, M., HOYER, S., VAN KERKWIJK, M. H., BRETT, M., HALDANE, A., DEL RÍO, J. F., WIEBE, M., PETERSON, P., GÉRARD-MARCHANT, P., SHEPPARD, K., REDDY, T., WECKESSER, W., ABBASI, H., GOHLKE, C. & OLIPHANT, T. E. 2020. Array programming with NumPy. *Nature,* 585**,** 357-362.

JULIER, S., UHLMANN, J. & DURRANT-WHYTE, H. F. 2000. A new method for the nonlinear transformation of means and covariances in filters and estimators. *IEEE transactions on automatic control,* 45**,** 477-482.

KROG, H. A. & JÄSCHKE, J. 2022. Systematic Estimation of Noise Statistics for Nonlinear Kalman Filters. *IFAC PapersOnLine,* 55**,** 19-24.

KULIKOVA, M. V. & KULIKOV, G. Y. 2020. The J-Orthogonal Square-Root MATLAB-Based Continuous-Discrete Unscented Kalman Filtering Method. *IFAC PapersOnLine,* 53**,** 4967-4972.

MERWE, R. V. D. & WAN, E. A. 2004. *Sigma-Point Kalman Filters for Probabilistic Inference in Dynamic State-Space Models.* PhD, Oregon Health & Science University.

SIMON, D. 2006. *Optimal state estimation : Kalman, H [infinity] and nonlinear approaches,* Hoboken, N.J., Wiley-Interscience.

VAN DER MERWE, R. & WAN, E. A. The square-root unscented Kalman filter for state and parameter-estimation. 2001 IEEE International Conference on Acoustics, Speech, and Signal Processing. Proceedings, 2001. IEEE, 3461–3464 vol.6.

YUANXIN, W., HU, D., MEIPING, W. & XIAOPING, H. 2006. A Numerical-Integration Perspective on Gaussian Filters. *IEEE transactions on signal processing,* 54**,** 2910-2921.