Partitioned Update Kalman Filter

Matti Raitoharju*, Robert Piché*, Juha Ala-Luhtala†, Simo Ali-Löytty†
{matti.raitoharju, robert.piche, juha.ala-luhtala, simo.ali-loyty}@tut.fi

Abstract—In this paper we present a new Kalman filter extension for state update called Partitioned Update Kalman Filter (PUKF). PUKF updates state using multidimensional measurements in parts. PUKF evaluates the nonlinearity of the measurement function within Gaussian prior by comparing the innovation covariance caused by the second order linearization to 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 applied then 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 measurement geometry allows effective partitioning, the proposed algorithm improves estimation accuracy and produces accurate covariance estimates.

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

$$x_t = f_t(x_{t-1}, \varepsilon^x_t),$$

where $f_t$ is the state transition function at time index $t$ and $\varepsilon^x_t$ is the state transition noise. The state estimate is updated using measurements that are modeled as

$$y_t = h_t(x_t, \varepsilon^y_t),$$

where $h_t$ is a measurement function and $\varepsilon^y_t$ is the measurement noise. If the measurement and state transition are linear, noises are additive and Gaussian so that (1-2) take the form

$$x_t = f_t(x_{t-1}) + \varepsilon^x_t,$$

$$y_t = h_t(x_t) + \varepsilon^y_t,$$

where $\varepsilon^x_t \sim N(0, W_t)$, $W_t$ is the state transition noise covariance, $\varepsilon^y_t \sim N(0, R_t)$, 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 (grid and particle filters)

2) Approximate probabilities by Gaussians (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 usually 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.

Some algorithms do linearizations iteratively to improve them. In the Iterated Kalman Filter (IKF) a Kalman update is computed and then the new mean is used as the new linearization point [6]. This can be done several times. The Recursive Update Filter (RUF) updates the prior with measurement with reduced weight several times [7]. 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 measurements. The nonlinearity measure is based on comparing the effect of second order
term on linearization and the Gaussian measurement noise. Computation of this nonlinearity measure requires the same matrices as the EKF2 update and for this we use the derivative-free algorithm for computing EKF2 presented in [3]. PUKF applies a linear transformation to the measurement to make a new measurement that has linearly independent measurement noise for measurement elements and the smallest nonlinearity corresponding to a measurement element minimized, then the second smallest nonlinearity etc. 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 become smaller. The remaining measurements’ nonlinearity is re-evaluated using partially updated state. 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.

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 a Particle Filter (PF) in Section IV. Section V concludes the article.

II. EKF2 AND ITS NUMERICAL UPDATE

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 (8):

\[ \mu_t^- = f_t(\mu_{t-1}^+) + \frac{1}{2} \xi_f \]
\[ P_t^- = J_f P_{t-1}^+ J_f^T + \frac{1}{2} \Xi_f + W_t, \]

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 \( \xi_f \) and \( \Xi_f \) are defined as

\[ \xi_{[i]} = \text{tr} P_t^- H_i^h \]
\[ \Xi_{[i,j]} = \text{tr} P_t^- H_i^h P_t^- H_j^h, \]

where \( H_i^h \) 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 (8)

\[ y^- = h(\mu^-) + \frac{1}{2} \xi_h \]
\[ S = J_h P^- H_h^T + \frac{1}{2} \Xi_h + R \]
\[ K = P^- H_h^T S^{-1} \]
\[ \mu^+ = \mu^- + K(y^- - y^-) \]
\[ P^+ = P^- - K S K^T, \]

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

\[ \xi_{[i]} = \text{tr} P^- H_i^h \]
\[ \Xi_{[i,j]} = \text{tr} P^- H_i^h P^- H_j^h, \]

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

In this paper we use a numerical algorithm for computing the EKF2 update that is equivalent to one presented in [3]. Let \( \sqrt{P^-} \) be a matrix such that

\[ \sqrt{P^-} \sqrt{P^-}^T = P^- \]

In our implementation the 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 \( \Delta_i = \gamma \sqrt{P^-[i,i]} \), where \( \gamma \) is an algorithm parameter that defines the spread of the function evaluations. Matrix \( M \), whose elements are

\[ M_{[i,j]} = \begin{bmatrix} J_h \sqrt{P^-} \end{bmatrix}_{[i,j]} \approx \gamma^{-1} h(\mu^- + \Delta_i) - h(\mu^- - \Delta_i) \]

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

\[ Q_{k[i,j]} = \gamma^{-2} \begin{bmatrix} h[|k|](\mu^- + \Delta_i) + h[|k|](\mu^- + \Delta_i) - 2h[|k|](\mu^-) \end{bmatrix} \]
\[ Q_{k[i,j]} = \gamma^{-2} \begin{bmatrix} h[|k|](\mu^- + \Delta_i) + h[|k|](\mu^- + \Delta_i) - h[|k|](\mu^- + \Delta_i) + h[|k|](\mu^-) \end{bmatrix}, i \neq j. \]

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

\[ \xi_{[i]} = \text{tr} P^- H_i^h \approx \text{tr} Q_i \]
\[ J_h P^- J_h^T \approx M M^T \]
\[ P^- J_h^T \approx \sqrt{P^-} M^T \]
\[ \Xi_{[i,j]} = \text{tr} P^- H_i^h P^- H_j^h \approx \text{tr} Q_i Q_j \]

The prediction step can be approximated by computing \( Q_f \) matrices (19) using the state transition function instead of the measurement function and doing the following substitutions:

\[ \text{tr} P^- H_i^h \approx \text{tr} Q_i^f \]
\[ \text{tr} P^- H_i^h P^- H_j^h \approx \text{tr} Q_i^f Q_j^f \]

In [9] an update algorithm similar to numerical EKF2 is proposed that uses only the diagonal elements of \( Q \) matrices
is presented. 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. PARTIAL UPDATE KALMAN FILTER

When measurements are linear and the measurement covariance is block diagonal the posterior estimate of the Kalman filter does not change if the state is updated one block at a time. 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 covariance $R$ is not diagonal a linear transformation (decorrelation) that transforms the measurement so that the transformed measurement has diagonal covariance $[10]$. In PUKF, we choose this decorrelation so that the nonlinearity measure 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 $\hat{\zeta}^h$ with the covariance of the measurement noise:

$$\eta = \text{tr} \sum_{i=1}^{d} \sum_{k=1}^{d} R_{(i,k)}^{-1} \hat{P} H^h_i P \hat{H}_k^h$$

(26)

This nonlinearity measure is a local approximation of the nonlinearity and is developed from the measure presented in $[11, 8]$. In $[12]$ 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 using a Kalman filter extension. Nonlinearity measure (26) can be approximated numerically using substitution (23). Numerical computation of a similar nonlinearity measure was proposed in $[13]$, but the algorithm presented in Section III does the nonlinearity computation with fewer measurement function evaluations.

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

$$D y = D h(x) + D e_y.$$ 

(27)

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

$$\hat{R} = I \wedge \text{tr} P \hat{H}_i^h P \hat{H}_k^h = 0, i \neq k,$$

(28)

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

In $[12]$, it was shown that when a measurement is transformed so that $\hat{R} = I$ the nonlinearity measure (26) simplifies to

$$\eta = \text{tr} \sum_{i=1}^{d} \sum_{k=1}^{d} R_{(i,k)}^{-1} \hat{P} H^h_i P \hat{H}_k^h = \sum_{i=1}^{d} \eta_i,$$

(29)

where

$$\eta_i = \text{tr} P \hat{H}_i^h P \hat{H}_i^h$$

(30)

is the nonlinearity measure of the $i$th transformed measurement element.

In Appendix A it is shown that

$$\hat{\zeta}_{[i,j]}^h = [D \hat{\zeta}^h D^T]_{[i,j]} \approx \text{tr} P \hat{H}_i^h P \hat{H}_j^h.$$

(31)

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

In PUKF nonlinearities are minimized so that $\eta_1$ (30) is as small as possible. Then $\eta_2$ is minimized so 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 covariance

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

(32)

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

$$U \Xi U^T = \sqrt{R}^{-1} \Xi \sqrt{R}^T.$$

(33)

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

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

(34)

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

After the measurements are decorrelated (multiplied with $D$), the part of measurements that have low nonlinearity ($\lambda_{[i,j]} \leq \eta_{\text{threshold}}$) are used in the EKF2 update step (Section III). If none of measurements fulfill this requirement then the most linear measurement is used to update the state. Then the same process is repeated for the remaining transformed measurements until all measurements are processed. The PUKF algorithm is presented in Algorithm 1.

IV. TESTS

We compare the proposed PUKF with other Kalman filter extensions and a PF in three different simulation setups. The PUKF was tested with 4 different values for $\eta_{\text{threshold}}$. When $\eta_{\text{threshold}} = \infty$ all measurements are updated at once and the algorithm is a numerical EKF2. When $\eta_{\text{threshold}} < 0$ measurements 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
input : Prior state: $\mu$ – mean $P$ – covariance  
Measurement: $y$ – value, $h(\cdot)$ – function,  
$R$ – covariance  
output : Updated state: $\mu$ – mean, $P$ – covariance  
parameters: $\eta_{\text{threshold}}$ – nonlinearity limit, $\gamma$ –  
measurement function evaluation spread  
\[(default \ \gamma = \sqrt{3})\]

1 Compute $\sqrt{R}$  
2 $d \leftarrow$ measurement dimension  
while $d > 0$ do  
3 Compute $\sqrt{F}$  
4 Compute $M$ and $Q_i, 1 \leq i \leq d$  
5 Compute $\xi^h$ and $\Xi^h$  
6 Compute $U$ and $\Lambda$  
7 $D \leftarrow U^T \sqrt{R}^{-1}$  
8 Choose largest $k$ so that  
\[\Lambda_{[1,i]} \leq \eta_{\text{threshold}}, i \leq k \land \Lambda_{[j,j]} > \eta_{\text{threshold}}, j > k\]
9 if $k = 0$ then  
10 $k \leftarrow 1$
11 end  
12 // Compute partial EKF2 update  
13 $y^- \leftarrow D_{[1:k,:]} [h(\mu) + \frac{1}{2} \xi^h]$  
14 $S \leftarrow [D_{[1:k,:]} M^T D_{[1:k,:]}^T] + \frac{1}{2} \Lambda_{[1:k,1:k]} + I$  
15 $K \leftarrow \sqrt{S}^{-1} M^T D_{[1:k,:]}^T$  
16 $\mu \leftarrow \mu + K (y - y^-)$  
17 $P \leftarrow P - K \Sigma K^T$  
18 // Update remaining measurement  
19 $y \leftarrow D_{[k+1:d,:]} y$  
20 $h(x) \leftarrow D_{[k+1:d,:]} h(x)$  
21 $\sqrt{R} \leftarrow I$ // Updated measurement  
22 $d \leftarrow d - k$ // Updated measurement  
23 dimension  
24 end  

Algorithm 1: Algorithm for doing the measurement update  
in PUKF

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 11  
with analytical Jacobians and Hessians. RUF is implemented  
according to [7] with 3 and 10 steps. UKF 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 the  
with similar levels of code optimizations, but the runtimes  
should still be considered to be only indicative. The PF  
used here is a bootstrap particle filter that does systematic  
resampling at every time step [14].

In every test scenario the state transition model is linear  
time-invariant $x_t = f(x_{t-1} + \varepsilon^x)$, where $\varepsilon^x \sim N(0, W)$. 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 chosen to be very well suited for  
PUKF. The measurement model used is

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

where $\varepsilon^y \sim N(0, I)$. The first three elements of the  
transformed measurement model are linear and PUKF with  
$\eta_{\text{threshold}} \in \{0.1, 1\}$ uses the three linear measurements 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 (35) are  
presented in Figure 2. The markers in the figure on top shows the  
5%, 25%, 50%, 75% and 95% quantiles of mean errors  
for each method. The quantiles are computed from 1000 runs.
is, a true location is within the 50%, 75% and 95% ellipsoids of the Gaussian posterior. That we compute how often the true state is within the 5%, 25%, different Kalman filter extensions are compared. For this plot PUKF also outperforms PF with similar runtime.

methods that use EKF linearizations have very large errors. In this test scenario the PUKF performs clearly the best and margin. When the most accurate of the Kalman filter extensions by a large density function of the chi-squared distribution with degrees of freedom. The filter’s error estimate is reliable when markers covariance computed by the filter, and \( \chi^2_n \) 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.

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 is nonlinear, but when the target is far away the measurement becomes almost linear.

The measurement model is

\[
y = \arctan2(x[2] - r[2], x[1] - r[1]) + \varepsilon_y,
\]

where \( \arctan \) is the four quadrant inverse tangent and \( r \) is the sensor location. Because angles are periodic in the evaluation of measurement function an integer multiple of \( 2\pi \) is added to the measurement function values to make each measurement value 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 3. 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 from the nearby sensor is so large that EKF2 almost completely ignores that measurement and the prior is updated only using the measurement from the distant sensor. The iterative 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 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.
Fig. 4. Accuracy of different Kalman filter in bearings only tracking

Figure 4 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

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

where

$$\varepsilon x \sim N \left( 0, \begin{bmatrix} \frac{1}{2000}I & \frac{1}{2000}I \\ \frac{1}{2000}I & \frac{1}{2000}I \end{bmatrix} \right).$$

Figure 4 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 that the PUKF that does the update in parts has the most reliable error estimates while EKF2 and PUKF with $\eta_{\text{threshold}} = \infty$ have almost as good 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. The prior is as in previous test scenario. The state transition function is also (39) but the state transition noise is higher:

$$\varepsilon x \sim N \left( 0, \begin{bmatrix} \frac{1}{300}I & \frac{1}{300}I \\ \frac{1}{300}I & \frac{1}{300}I \end{bmatrix} \right).$$

The initial state and representative first updates are shown in Figure 5. 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 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 6. 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 had larger errors. The covariance estimates produced by RUF were again too small.

Fig. 5. Example of first update in bearings only tracking

Fig. 6. Accuracy of different Kalman filter in bearings only tracking

V. CONCLUSIONS AND FUTURE WORK

In this paper we presented a new extension of the Kalman filter: Partitioned Update Kalman Filter (PUKF). The proposed filter evaluates nonlinearity of a multidimensional measurement and transforms the measurement so that some dimensions of the measurement have as low nonlinearity as possible. PUKF does the update of the state using the measurement in parts, so that parts with 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 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 e.g. 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 [7] 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 \cite{7} to use EKF2-like statistical second order linearization and then fuse it with the proposed algorithm.

Another use case for PUKF is merging it with Binomial Gaussian mixture filter \cite{15}. Binomial Gaussian mixture filter decorrelates measurements and uses nonlinearity measure \cite{30} as indication whether the measurement is so nonlinear that the prior component should be split. By decorrelating measurements with algorithm proposed in this paper and doing the partial updates for the most linear components first unnecessary splits could be avoided.

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)90007-5
[2] A. Gelb, Applied Optimal Estimation. MIT Press, 1974.
[3] K. Ito and K. Xiong, “Gaussian filters for nonlinear filtering problems.” Automatic Control, IEEE Transactions on, vol. 45, no. 5, pp. 910–927, May 2000. doi: 10.1109/9.855552
[4] 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.
[5] 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
[6] B. Bell and F. Cathey, “The iterated Kalman filter update as a Gauss-Newton method,” Automatic Control, IEEE Transactions on, vol. 38, no. 2, pp. 294–297, Feb 1993. doi: 10.1109/9.250476
[7] R. Zanetti, “Recursive update filtering for nonlinear estimation,” Automatic Control, IEEE Transactions on, vol. 57, no. 6, pp. 1481–1490, June 2012. doi: 10.1109/TAC.2011.2178334
[8] A. H. Jazwinski, Stochastic Processes and Filtering Theory, ser. Mathematics in Science and Engineering. Academic Press, 1970, vol. 64.
[9] M. Raitoharju, 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
[10] R. F. Stengel, “Stochastic optimal control,” Theory and Application, pp. 1–638, 1986.
[11] 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
[12] 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
[13] M. Raitoharju and S. Ali-Löytty, “An adaptive derivative free method for Bayesian posterior approximation,” Signal Processing Letters, IEEE, vol. 19, no. 2, pp. 87–90, Feb 2012. doi: 10.1109/LSP.2011.2179800
[14] 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
[15] M. Raitoharju, S. Ali-Löytty, and R. Piché, “Binomial Gaussian mixture filter,” EURASIP Journal on Advances in Signal Processing, submitted.

APPENDIX A

INvariance of EKF and EKF2 to a Linear Transformation of the Measurement

The second order Taylor polynomial approximation of the measurement function is

\[ h(x) = h(\mu^\circ) + J^h(x - \mu^\circ) + \frac{1}{2} D \left[ (x - \mu^\circ)^T H^h_1 (x - \mu^\circ)^T \right] + \epsilon^y \]  

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

In linear transformation the measurement function \( h(x) \) is multiplied by \( D \). The second order approximation is

\[ \hat{h}(x) = Dh(x) = D h(\mu^\circ) + D J^h(x - \mu^\circ) + \frac{1}{2} D \left[ (x - \mu^\circ)^T H^h_1 (x - \mu^\circ)^T \right] + D \epsilon^y \]

The transformed Jacobian is

\[ J^h = DJ^h \]

and \( i \) th transformed Hessian is

\[ \hat{H}^h_i = \sum_{k=1}^{n} D_{i,k} H^h_{k} \]
The terms $\xi^h$ and $\Xi^h$ are
\[
\xi^h = \begin{bmatrix}
\text{tr} P^{-1} \hat{H}^h_1 \\
\text{tr} P^{-1} \hat{H}^h_2 \\
\vdots \\
\text{tr} P^{-1} \hat{H}^h_n
\end{bmatrix} = D \xi^h
\]

\[
\Xi^h = \begin{bmatrix}
\text{tr} P^{-1} \sum_{k=1}^n D_{[1,k]} H_k^h \\
\text{tr} P^{-1} \sum_{k=1}^n D_{[2,k]} H_k^h \\
\vdots \\
\text{tr} P^{-1} \sum_{k=1}^n D_{[n,k]} H_k^h
\end{bmatrix} = D \Xi^h
\]

For EKF update these terms are replaced with zero matrices.

Now using these transformed quantities in EKF2 update equations (46) gives
\[
\hat{y} = \hat{h}(\mu^-) + \frac{1}{2} \xi^h = D \left( h(\mu^-) + \frac{1}{2} \xi^h \right)
\]

\[
\hat{S} = D J^h P^- J^h^T D^T + \frac{1}{2} D \Xi^h D^T + D R D^T
\]

\[
\hat{K} = P^- J^h^T D^T
\]

\[
\hat{\mu}^+ = \mu^- + \hat{K} \left[ D y - D h(\mu^-) - \frac{1}{2} D \xi^h \right]
\]

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

\[
\hat{P}^+ = P^- - \hat{K} \hat{S} \hat{K}^T = P^- - K D^{-1} D S D^T (K D^{-1})^T = P^- - K S K^T = P^+,
\]

\section*{APPENDIX B}
\textbf{PROOF THAT THE NONLINEARITIES ARE MINIMIZED}

Let $\Xi^h$ be a diagonal matrix containing nonlinearity values ordered ascending on the diagonal and let measurement 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 measurement is transformed by multiplying it with $V$, the transformed variables are $\hat{\Xi}^h = V \Xi^h V^T$ and $R = V I V^T = V V^T$. Because we want to have $R = I$, $V$ has to be unitary. $i$th diagonal element of the transformed matrix is $v_i^T \hat{\Xi}^h v_i = \sum_{j=1}^d v_i^T \Xi_{[i,j]}^h v_j$, where $v_i$ is the $i$th column of $V$. Because $V$ is unitary $\sum_{j=1}^d v_i^2_{[i,j]} = 1$ and the $i$th diagonal element of the transformed matrix $\hat{\Xi}^h$ is

\[
\hat{\Xi}^h_{[i,i]} = \sum_{j=1}^d v_i^2_{[i,j]} \Xi^h_{[i,j]} \geq \sum_{j=1}^d v_i^2_{[i,j]} \min_j \{\Xi^h_{[j,j]} \} = \min_j \{\Xi^h_{[j,j]} \}.
\]

Thus, the new diagonal element cannot be smaller than the smallest diagonal element of $\Xi^h$. If the smallest element is in the first diagonal the possible transformation for the second smallest element is

\[
\hat{\Xi}^h = \begin{bmatrix}
1 & 0^T \\
0 & V^T
\end{bmatrix} \Xi^h \begin{bmatrix}
1 & 0^T \\
0 & V^T
\end{bmatrix}.
\]

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