# Partitioned Update Kalman Filter

**MATTI RAITOHARJU**
**ROBERT PICHÉ**
**JUHA ALA-LUHTALA**
**SIMO ALI-LÖYTTY**

**In this paper we present a new Kalman filter extension for state update called Partitioned Update Kalman Filter (PUKF). PUKF updates the state using multidimensional measurements in parts. PUKF evaluates the nonlinearity of the measurement function within a Gaussian prior by comparing the effect of the 2nd order term on the Gaussian measurement noise. A linear transformation is applied to measurements to minimize the nonlinearity of a part of the measurement. The measurement update is then applied using only the part of the measurement that has low nonlinearity and the process is then repeated for the updated state using the remaining part of the transformed measurement until the whole measurement has been used. PUKF does the linearizations numerically and no analytical differentiation is required. Results show that when the measurement geometry allows effective partitioning, the proposed algorithm improves estimation accuracy and produces accurate covariance estimates.**

Manuscript received March 9, 2015; released for publication November 25, 2015.

Refereeing of this contribution was handled by Ondrej Straka.

Authors' addresses: M. Raitoharju and R. Piché, Department of Automation Science and Engineering, Tampere University of Technology (e-mail: {matti.raitoharju, robert.piche}@tut.fi). J. Ala-Luhtala and S. Ali-Löytty, Department of Mathematics, Tampere University of Technology (e-mail: {juha.ala-luhtala, simo.ali-loytty}@tut.fi).

The authors declare that they have no competing interests. Juha Ala-Luhtala received financial support from the Tampere University of Technology Doctoral Programme in Engineering and Natural Sciences. The simulations were carried out using the computing resources of CSC–IT Center for Science.



## I. INTRODUCTION

Bayesian filtering algorithms are used to compute the estimate of an $n$-dimensional state $x$. In a general discrete-time model the state evolves according to a state transition equation

$$x_t = \tilde{f}_t(x_{t-1}, \varepsilon_t^x), \tag{1}$$

where $\tilde{f}_t$ is the state transition function at time index $t$ and $\varepsilon_t^x$ is the state transition noise. The state estimate is updated using measurements that are modeled as

$$y_t = \tilde{h}_t(x_t, \varepsilon_t^y), \tag{2}$$

where $\tilde{h}_t$ is a measurement function and $\varepsilon_t^y$ is the measurement noise. If the measurement and state transition are linear, noises are additive, white and normal distributed, and the prior state ($x_0$) is normal distributed, the Kalman update can be used to compute the posterior. If these requirements are not fulfilled, usually an approximate estimation method has to be used. In this work, we concentrate on situations where the noises are additive and Gaussian so that (1–2) take the form

$$x_t = f_t(x_{t-1}) + \varepsilon_t^x \tag{3}$$

$$y_t = h_t(x_t) + \varepsilon_t^y, \tag{4}$$

where $\varepsilon_t^x \sim \mathrm{N}(0, W_t)$, $W_t$ is the state transition noise covariance, $\varepsilon_t^y \sim \mathrm{N}(0, R_t)$, and $R_t$ is the measurement noise covariance.

There are two main approaches for computing an approximation of the posterior distribution:

1) Approximate probabilities using point masses (e.g. grid and particle filters)
2) Approximate probabilities by Gaussians (e.g. Kalman filter extensions)

In the first approach one problem is how to choose a good number of point masses. The first approach also often requires more computational resources than the second approach. A drawback of the second approach is that the state distribution is assumed normal and unimodal, which makes the estimate inaccurate when the true posterior is not normal. Gaussian Mixture Filters (GMFs) (a.k.a. Gaussian sum filters) can be considered as a hybrid approach that use multiple normal distributions to estimate the probability distributions and can approximate any probability density function (pdf) [1]. GMFs have the same kind of problems as the algorithms using point masses in choosing a good number of components. The algorithm that will be proposed in this paper uses the second approach and so we will concentrate on it.

The algorithms that are based on Gaussian approximations usually extend the Kalman filter update to nonlinear measurements (there are also other options, see for example [2], [3]). The Extended Kalman Filter (EKF) is a commonly used algorithm for estimation with nonlinear measurement models [4]. EKF is

based on the first order Taylor linearization of the measurement function at the mean of prior. In the Second Order Extended Kalman Filter (EKF2) the linearization takes also the second order expansion terms into account [4]. In contrast to EKF, in EKF2 the prior covariance also affects the linearization. Both EKF and EKF2 require analytical computation of the Jacobian matrix and EKF2 requires also the computation of Hessian matrices of the measurement function. In [5] a 2nd order Central Difference Filter (CDF), which can be interpreted as a derivative-free numerical approximation of EKF2, was presented. The most commonly used Kalman filter extension that does not require analytical differentiation is probably the Unscented Kalman Filter (UKF) [6]. The Gaussian approximations in UKF are based on the propagation of "sigma points" through the nonlinear functions. Cubature Kalman Filters (CKFs) are similar algorithms, but they have different theory in the background [7]. All these methods do the update as a single operation.

Some algorithms do multiple linearizations to improve the estimate. In [8] the posterior is computed using multiple EKF updates that use different linearization points. In the Iterated Extended Kalman Filter (IEKF), the EKF update is computed in the prior mean and then the new mean is used as the new linearization point [9]. This can be done several times. A similar update can be done also with other Kalman type filters [10]. The Recursive Update Filter (RUF) updates the prior with measurement with reduced weight several times [11]. In every update the linearization point is used from the posterior of the last reduced weight update. GMFs can also be considered to be filters that do the linearization multiple times, once for each Gaussian component, and any Kalman filter extension can be used for the update.

In this paper we present Partitioned Update Kalman Filter (PUKF) that updates the state also in several steps. PUKF first computes the nonlinearity of measurement models. The nonlinearity measure is based on comparing the covariance of the 2nd order term covariance of the Gaussian measurement noise. Computation of this nonlinearity measure requires the same matrices as the EKF2 update and for this we use the 2nd order CDF [5], which is a derivative free version of the EKF2.

PUKF applies a linear transformation to the measurement function to make a new measurement function that has linearly independent measurement noise for measurement elements; the smallest nonlinearity corresponding to a measurement element is minimized first, then the second smallest nonlinearity etc. After the transformation, the update is done using only measurement elements that have smaller nonlinearity than a set threshold value or using the measurement element with the smallest nonlinearity. After the partial measurement update the covariance has become smaller or remained the same and the linearization errors for remaining measurements may have also became smaller. The remaining measurements' nonlinearity is re-evaluated using the partially updated state, the remaining measurements are transformed and a new partial update is applied until the whole measurement is applied. This process is shown in Figure 1. The use of only some dimensions of the measurements to get a new prior and the optimization of measurement nonlinearities differentiates PUKF from other Kalman filter extensions.

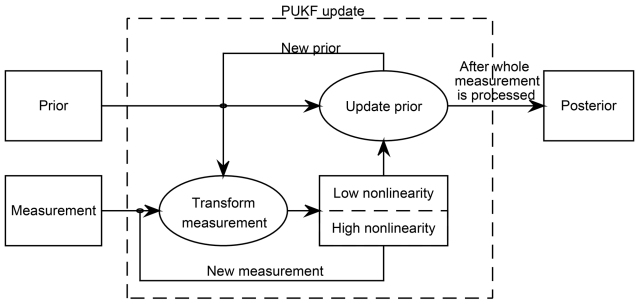


Fig. 1. Process diagram of the PUKF

The article is structured as follows: In Section II a numerical method for approximate EKF2 update is presented. The main algorithm is presented in Section III. The accuracy and reliability of the proposed algorithm is compared with other Kalman filter extensions and PFs in Section IV. Section V concludes the article.

## II. EKF2 AND ITS NUMERICAL UPDATE USING 2ND ORDER CDF

Kalman filter extensions, like all Bayesian filters, can be computed in two stages: prediction and update. For the state transition model (3) the state is propagated in EKF2 using equations [9]:

$$\mu_t^- = f_t(\mu_{t-1}^+) + \tfrac{1}{2}\xi_t^f \tag{5}$$

$$P_t^- = J^f P_{t-1}^+ J^{f^T} + \tfrac{1}{2}\Xi_t^f + W_t, \tag{6}$$

where $\mu_t^-$ is the predicted mean at time $t$, $\mu_{t-1}^+$ is the posterior mean of the previous time step, $J^f$ is the Jacobian of the state transition function evaluated at $\mu_{t-1}^+$, $P_t^-$ is the predicted covariance, $P_{t-1}^+$ is the posterior covariance of the previous time step and $\xi_t^h$ and $\Xi_t^h$ are defined as

$$\xi_{t[i]}^f = \operatorname{tr} P_{t-1}^+ H_i^f \tag{7}$$

$$\Xi_{t[i,j]}^f = \operatorname{tr} P_{t-1}^+ H_i^f P_{t-1}^+ H_j^f, \tag{8}$$

where $H_i^f$ is the Hessian of the $i$th element of the state transition function evaluated at $\mu_{t-1}^+$. To simplify the notation we do not further show the time indices.

The update equations of EKF2 for the measurement model (4) are [9]

$$y^- = h(\mu^-) + \tfrac{1}{2}\xi^h \tag{9}$$

$$S = J^h P^- J^{h^T} + \tfrac{1}{2}\Xi^h + R \tag{10}$$

$$K = P^- J^{h^T} S^{-1} \tag{11}$$

$$\mu^+ = \mu^- + K(y - y^-) \tag{12}$$

$$P^+ = P^- - KSK^T, \tag{13}$$

where $J^h$ is the Jacobian of the measurement function, $K$ is the Kalman gain, $S$ is the innovation covariance, and $\xi^h$ and $\Xi^h$ are defined as

$$\xi^h_{[i]} = \operatorname{tr} P^- H^h_i \tag{14}$$

$$\Xi^h_{[i,j]} = \operatorname{tr} P^- H^h_i P^- H^h_j, \tag{15}$$

where $H^h_i$ is the Hessian matrix of the $i$th component of the measurement function. Eqns (9–13) can be turned into the EKF update using $\xi^h = 0$ and $\Xi^h = 0$.

If the measurement model is linear, the trace terms in EKF2 are zero and the update is the optimal update of the Kalman filter. When the measurement function is a second order polynomial the EKF2 update is not optimal as the distributions are no longer Gaussian, but the mean (9) and innovation covariance (10) are correct.

In this paper we use a numerical algorithm to compute an EKF2 like update. To derive this algorithm, we start with the formulas of the 2nd-order CDF from [5]. Let $\sqrt{P^-}$ be a matrix such that

$$\sqrt{P^-}\sqrt{P^-}^T = P^-. \tag{16}$$

In our implementation this matrix square root is computed using Cholesky decomposition.

Next we define matrices $M$ and $Q$ that are used for computing the numerical EKF2 update. We use notation $\Delta_i = \gamma\sqrt{P^-}_{[:,i]}$, where $\sqrt{P^-}_{[:,i]}$ is the $i$th column of matrix $\sqrt{P^-}$ and $\gamma$ is an algorithm parameter that defines the spread of the function evaluations. Matrix $M$, whose elements are

$$\begin{aligned} M_{[:,i]} &= \left[J^h\sqrt{P^-}\right]_{[:,i]} \\ &\approx \gamma^{-1}\frac{h(\mu^- + \Delta_i) - h(\mu^- - \Delta_i)}{2}, \end{aligned} \tag{17}$$

is needed for the terms with Jacobian. The matrices $Q_k \approx \sqrt{P^-}H^{h_k}\sqrt{P^-}^T$ are needed to compute terms with Hessians. Elements of $Q_k$ are

$$\begin{aligned} Q_{k_{[i,i]}} &= \gamma^{-2}[h_{[k]}(\mu^- + \Delta_i) + h_{[k]}(\mu^- - \Delta_i) - 2h_{[k]}(\mu^-)] \\ Q_{k_{[i,j]}} &= \gamma^{-2}[h_{[k]}(\mu^- + \Delta_i + \Delta_j) - h_{[k]}(\mu^- + \Delta_i) \\ &\quad - h_{[k]}(\mu^- + \Delta_j) + h_{[k]}(\mu^-)], \quad i \neq j. \end{aligned} \tag{18}$$

The EKF2 update can be approximated with these by doing the following substitutions:

$$\xi^h_i = \operatorname{tr} P^- H^h_i \approx \operatorname{tr} Q_i \qquad \text{in (9)} \tag{19}$$

$$J^h P^- J^{h^T} \approx MM^T \qquad \text{in (10)} \tag{20}$$

$$P^- J^{h^T} \approx \sqrt{P^-}M^T \qquad \text{in (14)} \tag{21}$$

$$\Xi^h_{[i,j]} = \operatorname{tr} P^- H^h_i P^- H^h_j \approx \operatorname{tr} Q_i Q_j. \qquad \text{in (15).} \tag{22}$$

The prediction step can be approximated by computing $M^f$ (17) and $Q^f$ (18) matrices using the state transition function instead of the measurement function and doing the following substitutions:

$$J^h P^+_{t-1} J^{h^T} \approx M^f M^{f^T} \qquad \text{in (6)} \tag{23}$$

$$\operatorname{tr} P^- H^f_i \approx \operatorname{tr} Q^f_i \qquad \text{in (7)} \tag{24}$$

$$\operatorname{tr} P^- H^f_i P^- H^f_j \approx \operatorname{tr} Q^f_i Q^f_j \qquad \text{in (8).} \tag{25}$$

In [12], an update algorithm similar to numerical EKF2 is proposed that uses only the diagonal elements of $Q$ matrices. They state that $\gamma = \sqrt{3}$ for Gaussian distributions is optimal because it preserves the fourth moment and so we use this $\gamma$ value in our algorithm.

## III. PARTITIONED UPDATE KALMAN FILTER

When the measurement function is linear and the measurement noise covariance is block diagonal, the Kalman update produces identical results whether measurements are applied one block at a time or all at once. In our approach we try to find as linear as possible part of the measurement and use this part to update the state estimate to reduce approximation errors in the remaining measurement updates. When the measurement noise covariance $R$ is not diagonal a linear transformation (decorrelation) is applied to transform the measurement so that the transformed measurement has diagonal covariance [13]. In PUKF, we choose this decorrelation so that the nonlinearity of the least nonlinear measurement element is minimized. The prior is updated using the least nonlinear part of the decorrelated measurements. After the partial update the process is repeated for the remaining dimensions of the transformed measurement.

For measuring the amount of nonlinearity we compare the trace term $\Xi^h$ with the covariance of the measurement noise:

$$\begin{aligned} \eta &= \operatorname{tr}\sum_{i=1}^{d}\sum_{k=1}^{d} R^{-1}_{[k,i]} P^- H^h_i P^- H^h_k \\ &= \operatorname{tr}\sum_{i=1}^{d}\sum_{k=1}^{d} R^{-1}_{[k,i]}\Xi^h_{[i,k]} \end{aligned} \tag{26}$$

This nonlinearity measure is a local approximation of the nonlinearity and is developed from the measure presented in [9], [14]. In [15] it was compared with other nonlinearity measures and it was shown to be a good indication of how accurately state can be updated with a nonlinear measurement model using a Kalman filter extension. When the measurement model is linear the nonlinearity measure is $\eta = 0$.

The matrix $\Xi^h$ depends on the nonlinearity of the measurement function and contributes to the innovation covariance (10) similarly, but multiplied with $\frac{1}{2}$, as the Gaussian measurement noise $R$. The measure (26) compares the ratio of Gaussian covariance $R$ and non-Gaussian covariance $\Xi^h$. Figure 2 shows how the pdf

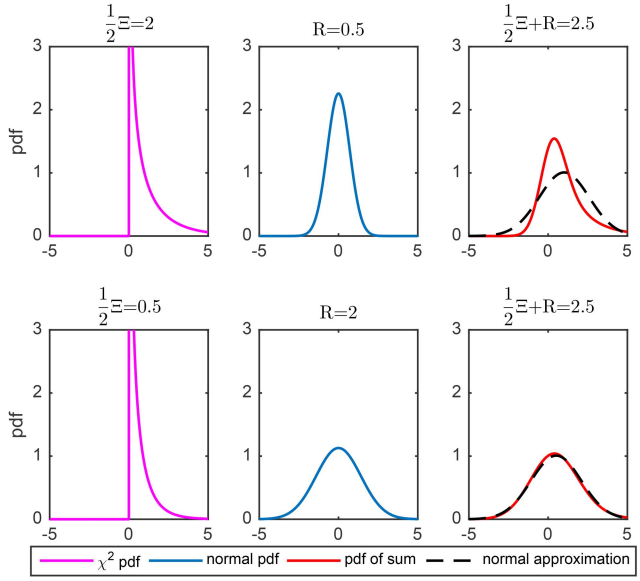



Fig. 2. Probability density functions of sums of independent $\chi^2$ and normal random variables with different variances

of the sum of independent normal and $\chi^2$ distributed random variables is closer to normal when $R > \Xi$ than when $R < \Xi$. The $\chi^2$ distribution is chosen in the example, because a normal distributed variable squared is $\chi^2$ distributed and in the second order polynomial approximations the squared term is the nonlinear part.

The nonlinearity measure (26) can be approximated numerically using the substitution (22). Numerical computation of a similar nonlinearity measure was proposed in [16], but the algorithm presented in Section II does the nonlinearity computation with fewer measurement function evaluations.

Multiplying (4) by an invertible square matrix $D$ gives a transformed measurement model

$$Dy = Dh(x) + D\varepsilon^y. \tag{27}$$

We use the following notations for the transformed measurement model: $\hat{y} = Dy$, $\hat{h}(x) = Dh(x)$, $\hat{R} = DRD^T$, and $\hat{\varepsilon}^y = D\varepsilon^y \sim \mathrm{N}(0, \hat{R})$. We will show that $D$ can be chosen so that

$$\hat{R} = I \quad \text{and} \quad \operatorname{tr} P^- \hat{H}_i^h P^- \hat{H}_k^h = 0, \quad i \neq k, \tag{28}$$

where $\hat{H}_i^h$ and $\hat{H}_k^h$ denote the Hessians of the $i$th and $k$th element of $\hat{h}(x)$.

In [15], it was shown that when a measurement model is transformed so that $\hat{R} = I$ the nonlinearity measure (26) is equal to the nonlinearity measure of the transformed measurements

$$\begin{aligned}\eta &= \operatorname{tr} \sum_{i=1}^{d} \sum_{k=1}^{d} R_{[k,i]}^{-1} P^- H_i^h P^- H_k^h \\ &= \operatorname{tr} \sum_{i=1}^{d} \sum_{k=1}^{d} \hat{R}_{[k,i]}^{-1} P^- \hat{H}_i^h P^- \hat{H}_k^h = \operatorname{tr} \sum_{i=1}^{d} P^- \hat{H}_i^h P^- \hat{H}_i^h.\end{aligned} \tag{29}$$

Because the cross terms do not affect to the amount of nonlinearity we can extract the nonlinearity caused by individual elements of the transformed measurements

$$\eta_i = \operatorname{tr} P^- \hat{H}_i^h P^- \hat{H}_i^h \tag{30}$$

and the total nonlinearity is

$$\eta = \sum_{i=1}^{d} \eta_i. \tag{31}$$

In Appendix A it is shown that

$$\hat{\Xi}_{[i,j]}^h = [D\Xi^h D^T]_{[i,j]} \approx \operatorname{tr} P^- \hat{H}_i^h P^- \hat{H}_j^h. \tag{32}$$

In this case the measurement related error terms of the transformed measurement $\hat{R}$ and $\hat{\Xi}^h$ are diagonal. This makes the measurements independent and allows the update of the state one element at a time.

In PUKF nonlinearities are minimized in such a way that $\eta_1$ (30) is as small as possible. Then $\eta_2$ is minimized such that $\eta_1$ does not change, and $\eta_3$ so that $\eta_1$ and $\eta_2$ do not change etc. The decorrelation transformation $D$ that does the desired nonlinearity minimization can be computed by first computing a matrix square root (16) of the measurement noise covariance

$$\sqrt{R}\sqrt{R}^T = R \tag{33}$$

and then an eigendecomposition of $\sqrt{R}^{-1} \Xi^h \sqrt{R}^{-T}$

$$U \Lambda U^T = \sqrt{R}^{-1} \Xi^h \sqrt{R}^{-T}. \tag{34}$$

We assume that the eigenvalues in the diagonal matrix $\Lambda$ are sorted in ascending order. The transformation matrix is now

$$D = U^T \sqrt{R}^{-1}. \tag{35}$$

A proof that this transformation minimizes the nonlinearity measures is given in Appendix B. After transforming the measurement model with this matrix, the measurement noise covariance is $\hat{R} = I$ and $\hat{\Xi}^h = \Lambda$.

After the measurement model is decorrelated (multiplied with $D$), the parts of measurement model that have low nonlinearity ($\Lambda_{[i,i]} \leq \eta_{\text{threshold}}$) are used in the update (Section II). If there is no such part then the most linear element of the measurement model is used to update the state. Then the same process is repeated for the remaining transformed measurement model until the whole measurement is processed.

In summary the PUKF update is:

1) Transform the measurement model using $D$ (35)
2) Update the prior using only the least nonlinear measurement elements of the transformed measurement
3) If there are measurement elements left, use them as new measurement and use the updated state as a new prior and return to step 1

The detailed PUKF algorithm is presented in Algorithm 1 and a Matlab implementation is available online [17].

ALGORITHM 1 *Algorithm for doing the measurement update in PUKF*

**input** : Prior state: $\mu$–mean $P$–covariance
Measurement model: $y$–value, $h(\cdot)$–function, $R$–covariance
$\eta_{\text{threshold}}$–nonlinearity limit,
$\gamma$–measurement function evaluation spread (default $\gamma = \sqrt{3}$)
**output**: Updated state: $\mu$–mean, $P$–covariance

Compute $\sqrt{R}$ (33)
$d \leftarrow$ measurement dimension
**while** $d > 0$ **do**
  Compute $\sqrt{P}$ (16)
  Compute $M$ and $Q_i$, $1 \le i \le d$ (17–18)
  Compute $\xi^h$ and $\Xi^h$ (19) and (22)
  Compute $U$ and $\Lambda$ (34)
  $D \leftarrow U^T \sqrt{R}^{-1}$
  Choose largest $k$ so that $\Lambda_{[i,i]} \le \eta_{\text{threshold}}$, $i \le k \wedge \Lambda_{[j,j]} > \eta_{\text{threshold}}$, $j > k$
  **if** $k == 0$ **then**
    $k \leftarrow 1$
  **end**
  // Compute partial EKF2 update
    $y^- \leftarrow D_{[1:k,:]}[h(\mu) + \frac{1}{2}\xi^h]$
    $S \leftarrow D_{[1:k,:]} M M^T D^T_{[1:k,:]} + \frac{1}{2}\Lambda_{[1:k,1:k]} + I$
    $K \leftarrow \sqrt{P} M^T D^T_{[1:k,:]} S^{-1}$
    $\mu \leftarrow \mu + K(D_{[1:k,:]} y - y^-)$
    $P \leftarrow P - KSK^T$
  // Update remaining measurement
    $y \leftarrow D_{[k+1:d,:]} y$
    $h(x) \leftarrow D_{[k+1:d,:]} h(x)$
    $\sqrt{R} \leftarrow I$ Updated measurement noise covariance is an identity matrix due to decorrelation
    $d \leftarrow d - k$ Updated measurement dimension
**end**

The amount of nonlinearity (26) for independent measurements is equal to the sum of the nonlinearities for each of the measurements. The quantity $\eta_{\text{threshold}}$ is compared separately to independent transformed measurements elements and, thus, we propose to use same $\eta_{\text{threshold}}$ regardless of the measurement dimension. As a rule of thumb the nonlinearity threshold can be set to $\eta_{\text{threshold}} = 1$, which is equal to the threshold proposed for one dimensional measurements in [9].

Figure 3 shows how PUKF treats a two-dimensional second order polynomial measurement function

$$y = \begin{bmatrix} x^2 - 2x - 4 \\ -x^2 + \frac{3}{2} \end{bmatrix} + \varepsilon, \tag{36}$$

where $\varepsilon \sim \mathrm{N}(0, I)$. The prior has mean 1 and covariance 1. The nonlinearity of each measurement is 4 and the total nonlinearity is 8. Then $D = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$ and the

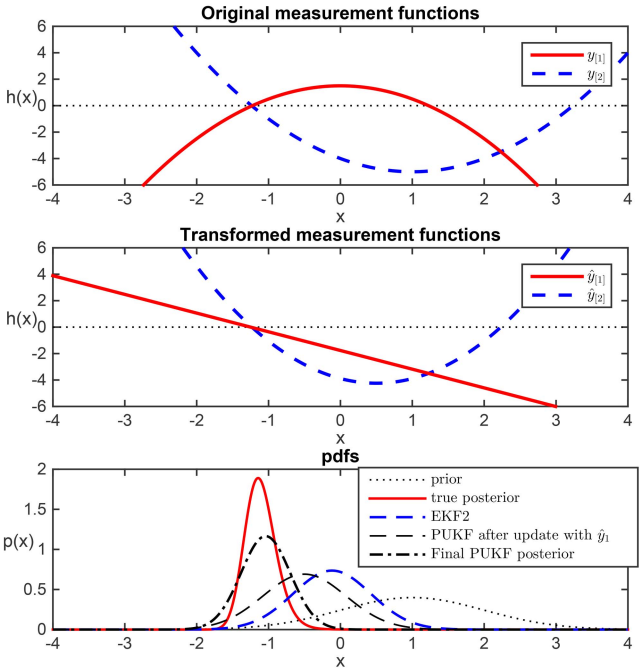



Fig. 3. Transforming second order polynomial measurements to minimize nonlinearity of $\hat{y}_1$ and posterior comparison of PUKF and EKF2

transformed measurement model has a linear term and a polynomial term

$$\hat{y} = \sqrt{2} \begin{bmatrix} -x - \frac{5}{4} \\ x^2 - x - \frac{11}{4} \end{bmatrix} + \hat{\varepsilon}, \tag{37}$$

where $\hat{\varepsilon} \sim \mathrm{N}(0, I)$. After transformation the first element of the measurement function is linear and $\eta_1 = 0$ and all the nonlinearity is associated with the second element $\eta_2 = 8$. In PUKF the linear measurement function is applied first and the partially updated state has mean $-\frac{1}{2}$ and covariance $\frac{1}{3}$. The polynomial measurement function is applied using this partially updated state. The amount of nonlinearity for the second order polynomial has decreased from 8 to $\frac{8}{9}$. EKF2 applies both measurements at once and the posterior estimate is the same for the original and transformed measurement models as shown in Appendix A. When comparing to the true posterior, which is computed using a dense grid, the posterior estimate of PUKF is significantly more accurate than the EKF2 posterior estimate.

## IV. TESTS

We compare the proposed PUKF with other Kalman filter extensions and a PF in three different test scenarios. The PUKF was tested with 4 different values for $\eta_{\text{threshold}}$. When $\eta_{\text{threshold}} = \infty$ the whole measurement is applied at once and the algorithm is a numerical EKF2. When $\eta_{\text{threshold}} < 0$ measurement elements are processed one at a time and when $\eta_{\text{threshold}} = 0$ all linear measurement elements are first processed together and then nonlinear measurement elements one by one. Due to numerical roundoff errors it is better to use a small positive

$\eta_{\text{threshold}}$ to achieve this kind of behaviour. In our tests we use values $\{-\infty, 0.1, 1, \infty\}$ for $\eta_{\text{threshold}}$.

EKF and EKF2 are implemented as explained in Section II with analytical Jacobians and Hessians. RUF is implemented according to [11] with 3 and 10 steps. IEKF uses 10 iterations. For UKF the values for sigma point parameters are $\alpha = 10^{-3}$, $\kappa = 0$, $\beta = 2$. All Kalman filter extensions are programmed in Matlab with similar levels of code optimizations, but the runtimes should still be considered to be only indicative.

For reference we computed estimates with a bootstrap particle filter that does systematic resampling at every time step [18] using various numbers of particles and with a PF that uses EKF for computing the proposal distribution [19] with 10 particles.

In every test scenario the state transition model is linear time-invariant $x_t = J^f x_{t-1} + \varepsilon^x$, where $\varepsilon^x \sim \mathrm{N}(0, W)$. Thus, the prediction step (5)–(6) can be computed analytically and all Kalman filter extensions in tests use the analytical prediction.

The first test scenario is an aritificial example chosen to show the maximal potential of PUKF. The measurement model used is

$$h(x) = \begin{bmatrix} 2x_{[1]} + x_{[2]} + x_{[3]} + \frac{1}{2}x_{[1]}^2 + \frac{1}{2}x_{[2]}^2 + \frac{1}{2}x_{[3]}^2 \\ x_{[1]} + 2x_{[2]} + x_{[3]} + \frac{1}{2}x_{[1]}^2 + \frac{1}{2}x_{[2]}^2 + \frac{1}{2}x_{[3]}^2 \\ x_{[1]} + x_{[2]} + 2x_{[3]} + \frac{1}{2}x_{[1]}^2 + \frac{1}{2}x_{[2]}^2 + \frac{1}{2}x_{[3]}^2 \\ x_{[1]} + x_{[2]} + x_{[3]} + x_{[1]}^2 + \frac{1}{2}x_{[2]}^2 + \frac{1}{2}x_{[3]}^2 \\ x_{[1]} + x_{[2]} + x_{[3]} + \frac{1}{2}x_{[1]}^2 + x_{[2]}^2 + \frac{1}{2}x_{[3]}^2 \\ x_{[1]} + x_{[2]} + x_{[3]} + \frac{1}{2}x_{[1]}^2 + \frac{1}{2}x_{[2]}^2 + x_{[3]}^2 \end{bmatrix} + \varepsilon^y, \tag{38}$$

where $\varepsilon^y \sim \mathrm{N}(0, 8I + \mathbf{1})$ and $\mathbf{1}$ is a matrix of ones. This model is a linear transformation of

$$\hat{h}(x) = \begin{bmatrix} x_{[1]} \\ x_{[2]} \\ x_{[3]} \\ \frac{1}{2}x_{[1]}^2 \\ \frac{1}{2}x_{[2]}^2 \\ \frac{1}{2}x_{[3]}^2 \end{bmatrix} + \hat{\varepsilon}^y, \tag{39}$$

where $\hat{\varepsilon}^y \sim \mathrm{N}(0, I)$. The first three elements of (39) are linear and PUKF with $\eta_{\text{threshold}} \in \{0.1, 1\}$ uses the three linear measurement functions first to update the state. In this test scenario the prior mean is at the origin, the prior and state transition noise covariances are both $16I$, and the state transition matrix is an identity matrix.

Results for positioning with measurement model (38) are presented in Figure 4. The markers in the upper plot show the 5%, 25%, 50%, 75% and 95% quantiles of mean errors for each method. The quantiles are computed from 1000 runs consisting of 10 steps each. To show the quantiles better a logarithmic scale for error is used. PUKF ($\eta_{\text{threshold}} < \infty$) is the most accurate of

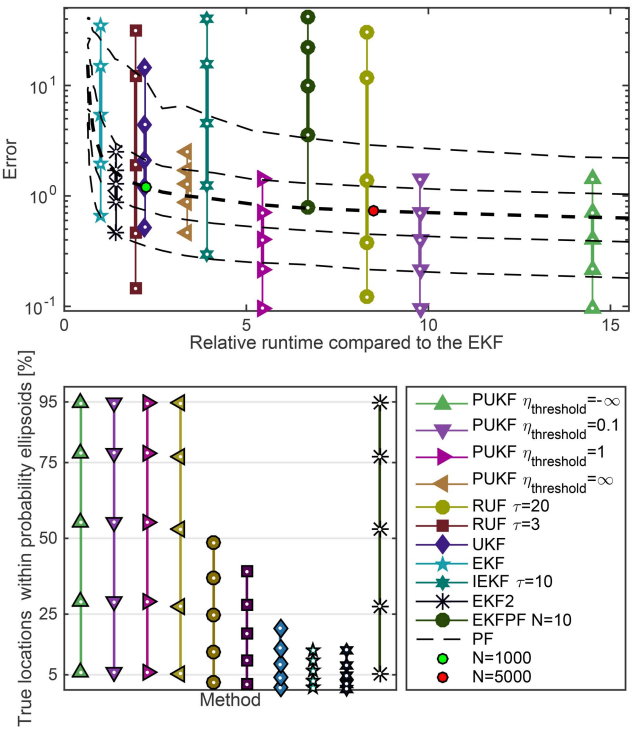



Fig. 4. Accuracy of different Kalman filter extensions in estimation with second order polynomial measurement model (38). In the top figure markers show the 5%, 25%, 50%, 75% and 95% quantiles of errors for each method for every estimated step. The errors are computed as the norm of the difference of the true and estimated mean. In the bottom figure the markers show how often the true state was within estimated error ellipsoids containing 5%, 25%, 50%, 75% and 95% of the probability mass.

the Kalman filter extensions by a large margin. When $\eta_{\text{threshold}} = \infty$ the whole measurement is processed at once and the result is the same as with EKF2, as expected. In this test scenario the PUKF performs clearly the best and methods that use EKF linearizations have very large errors. PUKF also outperforms PF with similar runtime.

In the bottom plot the accuracy of covariance estimates of different Kalman filter extensions are compared. For this plot we compute how often the true state is within the 5%, 25%, 50%, 75% and 95% ellipsoids of the Gaussian posterior. That is, a true location is within the $p$ ellipsoid when

$$\chi_n^2((\mu - x_{\text{true}})^T P^{-1}(\mu - x_{\text{true}})) < p, \tag{40}$$

where $x_{\text{true}}$ is the true state, $\mu$ and $P$ are the posterior mean and covariance computed by the filter, and $\chi_n^2$ is the cumulative density function of the chi-squared distribution with $n$ degrees of freedom. The filter's error estimate is reliable when markers are close to the $p$ values (dotted lines in the Figure). From the figure it is evident that PUKF and EKF2 have the most reliable error estimates and all other methods have too small covariance matrices.

The EKFPF did not perform wery well. This is probably caused by the inconsistency of EKF estimates

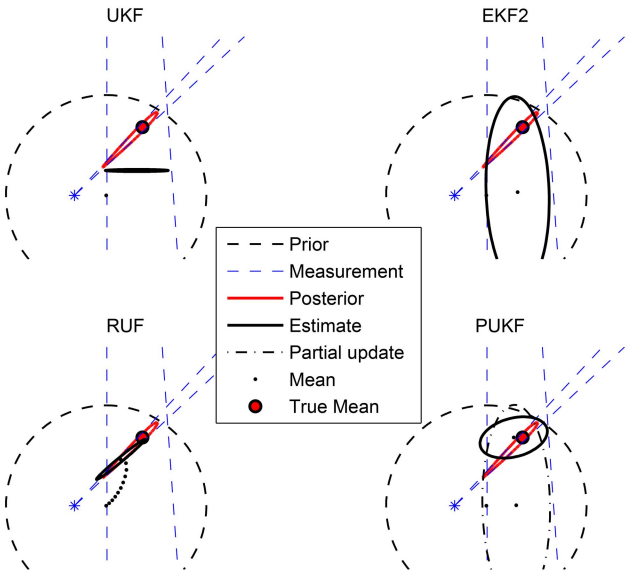



Fig. 5. Example situation of bearings positioning

that were used as the proposal distribution. We tested EKFPF also with 1000 particles. The estimation accuracy was similar to that obtained with a bootrstrap PF with 1000 particles, but the algorithm was much slower than other algorithms.

In our second test scenario the planar location of a target is estimated using bearing measurements. When the target is close to the sensor the measurement model is nonlinear, but when the target is far away the measurement becomes almost linear. The measurement model is

$$y = \text{atan2}(x_{[2]} - r_{[2]}, x_{[1]} - r_{[1]}) + \varepsilon^y, \qquad (41)$$

where atan2 is the four quadrant inverse tangent, $r$ is the sensor location, and measurement noises are zero mean independent, with standard deviation of $2^\circ$. We choose the branch of atan2 so that evaluated values are as close as possible to the realized measurement value. In the test scenario two bearings measurements are used, one from a sensor close to the prior and the second from a sensor far away.

A representative initial state update using UKF, EKF2, RUF and PUKF is shown in Figure 5. This example is chosen so that the differences between estimates of different filters is clearly visible. The red line encloses the same probability mass of the true posterior as the $1 \cdot \sigma$ ellipses (black lines) of the Gaussian approximations computed with different Kalman filter extensions. The measurement from the distant sensor is almost linear within the prior and UKF uses it correctly, but the linearization of the estimate from the nearby sensor is not good and the resulting posterior is very narrow (EKF would be similar). In the EKF2 update the second order term of the measurement model from the nearby sensor is so large that EKF2 almost completely ignores that measurement and the prior is updated using only the measurement from the distant sensor. The iterative

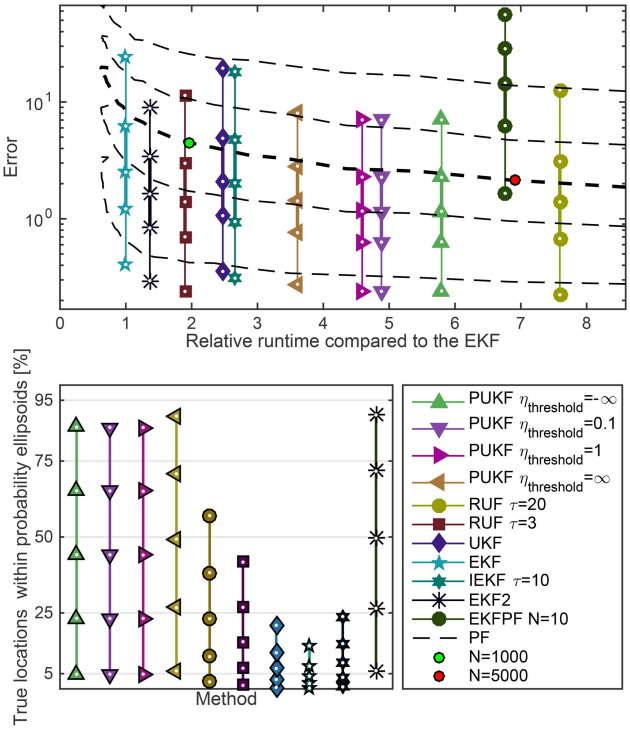



Fig. 6. Accuracy of different filters in bearings only tracking

update of RUF results in an estimate with small covariance that has similar shape as the true covariance. The mean of the true posterior is not inside the one-sigma ellipses of the RUF estimate and the mean is too close to the nearby sensor.

The first transformed measurement used by PUKF is almost the same as the measurement from the distant sensor and the estimate after the first partial update is similar to the EKF2 estimate. Because the estimate updated with the first measurement is further away from the nearby sensor the linearization of the second measurement is better and the posterior estimate is closer to the true posterior than with EKF2. The covariance estimate produced by PUKF is more conservative than the RUF of UKF covariances.

Figure 6 shows the statistics for this scenario. For this Figure the scenario was ran 1000 times using the same sensor locations and 10 step estimation with a 4-dimensional state model containing 2 position and 2 velocity dimensions. The prior has zero mean and covariance $10I$. The state transition function is

$$f(x) = \begin{bmatrix} I & I \\ \mathbf{0} & I \end{bmatrix} x + \varepsilon^x, \qquad (42)$$

where

$$\varepsilon^x \sim \text{N}\left(0, \begin{bmatrix} \frac{1}{300}I & \frac{1}{200}I \\ \frac{1}{200}I & \frac{1}{100}I \end{bmatrix}\right). \qquad (43)$$

Figure 6 shows that the PUKF provides the best accuracy. Interestingly RUF with 3 iterations has better accuracy than with 20 iterations. From the plot that shows the accuracy of the error estimates we can see

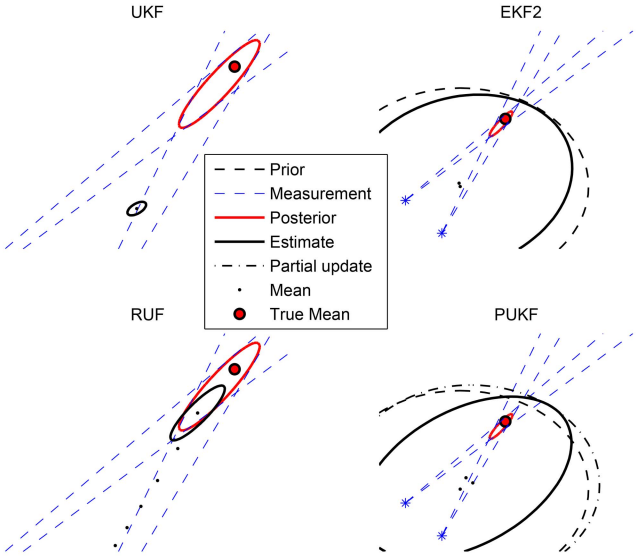



Fig. 7. Example of first update in bearings only tracking

that the PUKF and EKF2 have the best error estimates. Other methods have too optimistic covariance estimates. In this test scenario the PF did not manage to get good estimates with similar runtimes.

In the third test scenario we consider bearings only tracking with sensors close to each other. Otherwise the measurement model is the same as in the previous scenario The prior is as in previous test scenario. The state transition function is also (42) but the state transition noise is higher:

$$\varepsilon^x \sim \mathrm{N}\left(0, \begin{bmatrix} \frac{1}{3}I & \frac{1}{2}I \\ \frac{1}{2}I & I \end{bmatrix}\right). \tag{44}$$

The initial state and representative first updates are shown in Figure 7. In this Figure UKF and RUF estimates have very small covariances and so the plots are magnified. The UKF estimate mean is closer to the true mean than EKF2 and PUKF estimates, but the covariance of the estimate is very small. RUF has a better estimate than UKF, but the estimate is biased towards the sensor locations. Because both sensors are nearby and have large second order terms EKF2 and PUKF estimates do not differ much.

Results for estimating 10 step tracks 1000 times are shown in Figure 8. In this case the RUF has the best accuracy. In PUKF there is only very small differences whether all of the measurement are used at once or a nonlinearity threshold is used. This means that in this measurement geometry the partitioned update does not improve accuracy. EKF2 has better covariance estimates than the numerical update PUKF even though it has larger errors. The covariance estimates produced by RUF were again too small. In this test the PF has better accuracy than the Kalman filter extensions.

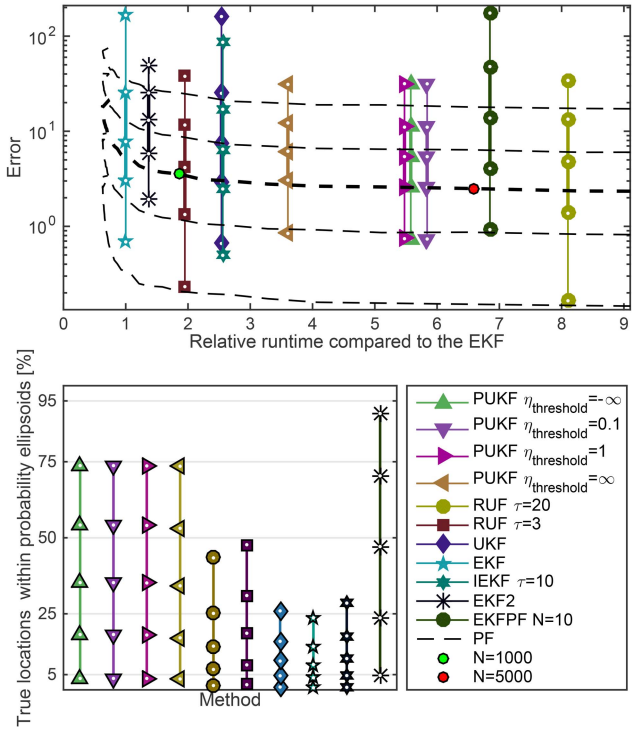



Fig. 8. Results for bearings only tracking with sensors close to each other

To further evaluate the accuracy of the estimates, we compare Kullback-Leiber (KL) divergences of estimates. The KL divergence is defined as

$$\int \ln\left(\frac{p(x)}{q(x)}\right) p(x)dx, \tag{45}$$

where $p(x)$ is the pdf of the true distribution and $q(x)$ is the pdf of the approximate distribution [20]. We computed the KL divergence for position dimensions. The true pdf is approximated using a $50 \times 50$ grid. The probability for each grid is computed as the sum of particle weights of a PF particles within each cell. For this we used $10^6$ particles. Table I shows the median Kullback-Leibler divergences for each method in the two bearings measurement test scenarios. PUKF has the smallest KL divergence in both test scenarios.

## V. CONCLUSIONS AND FUTURE WORK

In this paper we presented a new extension of the Kalman filter: the Partitioned Update Kalman Filter (PUKF). The proposed filter evaluates the nonlinearity of a multidimensional measurement and transforms the measurement model so that some dimensions of the measurement model have as low nonlinearity as possible. PUKF does the update of the state using the measurement in parts, so that the parts with the smallest amounts of nonlinearity are processed first. The proposed algorithm improves estimation results when measurements are such that the partial update reduces the nonlinearity of the remaining part. According to the simulated tests the PUKF improves the estimates when

TABLE I
Median Kullback-Leibler divergences of position dimensions in the two bearings only tests

| Method | PUKF $\eta_{\text{threshold}} = -\infty$ | PUKF $\eta_{\text{threshold}} = 0.1$ | PUKF $\eta_{\text{threshold}} = 1$ | PUKF $\eta_{\text{threshold}} = \infty$ | RUF $\tau = 20$ | RUF $\tau = 3$ | UKF | EKF | IEKF $\tau = 10$ | EKF2 |
|---|---|---|---|---|---|---|---|---|---|---|
| First bearings test | 0.62 | 0.62 | 0.63 | 1.07 | 0.99 | 1.70 | 5.20 | 14.42 | 6.69 | 1.16 |
| Second bearings test | 2.14 | 2.14 | 2.14 | 2.33 | 3.56 | 2.53 | 9.36 | 10.97 | 8.84 | 2.60 |

measurements can be transformed so that an informative linear part of the measurement can be extracted.

In many practical situations the almost linear part could be extracted manually. For example, Global Positioning System (GPS) measurements are almost linear and they could be applied before other measurements. The proposed algorithm does the separation automatically and when using the numerical algorithm for computing the prediction and update analytical differentiation is not required.

In our tests the estimated covariances produced by EKF2 and PUKF were the most accurate. In [11] it was claimed that RUF produces more accurate error estimates than EKF2. Their results were based on comparing $3 \cdot \sigma$ errors in 1D estimation. In this comparison 92% of samples should be within the $3 \cdot \sigma$ range. For their results they had only 100 samples and from the resulting figure it is hard to see how many samples exactly are within the range, but for EKF2 most of the points are within the range and some are outside.

In our tests, among other Kalman filter extensions RUF had good accuracy, but it provided too small covariance matrices. In future it could be interesting to extend RUF [11] to use EKF2-like statistical second order linearization and then combine it with the proposed algorithm.

Another use case for PUKF would be merging it with the Binomial Gaussian mixture filter [21]. This filter decorrelates measurements and uses nonlinearity measure (30) as an indication of whether the measurement model is so nonlinear that the prior component should be split. By decorrelating measurements with the algorithm proposed in this paper and doing the partial updates for the most linear components first, unnecessary splits could be avoided.

## APPENDIX A INVARIANCE OF EKF AND EKF2 TO A LINEAR TRANSFORMATION OF THE MEASUREMENT MODEL

The second order Taylor polynomial approximation of the measurement function is

$$h(x) = h(\mu^-) + J^h(x-\mu^-) + \frac{1}{2}\begin{bmatrix} (x-\mu^-)^T H^{h_1}(x-\mu^-) \\ (x-\mu^-)^T H^{h_2}(x-\mu^-) \\ \vdots \\ (x-\mu^-)^T H^{h_d}(x-\mu^-) \end{bmatrix} + \varepsilon^y \quad (46)$$

where Jacobian $J^h$ and Hessians $H^h$ are evaluated at prior mean, $\varepsilon^y$ is the measurement function noise.

In the linear transformation the measurement function (46) is multiplied by $D$. The second order approximation is

$$\hat{h}(x) = Dh(x) = Dh(\mu^-) + DJ^h(x-\mu^-) + \frac{1}{2}D\begin{bmatrix} (x-\mu^-)^T H_1^h(x-\mu^-) \\ (x-\mu^-)^T H_2^h(x-\mu^-) \\ \vdots \\ (x-\mu^-)^T H_n^h(x-\mu^-) \end{bmatrix} + D\varepsilon^y \quad (47)$$

The transformed Jacobian is

$$\hat{J}^h = DJ^h \quad (48)$$

and $i$th transformed Hessian is

$$\hat{H}_i^h = \sum_{k=1}^n D_{[i,k]} H_k^h. \quad (49)$$

The terms $\xi^h$ and $\Xi^h$ are

$$\begin{aligned} \hat{\xi}^h &= \begin{bmatrix} \operatorname{tr} P^- \hat{H}_1^h \\ \operatorname{tr} P^- \hat{H}_2^h \\ \vdots \\ \operatorname{tr} P^- \hat{H}_n^h \end{bmatrix} = \begin{bmatrix} \operatorname{tr} P^- \sum_{k=1}^n D_{[1,k]} H_k^h \\ \operatorname{tr} P^- \sum_{k=1}^n D_{[2,k]} H_k^h \\ \vdots \\ \operatorname{tr} P^- \sum_{k=1}^n D_{[n,k]} H_k^h \end{bmatrix} \\ &= D\begin{bmatrix} \operatorname{tr} P^- H_1^h \\ \operatorname{tr} P^- H_2^h \\ \vdots \\ \operatorname{tr} P^- H_n^h \end{bmatrix} = D\xi^h \end{aligned} \quad (50)$$

$$\begin{aligned} \hat{\Xi}_{[i,j]}^h &= \operatorname{tr} P^- \hat{H}_i^h P^- \hat{H}_j^h \\ &= \operatorname{tr} P^- \left(\sum_{k=1}^n D_{[i,k]} H_k^h\right) P^- \left(\sum_{l=1}^n D_{[i,l]} H_l^h\right) \\ &= \sum_{k=1}^n \sum_{l=1}^n D_{[i,k]} D_{[j,l]} \operatorname{tr} P^- H_k^h P^- H_l^h \\ \Rightarrow \hat{\Xi}^h &= D\Xi^h D^T \end{aligned} \quad (51)$$

For EKF update these terms are replaced with zero matrices.

Now using these transformed quantities in the EKF2 update equations (9–13) gives

$$\hat{y}^- = \hat{h}(\mu^-) + \tfrac{1}{2}\hat{\xi}^h = D(h(\mu^-) + \tfrac{1}{2}\xi^h) \quad (52)$$

$$\hat{S} = DJ^h P^- J^{h^T} D^T + \tfrac{1}{2} D\Xi^h D^T + DRD^T \quad (53)$$
$$= DSD^T$$
$$\hat{K} = P^- \hat{J}^{h^T} \hat{S}^{-1}$$
$$= P^- J^{h^T} D^T D^{-T} S^{-1} D^{-1} = P^- J^{h^T} S^{-1} D^{-1} \quad (54)$$
$$= KD^{-1}$$
$$\hat{\mu}^+ = \mu^- + \hat{K}[Dy - Dh(\mu^-) - \tfrac{1}{2} D\xi^h]$$
$$= \mu^- + K(y - J^h \mu^- - \tfrac{1}{2}\xi^h) \quad (55)$$
$$= \mu^+$$
$$\hat{P}^+ = P^- - \hat{K}\hat{S}\hat{K}^T$$
$$= P^- - KD^{-1} DSD^T (KD^{-1})^T = P^- - KSK^T$$
$$= P^+, \quad (56)$$

which shows that the posterior is the same as with the non-transformed measurements.

## APPENDIX B PROOF THAT THE NONLINEARITIES ARE MINIMIZED

Let $\Xi^h$ be a diagonal matrix containing nonlinearity values ordered ascending on the diagonal and let measurement noise covariance matrix be identity matrix $R = I$. We will show that the smallest diagonal element of $\Xi^h$ is as small as possible under a linear transformation that preserves $R = I$ and further that the second smallest diagonal element is as small as possible, when the smallest is as small as possible etc.

If the measurement model is transformed by multiplying it with $V$, the transformed variables are $\hat{\Xi}^h = V\Xi^h V^T$ and $R = VIV^T = VV^T$. Because we want to have $R = I$, $V$ has to be unitary. The $i$th diagonal element of the transformed matrix is $v_i^T \Xi^h v_i = \sum_{j=1}^d v_{i,[j]}^2 \Xi_{[j,j]}^h$, where $v_i$ is the $i$th column of $V$. Because $V$ is unitary, we have $\sum_{j=1}^d v_{i,[j]}^2 = 1$ and the $i$th diagonal element of the transformed matrix $\hat{\Xi}^h$ is

$$\sum_{j=1}^d v_{i,[j]}^2 \Xi_{[j,j]}^h \geq \sum_{j=1}^d v_{i,[j]}^2 \min_j \{\Xi_{[j,j]}^h\} = \min_j \{\Xi_{[j,j]}^h\}. \quad (57)$$

Thus, the new diagonal element cannot be smaller than the smallest diagonal element of $\Xi^h$.

If the smallest element is in the first element of the diagonal the possible transformation for the second smallest element is

$$\hat{\Xi}^h = \begin{bmatrix} 1 & 0^T \\ 0 & V \end{bmatrix} \Xi^h \begin{bmatrix} 1 & 0^T \\ 0 & V^T \end{bmatrix}. \quad (58)$$

With the same reasoning as given already the second diagonal has to be already the smallest possible. Inductively this applies to all diagonal elements.

## REFERENCES


[1] H. W. Sorenson and D. L. Alspach
Recursive Bayesian estimation using Gaussian sums,
*Automatica*, vol. 7, no. 4, pp. 465–479, 1971. doi: 10.1016/0005-1098(71)90097-5.

[2] J. H. Kotecha and P. Djuric
Gaussian particle filtering,
*IEEE Transactions on Signal Processing*, vol. 51, no. 10, pp. 2592–2601, Oct 2003. doi: 10.1109/TSP.2003.816758.

[3] J. Steinbring and U. Hanebeck
Progressive Gaussian filtering using explicit likelihoods,
in *17th International Conference on Information Fusion (FUSION)*, July 2014, pp. 1–8.

[4] A. Gelb
*Applied Optimal Estimation*.
MIT Press, 1974.

[5] K. Ito and K. Xiong
Gaussian filters for nonlinear filtering problems,
*IEEE Transactions on Automatic Control*, vol. 45, no. 5, pp. 910–927, May 2000. doi: 10.1109/9.855552.

[6] S. J. Julier and J. K. Uhlmann
A New Extension of the Kalman Filter to Nonlinear Systems,
in *International Symposium Aerospace/Defense Sensing, Simulation and Controls*, vol. 3068, 1997. doi: 10.1117/12.280797 pp. 182–193.

[7] I. Arasaratnam and S. Haykin
Cubature Kalman filters,
*IEEE Transactions on Automatic Control*, vol. 54, no. 6, pp. 1254–1269, June 2009. doi: 10.1109/TAC.2009.2019800.

[8] S. Dmitriev and L. Šimelevič
A generalized Kalman filter with repeated linearization and its use in navigation over geophysical fields,
*Avtomatika i Telemekhanika*, no. 4, pp. 50–55, 1978.

[9] A. H. Jazwinski
*Stochastic Processes and Filtering Theory*,
ser. Mathematics in Science and Engineering. Academic Press, 1970, vol. 64.

[10] A. Garcia-Fernandez, L. Svensson, and M. Morelande
Iterated statistical linear regression for Bayesian updates,
in *17th International Conference on Information Fusion (FUSION)*, July 2014, pp. 1–8.

[11] R. Zanetti
Recursive update filtering for nonlinear estimation,
*IEEE Transactions on Automatic Control*, vol. 57, no. 6, pp. 1481–1490, June 2012. doi: 10.1109/TAC.2011.2178334.

[12] M. Nørgaard, N. K. Poulsen, and O. Ravn
New developments in state estimation for nonlinear systems,
*Automatica*, vol. 36, no. 11, pp. 1627–1638, Nov. 2000. doi: 10.1016/S0005-1098(00)00089-3.

[13] R. F. Stengel
Stochastic optimal control,
*Theory and Application*, pp. 1–638, 1986.

[14] S. Ali-Löytty
Box Gaussian mixture filter,
*IEEE Transactions on Automatic Control*, vol. 55, no. 9, pp. 2165–2169, September 2010. doi: 10.1109/TAC.2010.2051486.

[15] M. Raitoharju
"Linear models and approximations in personal positioning,"
Ph.D. dissertation, Tampere University of Technology, 2014. [Online]. Available: http://urn.fi/URN:ISBN:978-952-15-3421-8.

[16] M. Raitoharju and S. Ali-Löytty
An adaptive derivative free method for Bayesian posterior approximation,
*IEEE Signal Processing Letters*, vol. 19, no. 2, pp. 87–90, Feb 2012. doi: 10.1109/LSP.2011.2179800.

[17] M. Raitoharju
"Matlab implementation of the partitioned update Kalman filter,"
http://www.mathworks.com/matlabcentral/fileexchange/51838-partitioned-update-kalman-filter, 2015, [Online; accessed 30-June-2015].

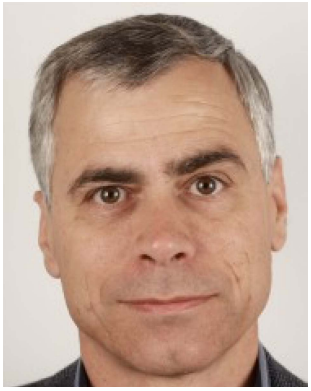
[18] J. Carpenter, P. Clifford, and P. Fearnhead
Improved particle filter for nonlinear problems,
*IEE Proceedings—Radar, Sonar and Navigation*, vol. 146, pp. 2–7, February 1999. doi: 10.1049/ip-rsn:19990255.

[19] J. F. de Freitas, M. Niranjan, A. H. Gee, and A. Doucet
Sequential Monte Carlo methods to train neural network models,
*Neural computation*, vol. 12, no. 4, pp. 955–993, 2000.

[20] S. Kullback and R. A. Leibler
On information and sufficiency,
*Ann. Math. Statist.*, vol. 22, no. 1, pp. 79–86, 1951.

[21] M. Raitoharju, S. Ali-Löytty, and R. Piché
Binomial Gaussian mixture filter,
*EURASIP Journal on Advances in Signal Processing*, vol. 2015, no. 1, p. 36, 2015. doi: 10.1186/s13634-015-0221-2.

**Matti Raitoharju** received M.Sc. and Ph.D. degrees in Mathematics in Tampere University of Technology, Finland, in 2009 and 2014 respectively. He works as a postdoctoral researcher in the Department of Automation Science and Engineering at Tampere University of Technology. His research interests include estimation algorithms and mathematical modelling.



**Robert Piché** received the Ph.D. degree in civil engineering in 1986 from the University of Waterloo, Canada. Since 2004 he is professor at Tampere University of Technology, Finland. His scientific interests include mathematical and statistical modelling, systems theory, and applications in positioning, computational finance, and mechanics.

**Juha Ala-Luhtala** received the M.Sc. (Tech.) degree in information technology from Tampere University of Technology in 2011. He is currently working towards finishing his Ph.D. degree in mathematics at the same university. His research interests include Bayesian inference in stochastic state-space models, and applications in positioning and navigation.

**Simo Ali-Löytty** is a University Lecturer at Tampere University of Technology. He received his M.Sc. degree in 2004 and his Ph.D. degree in 2009 from Tampere University of Technology. His research interests are mathematical modelling and industrial mathematics especially Bayesian filters in personal positioning.