Available online at www.sciencedirect.com

ScienceDirect

IFAC PapersOnLine 56-2 (2023) 8869–8874

The Weighted Kalman Filter

Damiano Rotondo [1]

[1] *Department of Electrical and Computer Engineering (IDE), University of Stavanger, 4009 Stavanger, Norway*

Abstract: This paper proposes a new version of the Kalman filter, referred to as *weighted Kalman filter* (WKF). In the WKF some recent results on the weighted linearization of nonlinear systems are exploited to incorporate modifications in the equations of the extended Kalman filter (EKF). More specifically, the computation of the Jacobian matrices at the current mean of the estimated state is replaced by the multiple integral over the state space of the Jacobian matrix functions multiplied by a weighting function. Similar modifications are introduced in the equations used to account for the available nonlinear model and compute the so-called *a priori state and output estimates*. The weighting function is chosen to be a multivariable Gaussian function where the generalized variance is selected as proportional to the current covariance matrix of the state estimate. An illustrative example is used to describe the step-by-step derivation of the WKF equations and compare its performance against the EKF in terms of convergence properties and estimation error performance.

Keywords: Kalman filter, linearization, nonlinear systems

1. INTRODUCTION

State estimation is one of the most relevant engineering problems, in which a dynamic system known as state observer is used to fuse some a priori knowledge (the model) with available real-time data (the measurements) so that an accurate estimate of the real state of another system is obtained. It is well known that when the state-space representation of the system under observation is linear, then the minimum mean squared error estimate is obtained when the state observer is designed/selected as the widely celebrated Kalman filter.

Albeit a loss of optimality and convergence properties, the extended Kalman filter (EKF) has managed to become the most widely applied state observer for nonlinear systems, and it is nowadays considered the *de facto* standard in many applications, such as navigation systems and GPS [Huang et al., 2009, Loron and Laliberte, 1993, Böhler et al., 2021, Narayanan et al., 2020]. Due to the limitations of the EKF, some research over the last few decades has focused on providing alternative estimators that perform better than the EKF in some situations.

For instance, Julier and Uhlmann [1997] proposed the unscented Kalman filter (UKF), in which a set of discretely sampled points is used to parameterize the mean and covariance, which are later transformed using an unscented transform. Some adaptive Kalman filters have been applied with success to mitigate the impact of modeling uncertainty [Myers and Tapley, 1976, Huang et al., 2020]. In Chui et al. [1990], a modified EKF (MEKF) was introduced by improving the linearization procedure through a parallel computational scheme, whereas Ahmed and Radaideh [1994] proposed to interconnect the EKF with a noise-free model of the system. Glielmo et al. [1999] proposed an algorithm consisting of a bank of interlaced extended Kalman filters (IEKF), each of which estimates a part of the state, while considering the remaining parts as known time-varying parameters whose values are evaluated by other filters at previous steps. Germani et al. [2005] proposed a polynomial version of the extended Kalman filter (PEKF) which was a generalization of the traditional EKF. On the other hand, a new class of Kalman filters based on the Fourier-Hermite series expansion of the nonlinear functions (FHKF) was proposed in Sarmavuori and Sarkka [2011]. The research aiming at improving the Kalman filter is far from over, with recent papers discussing its initialization [Zhao and Huang, 2020], or equipping it with machine learning abilities [Shen et al., 2020, Liu and Guo, 2021, Xin and Shi, 2021].

1.1 Theoretical background: weighted linearization

The Taylor-based linearization technique plays a central role in the EKF, as the main idea behind it is to approximate the nonlinear state-space representation with a linear model obtained at the current mean of the state estimate. The recent paper Rotondo [2022] has proposed a *weighted* generalization of the linearization technique in which the Jacobian matrices are multiplied by a weighting function and then this product is integrated over the entire state and input spaces to obtain the state-space matrices of the linear representation. Linear quadratic regulator (LQR) design was used to show the potential advantages of the weighted linearization when applied to controller design, such as better performance when the initial state is far from the point of linearization, or an enlargement of the region of attraction. The main ideas behind the weighted linearization are summarized hereafter.

Let us consider the following discrete-time nonlinear system:

$$\begin{cases} x(k+1) = f\left(x(k), u(k)\right) \\ y(k) = h\left(x(k), u(k)\right) \end{cases} \tag{1}$$

where $k \in \mathbb{R}$ stands for the sample, $x \in \mathbb{R}^n$ is the state vector, $u \in \mathbb{R}^m$ is the input vector, $y \in \mathbb{R}^p$ is the output vector, and $f : \mathbb{R}^{n+m} \to \mathbb{R}^n$, $h : \mathbb{R}^{n+m} \to \mathbb{R}^p$ are nonlinear functions, which are assumed to be differentiable w.r.t. their arguments.

10.1016/j.ifacol.2023.10.086

According to Rotondo [2022], given a state trajectory $\tilde{x}(k)$ corresponding to input and output signals $\tilde{u}(k)$ and $\tilde{y}(k)$, and a *weighting function* $\rho : \mathbb{R}^{n+m+1} \to \mathbb{R}_{\geq 0}$ satisfying:

$$\int \cdots \int_{\mathbb{R}^{n+m}} \rho(x,u,k)dxdu = 1 \qquad \forall k \in \mathbb{N} \tag{2}$$

we define as *linearized system weighted through* ρ the following:

$$\begin{cases} \Delta x(k+1) \approx A(k)\Delta x(k) + B(k)\Delta u(k) \\ \Delta y(k) \approx C(k)\Delta x(k) + D(k)\Delta u(k) \end{cases} \tag{3}$$

where $\Delta x(k) \triangleq x(k) - \tilde{x}(k)$, $\Delta u(k) \triangleq u(k) - \tilde{u}(k)$ and $\Delta y(k) \triangleq y(k) - \tilde{y}(k)$, and the matrices $A(k)$, $B(k)$, $C(k)$ and $D(k)$ are obtained as follows:

$$A(k) = \int \cdots \int_{\mathbb{R}^{n+m}} \rho(x,u,k)\frac{Df}{Dx}(x,u)dxdu \tag{4}$$

$$B(k) = \int \cdots \int_{\mathbb{R}^{n+m}} \rho(x,u,k)\frac{Df}{Du}(x,u)dxdu \tag{5}$$

$$C(k) = \int \cdots \int_{\mathbb{R}^{n+m}} \rho(x,u,k)\frac{Dh}{Dx}(x,u)dxdu \tag{6}$$

$$D(k) = \int \cdots \int_{\mathbb{R}^{n+m}} \rho(x,u,k)\frac{Dh}{Dx}(x,u)dxdu \tag{7}$$

Based on Rotondo [2022], at each sample $k \in \mathbb{N}$ the matrices $A(k)$, $B(k)$, $C(k)$, $D(k)$ minimize the weighted Frobenius norm of the differences between the Jacobians of f and h w.r.t. x and u and the state-space matrices of the linearized representation, i.e.:

$$A(k) = \arg\min_{\Phi \in \mathbb{R}^{n\times n}} J_k^{fx}(\Phi) \quad B(k) = \arg\min_{\Phi \in \mathbb{R}^{n\times m}} J_k^{fu}(\Phi)$$

$$C(k) = \arg\min_{\Phi \in \mathbb{R}^{p\times n}} J_k^{hx}(\Phi) \quad D(k) = \arg\min_{\Phi \in \mathbb{R}^{p\times m}} J_k^{hu}(\Phi)$$

where the cost function $J_k^{fx}(\Phi)$ is defined as:

$$J_k^{fx}(\Phi) = \int \cdots \int_{\mathbb{R}^{n+m}} \rho(x,u,k) \left\| \frac{Df}{Dx}(x,u) - \Phi \right\|_F^2 dxdu \tag{8}$$

with $\|\cdot\|_F$ denoting the Frobenius norm, and the cost functions $J_k^{fu}(\Phi)$, $J_k^{hx}(\Phi)$ and $J_k^{hu}(\Phi)$ are obtained from (8) after appropriate replacement of f and/or x with h and/or u.

Remark: The practical interpretation of the weighting function $\rho(\cdot)$ appearing in (2) is that it describes how to weight different regions of the state and input spaces when obtaining the values of the matrices $A(k)$, $B(k)$, $C(k)$, $D(k)$ of the linearized model (3) that *better fit* the original nonlinear system (1). For example, by choosing $\rho(\cdot)$ as the following multivariable Dirac delta function:

$$\rho(x,u,k) = \delta(x - \tilde{x}(k), u - \tilde{u}(k)) \tag{9}$$

the standard linearization is recovered, and only the behavior of $f(\cdot)$ and $h(\cdot)$ in an infinitesimal neighborhood of the current values $\tilde{x}(k)$ and $\tilde{u}(k)$ is taken into consideration. On the other hand, by choosing $\rho(\cdot)$ as a different function taking non-zero values when $x \neq \tilde{x}(k)$ and $u \neq \tilde{u}(k)$, the behavior of $f(\cdot)$ and $h(\cdot)$ in other regions of the state and input space is accounted for.

A class of weighting functions that has showed to be promising in terms of improved performance is that of multidimensional Gaussian functions with unit hypervolume and centered around the current value of state and input:

$$\rho(x,u,k) = \sqrt{\frac{\det \Xi}{\pi^{n+m}}} e^{-\begin{bmatrix} x - \tilde{x}(k) \\ u - \tilde{u}(k) \end{bmatrix} \Xi \begin{bmatrix} x - \tilde{x}(k) \\ u - \tilde{u}(k) \end{bmatrix}} \tag{10}$$

where Ξ is a positive definite $(n+m) \times (n+m)$ matrix.

1.2 Contribution

Motivated by the above discussion, the main contribution of this paper is to propose a new version of the Kalman filter, which incorporates the weighted linearization, and will be thus referred to as *weighted Kalman filter* (WKF). Although the overall structure of the WKF resembles that of the EKF, they differ in several aspects:

- rather than calculating the Jacobian matrices at the current mean of the estimated state, the WKF obtains the linearized model through the multiple integral over the state space of the product between the Jacobian matrix functions and a weighting function (the weighting function is chosen to be a multivariable Gaussian function with generalized variance proportional to the current covariance matrix of the state estimate);
- similar modifications are made to the equations used to estimate the *a priori state and output estimates* based on the available nonlinear model;
- according to the results obtained in the illustrative example, the WKF outperforms the EKF in terms of both the convergence properties and the estimation error performance.

1.3 Outline

The remainder of this paper is structured as follows. Section 2 presents the problem statement and describes the prediction and update equations of the WKF. Section 3 discusses the illustrative example. Finally, Section 4 draws the main conclusions.

2. WEIGHTED KALMAN FILTER

2.1 Problem statement

Let us consider the following discrete-time nonlinear system:

$$\begin{cases} x(k+1) = f\left(x(k), u(k)\right) + w(k) \\ y(k) = h\left(x(k)\right) + v(k) \end{cases} \tag{11}$$

where $x \in \mathbb{R}^n$ is the (unknown) state vector, $u \in \mathbb{R}^m$ is the (known) input vector, $y \in \mathbb{R}^p$ is the (known) output vector, $w \in \mathbb{R}^n$ is the (unknown) process noise, $v \in \mathbb{R}^p$ is the (unknown) measurement noise, and $f(\cdot)$ and $h(\cdot)$ are (known) nonlinear functions, which are assumed to be differentiable w.r.t. the state variables. It is also assumed that $w(k)$ and $v(k)$ are zero mean multivariate Gaussian noises with (known) covariances $Q(k) \in \mathbb{R}^{n\times n}$ and $R(k) \in \mathbb{R}^{p\times p}$.

The problem under consideration is obtaining a state estimate $\hat{x}(k)$ using a modified version of the EKF, referred to as WKF, that incorporates the ideas behind the weighted linearization recalled in Section 1.1. In the following, we will discuss the prediction and update equations of the WKF, in relationship with the corresponding equations of the EKF.

2.2 Prediction equations

Let us start our discussion by recalling that in the EKF, the predicted state estimate $\bar{x}(k) \in \mathbb{R}^n$ (also known as *a priori estimate*) is obtained by using the information about the model (i.e., the function $f(\cdot)$) taking into account the current value of the *a posteriori estimate* $\hat{x}(k) \in \mathbb{R}^n$:

$$\bar{x}(k) = f\left(\hat{x}(k-1), u(k-1)\right) \tag{12}$$

However, the equation (12) does not take into account that $\hat{x}(k-1)$ is an uncertain variable, for which the information about the degree of uncertainty is contained in the corresponding covariance matrix $P(k-1)$. In order to address this shortcoming, we propose to use the weighted linearization and choose the weighted function as in (10) by taking into account that the best guess for the previous unknown value $\tilde{x}(k-1)$ is $\hat{x}(k-1)$ and using a matrix Ξ that is proportional to the inverse of the covariance matrix $P(k-1)$, thus leading to:

$$\hat{\rho}(x,k) = \sqrt{\frac{\det\left(H(k-1)\right)}{\pi^n}} e^{-(x-\hat{x}(k-1))^T H(k-1)(x-\hat{x}(k-1))} \tag{13}$$

where:

$$H(k-1) = \hat{\gamma} P(k-1)^{-1} \tag{14}$$

where $\hat{\gamma}$ is a positive parameter that affects the narrowness of the weighting function. Then, we use the model function $f(\cdot)$ to project *all* the possible values of $\hat{x}(k-1) \in \mathbb{R}^n$, which are eventually combined together using the weighting function (13), thus obtaining:

$$\bar{x}(k) = \int\int\cdots\int_{\mathbb{R}^n} \hat{\rho}(x,k) f\left(x, u(k-1)\right) dx \tag{15}$$

In the EKF, the prediction phase is completed by projecting the covariance matrix using the state matrix obtained by computing the Jacobian of the function $f(\cdot)$ at $\hat{x}(k-1)$:

$$M(k) = A(k)P(k-1)A(k)^T + Q(k) \tag{16}$$

where:

$$A(k) = \left.\frac{\partial f}{\partial x}\right|_{\hat{x}(k-1),u(k-1)} \tag{17}$$

We propose to maintain (16) to propagate the covariance matrix in time, but using the state matrix $A(k)$ of the linearized system weighted through $\hat{\rho}(x,k)$ instead:

$$A(k) = \int\int\cdots\int_{\mathbb{R}^n} \hat{\rho}(x,k) \left.\frac{\partial f}{\partial x}\right|_{x,u(k-1)} dx \tag{18}$$

If the state estimate $\hat{x}(k-1)$ is quite uncertain (the eigenvalues of the covariance matrix $P(k-1)$ are relatively big), then the weighting function (13) will be considerably different from zero in a wider region of the state-space, and the multiple integrals appearing in Equations (15) and (18) will account for the general behavior of the function $f(\cdot)$ not only at the current mean value of the state estimate $\hat{x}(k-1)$, but around this value as well. In the opposite case, i.e., when a very precise estimate $\hat{x}(k-1)$ is available ($P(k-1) \to 0$), then the multivariable Gaussian function will tend to a multivariable Dirac delta function, which means that in this particular situation the WKF would behave exactly as an EKF. It is worth mentioning that the input signal $u(k-1)$ has been considered to be perfectly known, which is the reason why the weighting function $\hat{\rho}$ in (13) has been chosen not to depend upon u, but the ideas described above can be extended to the case of an unknown input signal with known probability distribution by tweaking the weighting function $\hat{\rho}$ and integrating with respect to du as well, so that $\mathbb{R}^n$ gets replaced by $\mathbb{R}^n \times \mathbb{R}^m$. This latter case is not detailed in the formulation of the WKF presented in this paper, for the sake of keeping the notational burden limited.

2.3 Update equations

Let us continue our discussion by recalling that in the EKF, the computation of the Kalman gain $K(k)$, used to update the state estimate taking into account the so-called *innovation*, is performed using an output matrix that is obtained from evaluating the Jacobian of the function $h(\cdot)$ at $\bar{x}(k)$:

$$K(k) = M(k)C(k)^T\left[C(k)M(k)C(k)^T + R_v\right]^{-1} \tag{19}$$

where:

$$C(k) = \left.\frac{\partial h}{\partial x}\right|_{\bar{x}(k)} \tag{20}$$

We propose to maintain (19) for computing the Kalman gain, but we use the weighted linearization to get the matrix $C(k)$ by using a weighting function $\rho(x,u,k)$ as in (10) with $\bar{x}(k)$ as the best guess for the current unknown $\tilde{x}(k)$ and choosing the matrix Ξ to be proportional to the inverse of the covariance matrix $M(k)$:

$$C(k) = \int\cdots\int_{\mathbb{R}^n} \bar{\rho}(x,k) \left.\frac{\partial h}{\partial x}\right|_{x} dx \tag{21}$$

where:

$$\bar{\rho}(x,k) = \sqrt{\frac{\det\left(W(k)\right)}{\pi^n}} e^{-(x-\bar{x}(k))^T W(k)(x-\bar{x}(k))} \tag{22}$$

with:

$$W(k) = \bar{\gamma} M(k)^{-1} \tag{23}$$

where $\bar{\gamma}$ is a positive parameter that, similar to $\hat{\gamma}$ in (14), affects the narrowness of the weighting function.

The following step in the update phase is to obtain the a posteriori state estimate $\hat{x}$ by modifying the a priori estimate $\bar{x}$, as follows:

$$\hat{x}(k) = \bar{x}(k) + K(k)\left[y(k) - \bar{y}(k)\right] \tag{24}$$

However, while in the EKF the estimated output $\bar{y}(k)$ is obtained only from the current mean value of $\bar{x}(k)$:

$$\bar{y}(k) = h\left(\bar{x}(k)\right) \tag{25}$$

we propose to calculate $\bar{y}(k)$ by weighting the nonlinear function $h(x)$ through $\bar{\rho}(x,k)$, as follows:

$$\bar{y}(k) = \int\int\cdots\int_{\mathbb{R}^n} \bar{\rho}(x,k) h\left(x\right) dx \tag{26}$$

Finally, the last step during the update phase is to compute the current value of the a posteriori covariance matrix $P(k)$ as follows:

$$P(k) = \left[I - K(k)C(k)\right] M(k) \tag{27}$$

where the difference between the EKF and the proposed WKF lies in the different interpretation of the matrix $C(k)$ (and, consequently, a different value of the Kalman gain $K(k)$).

Note that, as already discussed in Section 2.2 for the prediction phase, the WKF update equations account for the uncertainty of the state estimate $\bar{x}(k)$ when computing $C(k)$ and $\bar{y}(k)$. In the ideal case of no uncertainty, the weighting function $\bar{\rho}(x,k)$ would become a multivariable Dirac delta function shifted at $\bar{x}(k)$, so that the EKF equations would be recovered.

2.4 Wrapping up

Let us finalize our discussion by summarizing the state estimation procedure referred to as WKF, which can be expressed as Algorithm 1.

Algorithm 1 The weighted Kalman filter

```
1:  Choose γ̂ > 0 and γ̄ > 0
2:  Initialize x̂(0) and P(0)
3:  k ← 1
4:  while TRUE (system is running) do
5:      Compute H(k − 1) using (14)        ▷ Prediction phase
6:      Compute x̄(k) using (15)
7:      Compute A(k) using (18)
8:      Compute M(k) using (16)
9:      Compute W(k) using (23)            ▷ Update phase
10:     Compute C(k) using (21)
11:     Compute K(k) using (19)
12:     Compute ȳ(k) using (26)
13:     Compute x̂(k) using (24)
14:     Compute P(k) using (27)
15:     k ← k + 1
16: end while
```

3. EXAMPLE

To evaluate the proposed WKF, we perform several simulations and compare the obtained performance in terms of convergence and average estimation error with that obtained by using a standard EKF. More specifically, let us consider the following nonlinear system:

$$\begin{cases} x_1(k+1) = x_1(k) + 0.1x_1(k)^2 + x_2(k) + w_1(k) \\ x_2(k+1) = -x_2(k) + 0.1x_2(k)^2 + u(k) + w_2(k) \\ y(k) = x_1(k) + v(k) \end{cases}$$

Given the a posteriori covariance matrix:

$$P(k-1) = \begin{bmatrix} p_{11}(k-1) & p_{12}(k-1) \\ p_{12}(k-1) & p_{22}(k-1) \end{bmatrix}$$

and the noise covariance matrices:

$$Q(k) = \begin{bmatrix} q_{11}(k) & q_{12}(k) \\ q_{12}(k) & q_{22}(k) \end{bmatrix} \qquad R(k) = r(k)$$

then Step 5 of Algorithm 1 happens as follows:

$$\begin{bmatrix} h_{11}(k-1) \\ h_{12}(k-1) \\ h_{22}(k-1) \end{bmatrix} = \frac{\begin{bmatrix} p_{22}(k-1) \\ -p_{12}(k-1) \\ p_{11}(k-1) \end{bmatrix}}{p_{11}(k-1)p_{22}(k-1) - p_{12}(k-1)^2}$$

During Steps 6-7, $\bar{x}_1(k)$, $\bar{x}_2(k)$, and the elements $a_{11}(k)$ and $a_{22}(k)$ of the matrix $A(k)$ are computed as follows ($a_{12}(k) = 1$ and $a_{21}(k) = 0$ for all k):

$$\bar{x}_1(k) = \iint_{-\infty}^{+\infty} \hat{\rho}(x_1, x_2, k) \left(x_1 + 0.1x_1^2 + x_2\right) dx_1 dx_2$$

$$\bar{x}_2(k) = \iint_{-\infty}^{+\infty} \hat{\rho}(x_1, x_2, k) \left(-x_2 + 0.1x_2^2\right) dx_1 dx_2 + u(k-1)$$

$$a_{11}(k) = \iint_{-\infty}^{+\infty} \hat{\rho}(x_1, x_2, k) \left(1 + 0.2x_1\right) dx_1 dx_2$$

$$a_{22}(k) = \iint_{-\infty}^{+\infty} \hat{\rho}(x_1, x_2, k) \left(-1 + 0.2x_2\right) dx_1 dx_2$$

where a value $\hat{\gamma} = 1$ has been chosen for computing the weighting function $\hat{\rho}(x_1, x_2, k)$, thus obtaining:

$$\hat{\rho}(x_1, x_2, k) = \exp\left[-h_{11}(k-1)\left(x_1 - \hat{x}_1(k-1)\right)^2 \cdots - 2h_{12}(k-1)\left(x_1 - \hat{x}_1(k-1)\right)\left(x_2 - \hat{x}_2(k-1)\right) \cdots -h_{22}(k-1)\left(x_2 - \hat{x}_2(k-1)\right)^2\right] / \cdots \left(\pi\sqrt{p_{11}(k-1)p_{22}(k-1) - p_{12}(k-1)^2}\right)$$

Then, the prediction phase is completed by Step 8, i.e.:

$$m_{11}(k) = a_{11}(k)^2 p_{11}(k-1) + 2a_{11}(k)p_{12}(k-1) + p_{22}(k-1) + q_{11}(k)$$

$$m_{12}(k) = a_{11}(k)a_{22}(k)p_{12}(k-1) + a_{22}(k)p_{22}(k-1) + q_{12}(k)$$

$$m_{22}(k) = a_{22}(k)^2 p_{22}(k-1) + q_{22}(k)$$

For this specific example, due to the linearity of the output equation $y(k) = x_1(k) + v(k)$ and the fact that the weighting function $\bar{\rho}(x, k)$ in (22) satisfies:

$$\int\int \cdots \int_{\mathbb{R}^n} \bar{\rho}(x, k) dx = 1$$

then the WKF equations for the update phase (Steps 9-14 of Algorithm 1) match the corresponding equations of the EKF:

$$k_1(k) = m_{11}(k) / \left(m_{11}(k) + r(k)\right)$$
$$k_2(k) = m_{12}(k) / \left(m_{11}(k) + r(k)\right)$$
$$\hat{x}_1(k) = \bar{x}_1(k) + k_1(k)\left(y(k) - \bar{x}_1(k)\right)$$
$$\hat{x}_2(k) = \bar{x}_2(k) + k_2(k)\left(y(k) - \bar{x}_1(k)\right)$$
$$p_{11}(k) = r(k)m_{11}(k) / \left(m_{11}(k) + r(k)\right)$$
$$p_{12}(k) = r(k)m_{12}(k) / \left(m_{11}(k) + r(k)\right)$$
$$p_{22}(k) = m_{22}(k) - m_{12}(k)^2 / \left(m_{11}(k) + r(k)\right)$$

In order to compare the WKF and the EKF, we will consider 32 different scenarios, as resumed in Table 1. For each of these scenarios, we will run 1000 different simulations, in which we create $w_1(k)$, $w_2(k)$, $v(k)$ as realizations of independent zero-mean white noise sequences with variances R_w, R_w and R_v, respectively. which corresponds to $Q(k) = \mathrm{diag}(R_w, R_w)$ and $R(k) = R_v$ for all k. Also, for each simulation, the initial condition x_0 is chosen as a random realization of a Gaussian variable with covariance given by $\mathrm{diag}\left(\mathrm{Var}(x_0), \mathrm{Var}(x_0)\right)$. Note that x_0 and the values in Table 1 are used to initialize the EKF and the WKF (Step 2 in Algorithm 1), as follows:

$$\hat{x}(0) = \left[x_0 \; 0\right]^T \qquad P(0) = \mathrm{diag}(R_v, \mathrm{Var}(x_0))$$

Finally, in each simulation the input signal $u(k)$ is computed so that $x_1(k)$ (the state variable that defines, up to the noise signal $v(k)$, the output $y(k)$) tracks [1] a sinusoidal signal with amplitude Λ and frequency ω:

$$x_1(k) = y(k) - v(k) = \Lambda \sin(\omega k)$$

For each simulation, we assess how many times the filters diverge (columns “Both diverge”, “EKF diverges” and “WKF diverges” in Table 2) and we compare the performance index (columns “EKF best” and “WKF best” in Table 2):

$$J = \sum_{k=0}^{k_f} \left(x(k) - \hat{x}(k)\right)^T \left(x(k) - \hat{x}(k)\right)$$

where $k_f = 50$ denotes the total number of samples of each simulation. Note that for each scenario, the values in the

[1] Since nonlinear controller design with guaranteed robust tracking performance goes beyond the scope of this paper, we compute $u(k)$ by using future values of the process noise $w(k)$. In spite of this not being an implementable solution, it serves the purpose of exemplifying the behavior of the WKF versus that of the EKF under different operating conditions.

Table 1. Description of the scenarios.

Scenario	R_w	R_v	Var(x_0)	Λ	ω
1	0.1	0.1	0.1	0.1	0.1
2	0.1	0.1	0.1	0.1	1
3	0.1	0.1	0.1	1	0.1
4	0.1	0.1	0.1	1	1
5	0.1	0.1	10	0.1	0.1
6	0.1	0.1	10	0.1	1
7	0.1	0.1	10	1	0.1
8	0.1	0.1	10	1	1
9	0.1	10	0.1	0.1	0.1
10	0.1	10	0.1	0.1	1
11	0.1	10	0.1	1	0.1
12	0.1	10	0.1	1	1
13	0.1	10	10	0.1	0.1
14	0.1	10	10	0.1	1
15	0.1	10	10	1	0.1
16	0.1	10	10	1	1
17	10	0.1	0.1	0.1	0.1
18	10	0.1	0.1	0.1	1
19	10	0.1	0.1	1	0.1
20	10	0.1	0.1	1	1
21	10	0.1	10	0.1	0.1
22	10	0.1	10	0.1	1
23	10	0.1	10	1	0.1
24	10	0.1	10	1	1
25	10	10	0.1	0.1	0.1
26	10	10	0.1	0.1	1
27	10	10	0.1	1	0.1
28	10	10	0.1	1	1
29	10	10	10	0.1	0.1
30	10	10	10	0.1	1
31	10	10	10	1	0.1
32	10	10	10	1	1

Table 2. Simulation results.

Scenario	Both diverge	EKF diverges	WKF diverges	EKF best	WKF best
1	0	0	0	178	**822**
2	0	0	0	184	**816**
3	0	0	0	186	**814**
4	0	0	0	222	**778**
5	177	1	**0**	297	**526**
6	149	2	**0**	309	**542**
7	169	0	0	325	**506**
8	192	0	0	285	**523**
9	0	0	0	**524**	476
10	0	0	0	**525**	475
11	0	0	0	301	**699**
12	0	0	0	**544**	456
13	177	23	**8**	304	**519**
14	196	22	**3**	294	**510**
15	178	15	**8**	205	**617**
16	220	25	**2**	309	**471**
17	241	320	**8**	107	**652**
18	244	321	**11**	91	**665**
19	224	316	**11**	91	**685**
20	225	311	**8**	112	**663**
21	469	236	**11**	63	**468**
22	438	250	**10**	85	**477**
23	445	229	**10**	73	**482**
24	446	235	**8**	75	**479**
25	538	223	**67**	117	**345**
26	545	201	**85**	136	**319**
27	561	234	**69**	109	**330**
28	526	234	**71**	125	**349**
29	701	139	**46**	85	**214**
30	688	163	**54**	80	**232**
31	708	147	**48**	80	**212**
32	705	145	**55**	83	**212**

columns "Both diverge", "EKF best" and "WKF best" sum the total number of simulations, i.e., 1000. For illustrative purposes, the best performing filter according to either convergence properties or attaining the least value of the performance index (fewer average error) is highlighted. It can be seen that in the vast majority of scenarios, the proposed WKF performs better than the EKF. It is worth noting that the only scenarios where the EKF performs slightly better than the WKF are Scenarios 9, 10 and 12, which are all characterized by low process noise, precise initialization of the filter (low variance of x_0), and high measurement noise. The results suggest that under a combination of these three factors, the WKF would not lead to any relevant performance improvement when compared to the EKF. It is also worth mentioning that the WKF involves multiple integrals that lead to an increased computational complexity when compared to the Taylor-based linearization used by the traditional EKF.

4. CONCLUSIONS

A new version of the Kalman filter that exploits recent results on the weighted linearization has been presented. In the proposed filter, instead of calculating the Jacobian matrices at the current mean of the estimated state, the linearized model is obtained by integrating over the state space the product of the Jacobian matrix function by a weighting function. Similar adjustments are applied to the equations used to determine the a priori state and output estimates. The illustrative example has shown that the proposed weighted Kalman filter exhibits better convergence properties and average estimation error than the extended Kalman filter. From a theoretical perspective, further investigation about the WKF will aim at investigating how the proposed method performs when handling unknown inputs. Furthermore, the practical applicability of the proposed filter to problems of industrial interest, such as for example fault diagnosis of nonlinear systems, will be researched.

REFERENCES

N. U. Ahmed and S. M. Radaideh. Modified extended Kalman filtering. *IEEE transactions on Automatic Control*, 39(6): 1322–1326, 1994.

L. Böhler, D. Ritzberger, C. Hametner, and S. Jakubek. Constrained extended Kalman filter design and application for on-line state estimation of high-order polymer electrolyte membrane fuel cell systems. *International Journal of Hydrogen Energy*, 46(35):18604–18614, 2021.

C. K. Chui, G. Chen, and H. C. Chui. Modified extended kalman filtering and a real-time parallel algorithm for system parameter identification. *IEEE Transactions on Automatic Control*, 35(1):100–104, 1990.

A. Germani, C. Manes, and P. Palumbo. Polynomial extended Kalman filter. *IEEE Transactions on Automatic Control*, 50 (12):2059–2064, 2005.

L. Glielmo, R. Setola, and F. Vasca. An interlaced extended Kalman filter. *IEEE Transactions on automatic control*, 44 (8):1546–1549, 1999.

Y. Huang, F. Zhu, G. Jia, and Y. Zhang. A slide window variational adaptive Kalman filter. *IEEE Transactions on Circuits and Systems II: Express Briefs*, 67(12):3552–3556, 2020.

Z. Huang, P. Du, D. Kosterev, and B. Yang. Application of extended Kalman filter techniques for dynamic model parameter calibration. In *IEEE Power & Energy Society General Meeting*, pages 1–8. IEEE, 2009.

S. J. Julier and J. K. Uhlmann. New extension of the Kalman filter to nonlinear systems. In *Signal processing, sensor fusion, and target recognition VI*, volume 3068, pages 182–193. International Society for Optics and Photonics, 1997.

J. Liu and G. Guo. Vehicle localization during GPS outages with extended Kalman filter and deep learning. *IEEE Transactions on Instrumentation and Measurement*, 70:1–10, 2021.

L. Loron and G. Laliberte. Application of the extended Kalman filter to parameters estimation of induction motors. In *5th European Conference on Power Electronics and Applications*, pages 85–90. IET, 1993.

K. Myers and B. Tapley. Adaptive sequential estimation with unknown noise statistics. *IEEE Transactions on Automatic Control*, 21(4):520–523, 1976.

H. Narayanan, L. Behle, M. F. Luna, M. Sokolov, G. Guillén-Gosálbez, M. Morbidelli, and A. Butté. Hybrid-EKF: Hybrid model coupled with extended Kalman filter for real-time monitoring and control of mammalian cell culture. *Biotechnology and Bioengineering*, 117(9):2703–2714, 2020.

D. Rotondo. Weighted linearization of nonlinear systems. In *IEEE Transactions on Circuits and Systems II: Express Briefs*, pages 1–5. IEEE, 2022.

J. Sarmavuori and S. Sarkka. Fourier-Hermite Kalman filter. *IEEE Transactions on Automatic Control*, 57(6):1511–1515, 2011.

C. Shen, Y. Zhang, X. Guo, X. Chen, H. Cao, J. Tang, J. Li, and J. Liu. Seamless GPS/inertial navigation system based on self-learning square-root cubature Kalman filter. *IEEE Transactions on Industrial Electronics*, 68(1):499–508, 2020.

D.-J. Xin and L.-F. Shi. Kalman filter for linear systems with unknown structural parameters. In *IEEE Transactions on Circuits and Systems II: Express Briefs*, pages 1–5. IEEE, 2021.

S. Zhao and B. Huang. Trial-and-error or avoiding a guess? Initialization of the Kalman filter. *Automatica*, 121:109184, 2020.