(Preprint) AAS 23-321

RECURSIVE UPDATE FILTERING: A NEW APPROACH

Kristen A. Michaelson[*], Andrey A. Popov[†], and Renato Zanetti[‡]

Maintaining an accurate statistical representation of the states and uncertainty in a dynamical system is a difficult task. For nonlinear systems, it is generally computationally intractable. The Extended Kalman Filter (EKF) assumes a Gaussian distribution and provides a state estimate and associated covariance. Recursive extensions to the EKF, such as the Iterated EKF (IEKF) and the Recursive Update Filter (RUF), improve the accuracy of the EKF with little additional computational cost. In this article, we present the Bayesian Recursive Update Filter (BRUF), a variant of the RUF based on Bayesian principles. The main advantage of the BRUF over the RUF is that it does not require the cross-covariance between the state error and the measurement noise. The BRUF is demonstrated in two examples. We show that even with relatively few recursions, the proposed update is significantly more accurate and consistent than the EKF update.

INTRODUCTION

The Kalman filter is a state estimation algorithm that provides the optimal solution for linear systems subject to additive Gaussian noise.[1,2] The Extended Kalman Filter (EKF) extends the algorithm to nonlinear applications by using dynamics and measurement Jacobians in lieu of linear terms in the Kalman filter equations.[3] While the EKF is sufficient for numerous practical applications, severe nonlinearities may lead to divergence. A plethora of nonlinear estimation algorithms have been developed to mitigate divergence in highly nonlinear systems.[4]

In practice, designers face a tradeoff between accuracy and computational complexity when choosing a nonlinear estimator. Algorithms with lower computational complexity are preferred in autonomous navigation applications, since they are suited to real-time operation.[5] The EKF is very computationally efficient. It performs exactly the same handful of computations every time a measurement is received. In this work, we detail a novel *iterative* method based on the EKF: the Bayesian Recursive Update Filter (BRUF). In general, iterative methods repeat EKF computations in order to improve the accuracy of the measurement update. If the number of repetitions is low, then the computational complexity is comparable to that of the EKF.

The Iterated Extended Kalman Filter (IEKF) poses the measurement update as a weighted least squares problem and computes the solution in an iterative fashion.[6] The first measurement update $\Delta\mathbf{x}^{(1)}$ is equivalent to the EKF update. This update is applied to the prior state estimate, $\hat{\mathbf{x}}^-$. Then, the measurement Jacobian is recomputed using the updated state values $\hat{\mathbf{x}}^- + \Delta\mathbf{x}^{(1)}$. A second EKF

[*]Graduate Research Assistant, Department of Aerospace Engineering and Engineering Mechanics, Cockrell School of Engineering, 2617 Wichita Street, C0600, Austin, Texas 78712.

[†]Post-doctoral Researcher, Oden Institute for Computational Engineering and Sciences, 201 E 24th St, C0200 Austin, Texas 78712. Austin, Texas 78712.

[‡]Assistant Professor, Department of Aerospace Engineering and Engineering Mechanics, Cockrell School of Engineering, 2617 Wichita Street, C0600, Austin, Texas 78712.

update, $\Delta\mathbf{x}^{(2)}$, is computed using the new measurement Jacobian. This new update is again applied to the prior state estimate, $\hat{\mathbf{x}}^{-} + \Delta\mathbf{x}^{(2)}$, and the result is used to compute a third measurement Jacobian, and so on, until there is little change in the updated state values between iterations. This approach is equivalent to solving the optimization problem with the Gauss-Newton method.[7] The IEKF update is $\hat{\mathbf{x}}^{+} = \hat{\mathbf{x}}^{-} + \Delta\mathbf{x}^{(n)}$, where n is the number of iterations it takes to meet the optimality condition; ultimately, only the final state update is applied.

Instead of applying a single, "optimal" EKF update, the Recursive Update Filter (RUF) applies a fixed number of Kalman updates for each measurement. These N updates are applied successively.[8] The first update $\Delta\mathbf{x}^{(1)}$ is equivalent to the EKF update reduced by a scaling factor. The new state estimate is $\hat{\mathbf{x}}_1 = \hat{\mathbf{x}}^{-} + \Delta\mathbf{x}^{(1)}$. Next, $\Delta\mathbf{x}^{(2)}$ is computed using $\hat{\mathbf{x}}_1$ *and applied to* $\hat{\mathbf{x}}_1$. However, since $\hat{\mathbf{x}}_1$ is now acting as the prior state estimate, the measurement noise and state error are correlated. Therefore, this correlation (while trivial to compute) must be accounted for in the EKF equations.[9] Now $\hat{\mathbf{x}}_2 = \hat{\mathbf{x}}_1 + \Delta\mathbf{x}^{(2)}$, and so on, such that the measurement Jacobian is recomputed at each step using the updated state estimate from the previous step. In this way, the state estimate is herded toward the optimal value. The updated RUF state estimate is $\hat{\mathbf{x}}^{+} = \hat{\mathbf{x}}_N$.

The RUF approximates the maximum *a posteriori* (MAP) estimate using gradient descent optimization.[8] The MAP estimate is the state that maximizes the Bayesian posterior distribution; the product of the prior distribution and the measurement likelihood adjusted by a scaling factor. In the linear Kalman filter, the updated state is the minimum mean square error (MMSE) state estimate. The MMSE is mean of the Bayesian posterior distribution. For linear systems with additive Gaussian noise, the MAP estimate and the MMSE estimate coincide. One advantage of the RUF over the IEKF is the constant number of iterations. The RUF performs N Kalman updates to reach the MAP approximation each time a measurement is received, making it N times as costly as the EKF. In contrast, the IEKF continues iterations until a convergence criterion is met.

Numerous other iterative update schemes have been proposed. One method applies successive measurement updates using a weighted Kalman gain, like the RUF.[10] An information geometric optimization method presented in Li et al. (2017) may be thought of as hybrid between the RUF and the IEKF; the state moves toward the updated state value at each step, but recursions do not finish until a convergence criterion is met.[11] The Adaptable RUF chooses the length of the ith state update such that it is contained within a region where the linearization assumption is valid.[12] A single EKF update may be chosen if the linearization assumption holds for a state update of small magnitude.

Another class of methods uses recursive update techniques in sigma point filters, Gaussian mixtures, and particle filters.[13,14] In 2003, Hanebeck et al. introduced Progressive Bayes, which computes the measurement update by minimizing the squared-integral distance between the true Bayesian posterior and an approximation density.[15,16] In a follow-up to Progressive Bayes, the Progressive Gaussian Filter (PGF 42), the state estimate also undergoes successive measurement updates of varying step length.[17,18] PGF 42 represents the prior density using a Dirac mixture. The "progression schedule" of the update steps is determined by comparing trial forward updates and the corresponding backward updates. The RUF and the method proposed in this work may be thought of as single-particle implementations of particle flow; in the sense that the state estimate "flows" from the prior toward the maximum of the true Bayesian posterior. Wu et al. (2018) presents an iterative EKF update derived from Daum-Huang particle flow.[19–21]

We propose a novel Recursive Update Filter, the Bayesian Recursive Update Filter (BRUF), which defines the recursion based on Bayesian principles rather than the MAP. The main advantage of the proposed method over the RUF is that it does not include terms for the cross-covariance between the state estimate and the measurement noise. We achieve this by re-expressing the Bayesian posterior distribution as a product of the prior distribution and N likelihood distributions, each with inflated covariance NR. The new expression obviates the need to compute cross-covariance terms while maintaining the recursive recomputation of the measurement Jacobian, an attractive quality of iterative methods. The proposed update equations are a simple modification of the standard EKF equations. The only term in the new update scheme that does not appear in the EKF equations is N, the number of iterations. N is a constant integer parameter chosen by the designer.

PROPOSED METHOD

Assume a measurement has additive Gaussian noise,

$$\mathbf{y} = h(\mathbf{x}^*) + \boldsymbol{\eta}, \tag{1}$$

where $\boldsymbol{\eta} \sim \mathcal{N}(0, R)$, and $\mathbf{x}^*$ is the true state. A prior estimate of $\mathbf{x}^*$, $\hat{\mathbf{x}}^-$, is available with distribution $p(\mathbf{x})$. The Bayesian posterior distribution,

$$p(\mathbf{x}|\mathbf{y}) \sim p(\mathbf{x}) \exp\left(-\frac{1}{2}(\mathbf{y} - h(\mathbf{x}))^T R^{-1} (\mathbf{y} - h(\mathbf{x}))\right), \tag{2}$$

is proportional to the product of the prior distribution and the measurement likelihood. We would like to apply the measurement update gradually, keeping the prior distribution the same. To this end, we re-express Eq. 2 as the product of N likelihood equations:

$$p(\mathbf{x}|\mathbf{y}) \sim p(\mathbf{x}) \prod_{i=1}^{N} \exp\left(-\frac{1}{2N}(\mathbf{y} - h(\mathbf{x}))^T R^{-1} (\mathbf{y} - h(\mathbf{x}))\right), \tag{3}$$

which is equivalent to increasing the measurement noise covariance by a factor of N:

$$p(\mathbf{x}|\mathbf{y}) \sim p(\mathbf{x}) \prod_{i=1}^{N} \exp\left(-\frac{1}{2}(\mathbf{y} - h(\mathbf{x}))^T (NR)^{-1} (\mathbf{y} - h(\mathbf{x}))\right). \tag{4}$$

In a linear Gaussian system, the prior distribution is Gaussian and $p(\mathbf{x}) \sim \mathcal{N}(\hat{\mathbf{x}}^-, P)$. If the measurement is also linear, then $\mathbf{y} = H\mathbf{x}^* + \boldsymbol{\eta}$. The Kalman gain is

$$K = PH^T(HPH^T + R)^{-1}. \tag{5}$$

The updated state is

$$\hat{\mathbf{x}}^+ = \hat{\mathbf{x}}^- + K(\mathbf{y} - H\hat{\mathbf{x}}^-), \tag{6}$$

and the updated covariance is

$$P^+ = (I - KH)P. \tag{7}$$

For a linear system, Eqs. 6-7 could be "broken up" into any number of smaller Kalman updates based on the principle demonstrated in Eq. 4.

$$\begin{aligned}
\hat{\mathbf{x}}_1 &= \hat{\mathbf{x}}^- + PH^T(HPH^T + NR)^{-1}(\mathbf{y} - H\hat{\mathbf{x}}^-) \\
P_1 &= (I - PH^T(HPH^T + NR)^{-1}H)P \\
\hat{\mathbf{x}}_2 &= \hat{\mathbf{x}}_1 + P_1H^T(HP_1H^T + NR)^{-1}(\mathbf{y} - H\hat{\mathbf{x}}_1) \\
P_2 &= (I - P_1H^T(HP_1H^T + NR)^{-1}H)P_1 \\
&\vdots \\
\hat{\mathbf{x}}_N &= \hat{\mathbf{x}}_{N-1} + P_{N-1}H^T(HP_{N-1}H^T + NR)^{-1}(\mathbf{y} - H\hat{\mathbf{x}}_{N-1}) \\
P_N &= (I - P_{N-1}H^T(HP_{N-1}H^T + NR)^{-1}H)P_{N-1}
\end{aligned} \tag{8}$$

Now, we can set

$$\hat{\mathbf{x}}^+ \leftarrow \hat{\mathbf{x}}_N \tag{9}$$

$$P^+ \leftarrow P_N. \tag{10}$$

In this work, we propose applying the same steps in an EKF. In general, the measurement is a nonlinear function of the state (Eq. 1). The measurement Jacobian is $H = \frac{d\mathbf{y}}{d\mathbf{x}}|_{\mathbf{x}=\hat{\mathbf{x}}}$. The proposed approach is summarized in Algorithm 1. Note that Algorithm 1 uses the Joseph form of the covariance update. The Joseph form is equivalent to Eq. 7 but more numerically stable.[22]

Algorithm 1 Proposed recursive update

Require: $\hat{\mathbf{x}}^-$, the prior state estimate; P, the prior covariance; $\mathbf{y}$, a measurement; R, the measurement covariance; N, the number of steps

```
x_0 ← x̂^-
P_0 ← P
for i = 1 ... N do
    H_i = dy/dx |_{x = x̂_{i-1}}
    K_i = P_{i-1} H_i^T (H_i P_{i-1} H_i^T + NR)^{-1}
    x̂_i ← x̂_{i-1} + K_i (y - h(x̂_{i-1}))
    P_i ← (I - K_i H_i) P_{i-1} (I - K_i H_i)^T + K_i (NR) K_i^T
end for
return x̂_N, P_N
```

In this way, the measurement Jacobian H_i is recomputed at each step. This leads to a more accurate estimate for a nonlinear system than a single, large step with high Kalman gain. Unlike in the RUF, there are no weights applied to the Kalman gain. There is also no need to compute the correlation between the state errors and the measurement noise from step to step, since all the steps together represent a single Bayesian update. The reduced gain at each step is simply a by-product of the inflated measurement noise covariance. The BRUF update with a value of $N = 1$ is equivalent to the EKF update.

NUMERICAL RESULTS

We demonstrate the performance of the BRUF in two numerical examples. In the first, we showcase the performance of the BRUF for different values of N in a scalar system with a highly nonlinear measurement. In the second, we implement the algorithm in a navigation filter.

Cubic measurement

Consider a scalar system with nonlinear measurement model

$$y = x^3 + \eta \tag{11}$$

where $\eta \sim \mathcal{N}(0, 0.1^2)$. The prior state estimate is $\hat{x}^- = 2.5$, and the true state value is $x = 3.5$. The covariance associated with the prior state is $P = 0.5^2$. Note that the prior state covariance is much larger than the covariance of the measurement noise.

Figure 1 shows the results of the EKF update and the BRUF update for this scenario. In the EKF update, the high Kalman gain value causes the state estimate to overshoot. A value of $N = 5$ was chosen for the BRUF. The center plot in Figure 1 shows the state estimate after each recursion step. The plot on the right shows a detailed view of steps two through five. The first updated state, $\hat{x}_1$, is close to the value of the EKF update, $\hat{x}^+_{EKF}$, since the Kalman gain K_1 is dominated by the large value of P. Slowly, though, the state estimates sink toward the true state value. The BRUF estimate, $\hat{x}_5$, is much closer to x than $\hat{x}_{EKF}$.

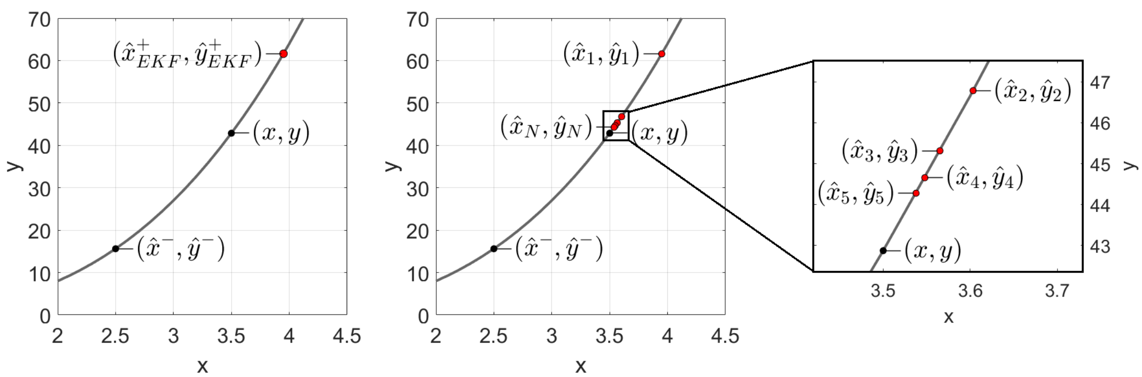

Figure 1: Visualization of results for cubic measurement. The curve $y = x^3$ is plotted in gray. Each state value $\hat{x}$ is plotted with its corresponding estimated measurement, $\hat{y}$. Left: EKF results. Center, Right: BRUF results with detailed view of state estimate for steps 2 through 5.

Table 1 compares the results of the EKF update with the BRUF update for $N = 5$, $N = 10$, and $N = 50$. The true Bayesian posterior values were calculated using rejection sampling. The table shows the mean $\hat{x}^+$ and P^+ values computed from 5000 Monte Carlo runs. Each Monte Carlo run has a different realization of the measurement $y = x^3 + \eta$, since η is a random noise.

It is clear from Table 1 that the proposed method more accurately captures statistics of the true posterior distribution than the EKF. Further, note that increasing the number of update steps N provides diminishing returns. Recall that the EKF update is equivalent to a BRUF update with $N = 1$. The improvement in estimation accuracy between $N = 1$ and $N = 5$ is much more

	True	EKF	BRUF ($N = 5$)	BRUF ($N = 10$)	BRUF ($N = 50$)
$\hat{x}^+$	3.500	3.953	3.540	3.520	3.504
P^+	7.508×10^{-6}	2.844×10^{-5}	7.189×10^{-6}	7.176×10^{-6}	7.305×10^{-6}

Table 1: Numerical results for cubic measurement

significant than the improvement between $N = 10$ and $N = 50$. This shows that relatively few additional computations are necessary to vastly improve on the EKF estimate.

Spacecraft Rendezvous

In this example, we implement the proposed recursive update scheme in a navigation filter. The filter tracks the relative position and velocity of a chaser spacecraft as it moves toward a target in the equatorial plane.[12] The chaser begins 100 m downrange of the target ($+V$-bar) and 5 m below it ($+R$-bar). After an initial burn (see, e.g., Chapter 3 of "Automated Docking and Rendezvous of Spacecraft"),[23] the motion of each spacecraft is governed by Keplerian two-body dynamics:

$$\ddot{\mathbf{r}} = -\frac{\mu_E}{r^3}\mathbf{r} \tag{12}$$

where $\mathbf{r}$ is a position vector expressed in Earth-centered inertial (ECI) coordinates, r is the magnitude of $\mathbf{r}$, and μ_E is Earth's gravitational parameter. Fig. 2 shows the trajectory in the local-vertical local-horizontal (LVLH) frame. The origin of the LVLH frame is located on the target spacecraft. The target spacecraft moves in a circular orbit around Earth at an altitude of 400 km. The rendezvous maneuver is completed in 1000 seconds.

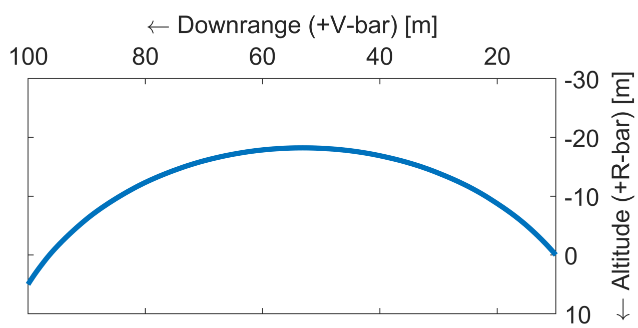

Figure 2: Rendezvous trajectory in the LVLH frame. The chaser moves from left to right, ending the maneuver 10 m from the origin.

The filter estimates position and velocity in the LVLH frame. The filter dynamics are governed by the Clohessy-Wiltshire equations:

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} \tag{13}$$

where

$$\mathbf{A} = \begin{bmatrix} \mathbf{0}_{3\times3} & \mathbf{I}_{3\times3} \\ \mathbf{A}_1 & \mathbf{A}_2 \end{bmatrix} \tag{14}$$

and

$$\mathbf{A}_1 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & -n^2 & 0 \\ 0 & 0 & 3n^2 \end{bmatrix} \quad \mathbf{A}_2 = \begin{bmatrix} 0 & 0 & 2n \\ 0 & 0 & 0 \\ -2n & 0 & 0 \end{bmatrix}. \tag{15}$$

The Clohessy-Wiltshire equations are a linear approximation of the dynamics of the chaser with respect to a target in circular orbit. Thus, the filter uses a different dynamics model than the true dynamics. The orbital rate of the target is $n = 0.001131$ rad/s.

The filter begins with an initial position uncertainty of 10 m and an initial velocity uncertainty of 0.05 m/s.

$$\mathbf{P}_0 = \begin{bmatrix} 10^2\mathbf{I}_{3\times3} & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & 0.05^2\mathbf{I}_{3\times3} \end{bmatrix} \tag{16}$$

Every two seconds, the chaser measures its position relative to the target. The distance to the target is $\rho = \sqrt{\mathbf{x}(1)^2 + \mathbf{x}(2)^2 + \mathbf{x}(3)^2}$. The azimuth is $\alpha = \tan^{-1}[\mathbf{x}(2)/\mathbf{x}(1)]$, and the elevation is $\epsilon = \sin^{-1}[\mathbf{x}(3)/\rho]$. The measurement is

$$\mathbf{y} = \begin{bmatrix} \rho \\ \alpha \\ \epsilon \end{bmatrix} + \boldsymbol{\eta} \tag{17}$$

where

$$\boldsymbol{\eta} \sim \mathcal{N}\left(\mathbf{0}_{3\times1}, \begin{bmatrix} 0.1^2 & \mathbf{0}_{1\times2} \\ \mathbf{0}_{2\times1} & (0.1\pi/180)^2\mathbf{I}_{2\times2} \end{bmatrix}\right). \tag{18}$$

Figures 3 and 4 show the performance of the EKF and the BRUF respectively. Each figure shows the estimation error recorded during 100 Monte Carlo runs. A value of $N = 10$ was chosen for the BRUF. The EKF performs poorly in the first few hundred seconds. The estimation errors are high, and the covariance bounds fail to capture them. This means the filter is overconfident. The proposed approach mitigates these problems and remains consistent from beginning to end.

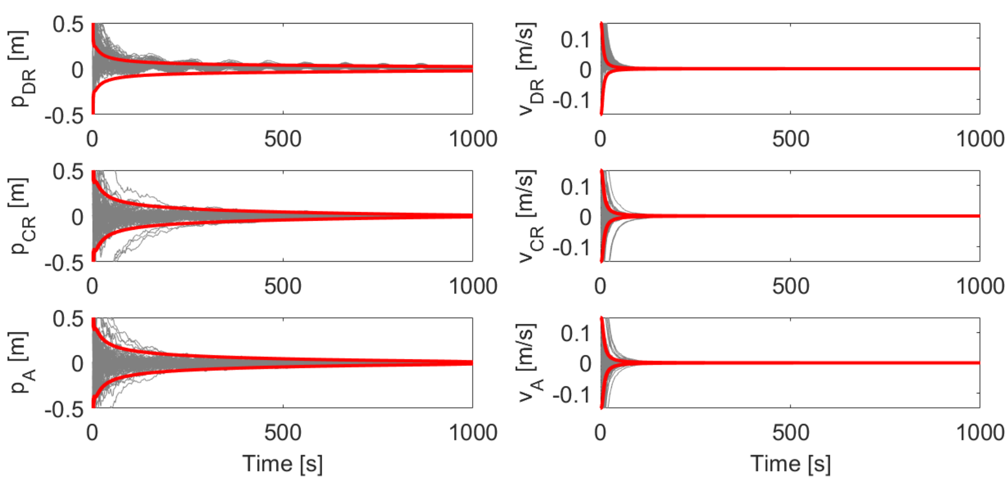

Figure 3: EKF estimation error and 3σ standard deviation. DR = Downrange, CR = Crossrange, A = Altitude.

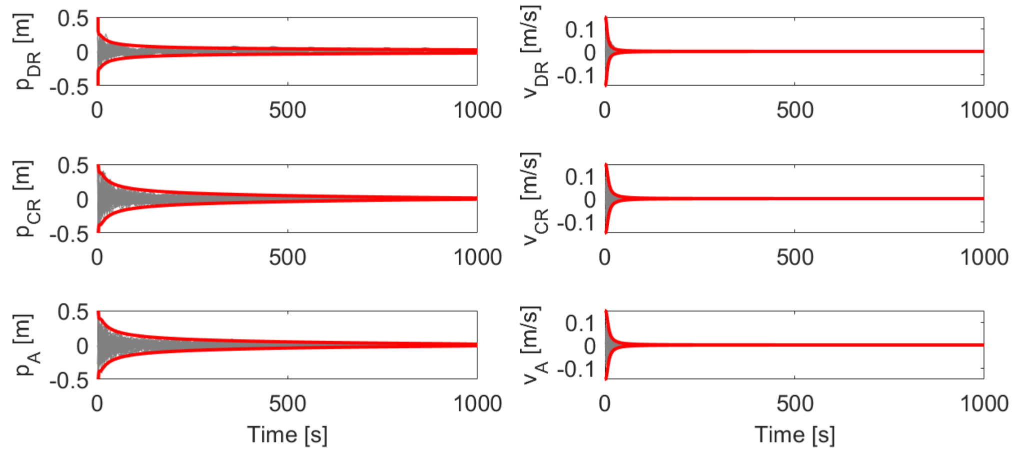

Figure 4: BRUF estimation error and 3σ standard deviation for $N = 10$. DR = Downrange, CR = Crossrange, A = Altitude.

Figure 3 shows a significant bias in the downrange position estimate in the EKF. The BRUF is able to effectively remove this bias. Figure 5 shows a comparison between the bias in the downrange position estimate in the EKF and the BRUF for $N = 2$, $N = 5$, $N = 10$, $N = 50$, and $N = 100$ during the first 200 seconds of the simulation. Each of the curves in Figure 5 represents the mean of 500 Monte Carlo runs. The bias in the EKF is highest, followed by $N = 2$. The bias in the filters with $N = 50$ and $N = 100$ is virtually indistinguishable.

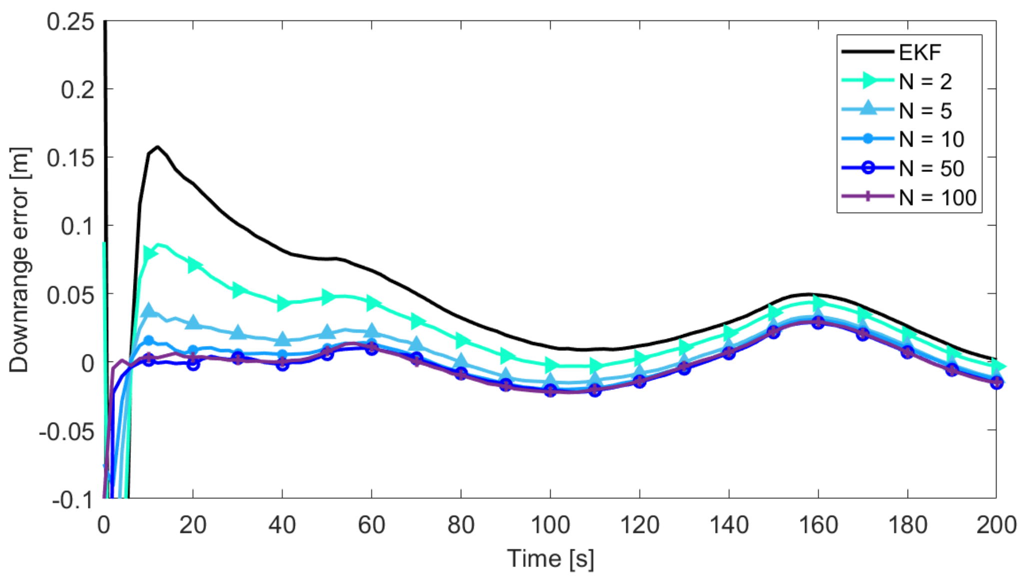

Figure 5: Downrange position error bias Monte Carlo study

CONCLUSION

This work introduces the Bayesian recursive update filter (BRUF), a novel recursive update filter based on Bayesian principles. It is an alternative to other EKF extensions such as the iterative extended Kalman filter (IEKF) and the recursive update filter (RUF). Instead of computing a series of N small EKF updates like the RUF, the BRUF divides a single EKF update into N small steps. This subtle difference gives rise to a more elegant algorithm that preserves the main benefit of iterative methods: the recalculation of the measurement Jacobian at each step. The BRUF is easy to implement, and its computational cost is an integer multiple of the computational cost of the EKF. Any EKF can be converted to a BRUF. To our knowledge, there are no drawbacks aside from the increase in computational complexity.

Designers requiring a highly accurate measurement update may be willing to devote computation time to more recursions, while those desiring a lower computational cost may delight in the improvement in estimation accuracy from just a few. We have demonstrated in two examples the principle of diminishing returns; $N = 2$ or $N = 3$ provides a vast improvement over the EKF, while $N = 50$ may not provide a noticeable improvement over $N = 10$. Ultimately, the number of recursions required in an application depends on many factors including the severity of the nonlinearity in the measurement. Empirical analysis may be employed to select the appropriate value.

REFERENCES

[1] R. E. Kalman, "A new approach to linear filtering and prediction problems," 1960.

[2] R. E. Kalman and R. S. Bucy, "New results in linear filtering and prediction theory," 1961.

[3] Y. Bar-Shalom, X. R. Li, and T. Kirubarajan, *Estimation with applications to tracking and navigation: theory algorithms and software*. John Wiley & Sons, 2004.

[4] A. Doucet, N. De Freitas, N. J. Gordon, *et al.*, *Sequential Monte Carlo methods in practice*, Vol. 1. Springer, 2001.

[5] M. Raitoharju and R. Piché, "On computational complexity reduction methods for Kalman filter extensions," *IEEE Aerospace and Electronic Systems Magazine*, Vol. 34, No. 10, 2019, pp. 2–19.

[6] A. Gelb *et al.*, *Applied optimal estimation*. MIT press, 1974.

[7] B. M. Bell and F. W. Cathey, "The iterated Kalman filter update as a Gauss-Newton method," *IEEE Transactions on Automatic Control*, Vol. 38, No. 2, 1993, pp. 294–297.

[8] R. Zanetti, "Recursive update filtering for nonlinear estimation," *IEEE Transactions on Automatic Control*, Vol. 57, No. 6, 2011, pp. 1481–1490.

[9] R. G. Brown and P. Y. Hwang, "Introduction to random signals and applied Kalman filtering: with MATLAB exercises and solutions," *Introduction to random signals and applied Kalman filtering: with MATLAB exercises and solutions*, 1997.

[10] Y. Huang, Y. Zhang, Z. Wu, and N. Li, "An improved nonlinear Kalman filter with recursive measurement update," *2016 35th Chinese Control Conference (CCC)*, IEEE, 2016, pp. 4792–4797.

[11] Y. Li, Y. Cheng, X. Li, H. Wang, X. Hua, and Y. Qin, "Bayesian nonlinear filtering via information geometric optimization," *Entropy*, Vol. 19, No. 12, 2017, p. 655.

[12] R. Zanetti, "Adaptable recursive update filter," *Journal of Guidance, Control, and Dynamics*, Vol. 38, No. 7, 2015, pp. 1295–1300.

[13] Q. Zhang and T. L. Song, "Improved bearings-only target tracking with iterated Gaussian mixture measurements," *IET Radar, Sonar & Navigation*, Vol. 11, No. 2, 2017, pp. 294–303.

[14] H. Nurminen, R. Piché, and S. Godsill, "Gaussian flow sigma point filter for nonlinear Gaussian state-space models," *2017 20th International Conference on Information Fusion (Fusion)*, IEEE, 2017, pp. 1–8.

[15] U. D. Hanebeck, K. Briechle, and A. Rauh, "Progressive Bayes: a new framework for nonlinear state estimation," *Multisensor, Multisource Information Fusion: Architectures, Algorithms, and Applications 2003*, Vol. 5099, SPIE, 2003, pp. 256–267.

[16] U. D. Hanebeck and O. Feiermann, "Progressive Bayesian estimation for nonlinear discrete-time systems: the filter step for scalar measurements and multidimensional states," *42nd IEEE International Conference on Decision and Control (IEEE Cat. No. 03CH37475)*, Vol. 5, IEEE, 2003, pp. 5366–5371.

[17] U. D. Hanebeck, "PGF 42: Progressive Gaussian filtering with a twist," *Proceedings of the 16th International Conference on Information Fusion*, IEEE, 2013, pp. 1103–1110.

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

[19] Z. Wu, Z. Shi, and L. Pan, "A Kalman Filter with Progressive Measurement Update," *2018 37th Chinese Control Conference (CCC)*, IEEE, 2018, pp. 4228–4235.

[20] F. Daum, J. Huang, and A. Noushin, "Exact particle flow for nonlinear filters," *Signal processing, sensor fusion, and target recognition XIX*, Vol. 7697, SPIE, 2010, pp. 92–110.

[21] T. Ding and M. J. Coates, "Implementation of the Daum-Huang exact-flow particle filter," *2012 IEEE Statistical Signal Processing Workshop (SSP)*, IEEE, 2012, pp. 257–260.

[22] R. S. Bucy and P. D. Joseph, *Filtering for stochastic processes with applications to guidance*, Vol. 326. American Mathematical Soc., 2005.

[23] W. Fehse, "Automated docking and rendezvous of spacecraft," *Cambridge University*, 2003.