Performance of the Unscented Kalman Filter

Tim Magoun*

Abstract. Filtering based state estimation techniques have been popular due to their simplicity and computational performance. This paper explores the key ideas behind the Unscented Kalman Filter (UKF) and how it performs as an extension of the Kalman Filter (KF) as a nonlinear estimation algorithm. We implement a variant of the UKF for attitude estimation, and compare its performance to competing algorithms. Finally, we discuss recommendations for a attitude estimation filter.

Key words. Unscented Kalman Filter, State Estimation, Kalman Filter, Cholesky Decomposition

1. Introduction.

State estimation is the derivation of the internal state of a system, such as an aircraft or a chemical reaction, by using sensor readings and known inputs from all previous time steps up to the present. The problem is challenging and is an ongoing field of study, with applications ranging from real-time control of airplanes to the balancing of power grids [21]. There are several general approaches to state estimation. A notable approach, which is currently gaining popularity in the field, is the use of neural networks to perform inference [18]. However, the relatively lower interpret-ability of neural networks leave something to be desired. This paper will focus solely on a probabilistic approach that defines the optimal estimation as the state with the maximum-a-posteri (MAP) likelihood, also known as the MAP estimate.

1.1. The Problem.

For the rest of the paper, we will be solving the following formulation of a state estimation problem: There exists a system with dynamics

$$X_k = f(X_{k-1}, u_{k-1}, t) + v_k$$

where f is a time varying nonlinear function that describes the ideal dynamics of the system, X is the state, u is the known control input to the system, and v_k is a zero mean Gaussian that we call the *process noise*. We also receive a sensor update with the form

$$y_k = h(X_k, t) + w_k$$

where h is also a time varying nonlinear function that describes the ideal sensor model, and w_k is a zero mean Gaussian that we call the *sensing noise*. We also know the covariance matrices Q_k, R_k for v and w, respectively.

The goal of our filter is to find a estimate of the state $\tilde{X}_k$ such that the expectation of the error $E[e_k^T e_k]$ is minimized. We define error as $e_k = X_k - \tilde{X}_k$. Note that this definition of the error is quadratic.

1.2. The Kalman Filter.

One of the most well known filters in the estimation field is the Kalman filter, named after Rudolf E. Kalman for his creation in 1960 [14]. The filter was created to provide a solution for an optimal linear filter without the need to derive it from

*Massachusetts Institute of Technology (magoun@mit.edu).

the impulse response of that filter. In his paper, Kalman proved that the optimal unbiased estimator for a linear system is in the form of a feedback system, with the gain determined by the equation

$$K_{opt} = P_k^- C_k^T (C_k P_k^- C_k^T + R_k)^{-1}$$

where Q_k^- is the covariance matrix before the update, C_k is the observation matrix which transforms the state to the observation, and R_k is the sensor noise previously defined.

The Kalman filter can be broken down into two different steps: propagation and update. The propagation step is used to create an estimate based on the dynamics of the system, and the update step then corrects the previous estimate with additional information from the sensor readings. We will use notation Z_k^- to describe the variable Z before update, and Z_k^+ to describe it after the update, both at time k.

Propagation equations:

$$X_k^- = AX_{k-1}^+ \tag{1.1}$$

$$P_k^- = AP_{k-1}^+ A^T + Q_k \tag{1.2}$$

Update equations:

$$X_k^+ = X_k^- + K_{opt}(Y_k - C_k X_k^-) \tag{1.3}$$

$$P_k^+ = (I - K_{opt} C_k) P_k^- \tag{1.4}$$

Intuitively, the estimate is propagated while its covariance P_k grows with Q_k, which means that our estimate becomes less accurate over time as the process noise continuously corrupts our certainty. However, we are able to increase our certainty with the update step, where $Y_k - C_k X_k^-$, also known as the *innovation* is used to correct our propagated estimate. The feedback nature is done through K_{opt} which is the optimal feedback such that the covariance of the updated estimate is minimized. In other words, the gain balances the information from our known system dynamics with the sensor readings. An accurate sensor reading will be incorporated more in the update step compared to an inaccurate sensor reading.

1.3. Nonlinear Extensions of the Kalman Filter. One key weakness of the Kalman filter comes from its assumption of a linear system, as seem in 1.1. Unfortunately, very few systems in the real world exhibit this characteristic, and even fewer that is interesting to us. As a result, there are several extensions to the Kalman filter that makes it applicable to nonlinear systems.

The Extended Kalman Filter (EKF) was created in 1968 by Sorenson et. al [24]. The key insight in that paper was to use a linearized version of the system dynamics and sensor model at every time step in order to approximate the behavior of the system.

$$A_k = f(X_{k-1}^+, u_{k-1}, t) + \frac{\partial f}{\partial X} f|_{X_{k-1}^+} (X_k^- - X_{k-1}^+) \tag{1.5}$$

$$C_k = h(X_{k-1}^+, u_{k-1}, t) + \frac{\partial h}{\partial X} f|_{X_{k-1}^+} (X_k^- - X_{k-1}^+) \tag{1.6}$$

This filter is very popular due to its simplicity and acceptable performance, especially when the local nonlinear models are well described by the linearized versions. Many extensions to the EKF have been created in the following decades. [8] [2] [30] [19].

However, some have critiqued the EKF's usefulness when it comes to very nonlinear problems [11]. In particular, it was claimed that the covariance estimate created by the linearized model does not approximate the true covariance, and that the EKF may lead to stability issues in certain occasions, for example, if the filter was given an incorrect initialization. To combat these issues, the Unscented Kalman Filter [11] was proposed in 2000. The namesake came from the use of the Unscented transform [11] , which operates on the idea that one can approximate the posterior distribution of a variable via sampling and feeding through the true nonlinearity.

A brief overview of the algorithm is as follows:

$$\chi_k = \text{Sampling Algorithm}(x_{k-1}^{+}) \tag{1.7}$$

$$\chi_k^{-} = f(\chi_k, u_k, t) \tag{1.8}$$

$$y_k = h(\chi_k^{-}) \tag{1.9}$$

where χ_k is a set of points sampled by the Unscented transform. The Kalman gain and covariances are then calculated with a weighted average of χ_k and used for the propagation and update steps. We will explain both of these operations in detail in 2.2.

The authors argue that this method, which only focuses on the approximation of the distribution, performs better than the EKF, which attempts to approximate the entire nonlinear function via linearization. It is important to note that this claim has been challenged in [9]. We will discuss how the UKF performs this approximation in detail in 2.1.

1.4. Prevalence of the UKF Today. The UKF became a very popular filter after its introduction, with many variants much like those of the EKF. It has been used in tracking of over-the-horizon objects using radars [3], state of power systems [31], and brain machine interfaces for the estimation of neuron activation [16]. In the field of robotics, the UKF has been very popular in vehicle state estimation [29] [22] [25] often using inertial measurement units (IMU) along with other forms of odometry. It is also important to note that the orientation of an object is represented by the special orthogonal group $SE(3)$, so special consideration must be taken to ensure the validity of operations on the state within the filter. Much work has been done on the use of the UKF on manifolds, such as [15] [5] [23] and [28].

The UKF is also used in problems adjacent to vehicle state estimation, such as in Simultaneous Localization and Mapping (SLAM) [10], where it jointly estimates the state of many objects in the environment.

The Unscented transform has also been applied to other applications as well, notably the Unscented particle filter [26] which increases the effectiveness of Markov Chain Monte Carlo methods.

1.5. Alternatives to Filter Based Estimation. One downside of the UKF, and other filter based methods, is its marginalization of information over time. This is beneficial for computation reasons, since it does not have an unbounded growth of observations, but it also prevents the use of previous information to create a more optimal estimate. Other methods that overcome this limitation exists, and some examples are given here:

If one augments the state of the system to include states over a fixed period of time before the current time, we arrive at a fixed-lag smoother. This formulation allows us to increase the available amount of non-marginalized information, and improves estimation accuracy [4]. This is still a filter based method but different enough from the formulation described in this paper.

If one drastically increases the number of samples of the UKF, we arrive at the particle filter [1], originally proposed in 1993. This method increases the accuracy of the approximation of the nonlinearity at the cost of computation.

If one preserves all previous information and perform Bayesian inference at every time step with clever reuse of previously computed results, we arrive at incremental smoothing and mapping (iSAM) [13] and its extensions [12] [6]. These approaches give more optimal solutions compared to filtering based methods for the SLAM problem, at the cost of computation.

1.6. Outline. The paper is organized as follows. First, a detailed description of the Unscented Kalman Filter (UKF) and its relation to the Kalman filter will be presented in 2. A numerical stability issue of the UKF is explored in 3, with the discussion of a more numerically stable variant of the UKF, the Square Root UKF (SR-UKF). Originally the scope of the paper includes an implementation and comparison of the SR-UKF, but unfortunately it had to be left out to reduce scope. An implementation of the UKF is then presented 4, in context of attitude estimation with a 6 degrees of freedom IMU, with a note about modifications to the algorithm needed for this particular application. We then present an experiment in 5 with simulated IMU data to verify the performance of our algorithm, and compare it to a popular alternative: the complementary filter [7]. Finally, we will conclude with a summary of the properties of the filter in 7 as well as recommendations for a user who wishes to implement an attitude estimation filter.

2. Description of KF and UKF.

This section will describe the intuition and some mathematical facts about the KF, and explain how the UKF modifies the KF by propagating and updating samples of the previous estimate, known as *sigma points*. Tim Barfoot's book [1] offers an excellent derivation, from three different perspectives, if the reader is interested in further exploration.

2.1. Brief Derivation of the Kalman Filter. The propagation step of the KF is described in 1.1 1.1. The key insight is that since the process noise v_k is modeled as a zero-mean Gaussian, $E[Av] = AE[v] = 0$. Note that this is not true in the nonlinear case! For example, a dynamics model $f(x) = x^2$ will violate our conclusion above.

Therefore, using our insight, the mean of the current estimate can be propagated directly using the dynamics matrix A, yielding 1.1. Similarly, propagation of the covariance matrix P_k can be derived, yielding a matrix equation for the update 1.2 . This propagation step can be done multiple times between updates, in cases where the sensor for the observation is polled at a lower frequency than is required by the system.

For the update step, the derivation of the optimal gain can be approached from a optimization perspective [1]:

Given we have the update step in the form of 1.4 and 1.3, find the optimal gain K_{opt} such that the cost

$$J = \frac{1}{2}E[e_k^T e_k] = \frac{1}{2}\mathrm{tr}E[e_k e_k^T]$$

is minimized. By setting the derivative to zero, we can solve for K_{opt}:

$$\begin{aligned}\frac{\partial J}{\partial K_k} &= -(I - K_k C_k)P_k^- C_k^T + K_k R_k \\ &= 0 \\ K_{opt} &= P_k^- C_k^T (C_k P_k^- C_k^T + R_k)^{-1}\end{aligned}$$

A key characteristic of the Kalman filter is that it is the best linear unbiased estimator (BLUE), and it is also called the linear quadratic estimator (LQE) due to the quadratic cost function that it minimizes.

If the dynamics of the system is stable (All eigenvalues of A is less than 1), the Kalman gain, as well as the post-update covariance converges to a steady state value. We can directly use this steady state gain to perform our filtering and gain a computational benefit. This is called the *Steady State Kalman Filter*.

2.2. Description of Unscented Kalman Filter.

2.2.1. The Unscented Transform and Sigma Points.

As mentioned previously, the UKF is an extension of the KF for nonlinear systems. The key idea behind the UKF is the use of the Unscented transform, which estimates the posterior distribution of a random variable x after the application of a nonlinear function [11]. We assume that a random variable with mean $x \in \mathbb{R}^n$ is Gaussian, and create samples with a positive and negative perturbation along n axis. There are several approaches to the selection of the axis as well as the magnitude of the perturbations, and in this paper, we will use the method that is the most common and highlighted in several highly cited sources [1] [27] [20].

Given the random variable with mean $x \in \mathbb{R}^n$ with a covariance matrix P, we would like to sample $2n+1$ sigma points $\chi(0 \ldots 2n)$ to propagate through the nonlinear function f. We calculate the perturbation σ and apply it to x through the following set of equations:

$$P' = (n+\lambda)P \tag{2.1}$$

$$L = \sqrt{P'}, \text{ s.t. } LL^T = P' \tag{2.2}$$

$$\sigma(i) = i^{th} \text{ column from } L \tag{2.3}$$

$$\chi(0) = x \tag{2.4}$$

$$\chi(i) = x + \sigma(i) \qquad i = 1, \ldots, n \tag{2.5}$$

$$\chi(i) = x - \sigma(i-n) \qquad i = n+1, \ldots, 2n \tag{2.6}$$

For any set of sigma points χ, we can then describe two functions to transform it back into its mean x and covariance P:

$$x = \frac{1}{n+\lambda}\left\{\lambda\chi(0) + \frac{1}{2}\sum_{i=1}^{2n}\chi(i)\right\} \tag{2.7}$$

$$P = \frac{1}{n+\lambda}\left\{\lambda[\chi(0)-x][\chi(0)-x]^T + \frac{1}{2}\sum_{i=1}^{2n}[\chi(i)-x][\chi(i)-x]^T\right\} \tag{2.8}$$

From now on, we will use $\chi = \text{sample}(x, P)$ to describe 2.1 to 2.6, $x = \text{mean}[\chi]$ to describe 2.7, and $P = \text{cov}[\chi]$ to describe 2.8. Using a combination of those functions, we can sample the current state estimate, pass it through a nonlinearity, and then reassemble it into a random variable that better approximates the true distribution than using linearization. This is the core idea of the Unscented transform.

2.2.2. Integration of Sigma Points into the Kalman Filter. This section highlights how sigma points are used in the UKF algorithm. Recall in in our problem statement that there are two nonlinear functions present: $f(x, u, t)$ and $h(x, t)$. The EKF linearized both functions, and the KF assumed that both are linear functions of the state and time. For the UKF, we will use $\text{sample}(x, P)$, $\text{mean}[\chi]$, and $\text{cov}[\chi]$ for both functions.

The propagation equations in 1.1 becomes the following

$$\chi_k = \text{sample}(x_{k-1}^+) \tag{2.9}$$

$$\chi_k^-(i) = f(\chi_k(i)) \quad i = 0, \dots, 2n+1 \tag{2.10}$$

$$x_k^- = \text{mean}[\chi_k^-] \tag{2.11}$$

$$P_k^- = \text{cov}[\chi_k^-] \tag{2.12}$$

The update equations 1.3 becomes

$$\chi_k^- = \text{sample}(x_k^-) \tag{2.13}$$

$$\gamma_k^-(i) = h(\chi_k^-(i)) \quad i = 0, \dots, 2n+1 \tag{2.14}$$

$$y_k^- = \text{mean}[\gamma_k^-] \tag{2.15}$$

$$P_k^{yy} = \text{cov}[\gamma_k^-] \tag{2.16}$$

We calculate one additional covariance P_k^{xy} which is used in the calculation of the Kalman gain:

$$P_k^{xy} = \frac{1}{n+\lambda} \left\{ \lambda[\chi_k^-(0) - x_k^-][\gamma_k^-(0) - y_k^-]^T + \frac{1}{2} \sum_{i=1}^{2n} [\chi_k^-(i) - x_k^-][\chi_k^-(i) - y_k^-]^T \right\}$$

The Kalman gain is then calculated as

$$K_k = P_k^{xy}(P_k^{yy} + R_k)^{-1}$$

All other steps of the UKF are the same as the KF. One does not have to explicitly calculate x_k^- and P_k^- because the sigma points after the update, χ_k^-, provides enough information for the update step.

3. Numerical Instability of UKF. One key step to calculating the sigma points in the UKF is finding the perturbation σ that describe the distribution of the state being sampled. The key idea presented in the Unscented Transform is that one can well approximate the distribution with only a constant number, $2n+1$ if $x \in \mathbb{R}^n$, of samples. The axis along which the samples are perturbed is determined by the square root of the covariance matrix, where the matrix square root operation is performed via *Cholesky decomposition* [27].

$$A = LL^T$$

It is required that the matrix A is positive semi-definite, which means that all eigenvalues of A must be positive.

However, during the application of the UKF, we might have a situation where the conditions on A no longer holds. Recall that we are taking the square root of P, the covariance of the state, which means that the matrix should be symmetric and positive definite due to it being expressed as an outer product of the deviations from mean. The eigenvalues of a covariance matrix represents the magnitude of the uncertainty along a particular axis. Using these ideas, we can show a system where the covariance of a part of the state grows unbounded while another part shrinks to a very small steady state value.

Consider a vehicle traveling at measured velocity v with a measurement noise of $\sigma_v = 10$. As we integrate the velocity measurement to propagate the vehicle state, the covariance grows unbounded. Now let's measure the y position with a very accurate instrument, $\sigma_y = 0.001$. A UKF will find an optimal gain that sets $P_{yy} \approx \sigma_y^2$ while keeping all other uncertainties unbounded. Therefore, the condition number of P will also grow in an unbounded fashion, and will lead to issues with catastrophic cancellation. Once an eigenvalue change in sign, P is no longer PSD, the Cholesky factorization will fail.

One solution to this problem is to propagate the square root of P directly, as shown in [27]. Unfortunately we will not explore that algorithm in this paper.

4. Implementation of UKF. We implemented a UKF for attitude estimation using a 6-DOF IMU, in `Python3`. Attitude estimation is when one finds the roll, pitch, and yaw of a vehicle in 3D space. One major modification to the UKF algorithm described 2.1 is that we use a "error state" representation of the attitude instead of a true state representation. This is necessary because a rotation in 3D space cannot be represented by vectors in $\mathbb{R}^3$ undergoing all vector operations, instead it belongs to a group called $SO(3)$ which dictates how we perform operations. One popular representation of rotation is a unit quaternion, which is a vector in $\mathbb{R}^4$ that is constrained to have a unit norm. The addition of two rotations is not the addition of the quaternion representations, because doing so will violate the norm constraint, instead we must perform quaternion multiplication. Likewise, the "average" rotation out of a group of rotations cannot be the average of the quaternions. Instead, we will use an unrestricted form to represent the error during every iteration, and reset by updating the true state with the error state. This assumes that the attitude doesn't change by a full rotation during a single iteration, which is reasonable for the sensor update rates and the expected angular velocities.

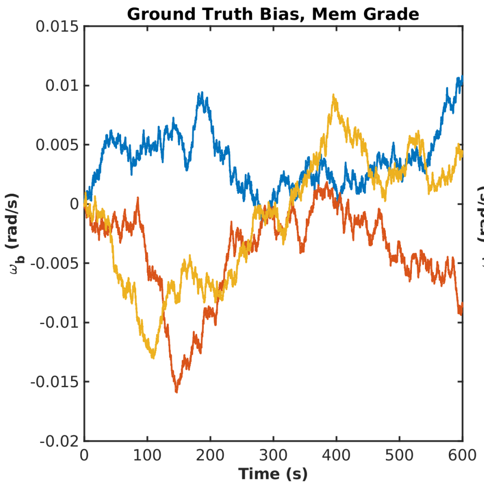
This was a implementation described in [5], originally intended for spacecraft attitude estimation. One small modification was made, we used an accelerometer to perform the update step while the original implementation used a magnetometer. This implementation uses `NumPy` and `SciPy` for basic matrix operations and Cholesky decomposition, all other functions are implemented by me.

5. Experiment. To simplify the experimental procedure, we used synthetic data that is generated by a model of a 6-DOF IMU. We assume that there are uncorrelated white noise on the gyroscope and accelerometer outputs, and that there is a white noise driven random walk on the bias of the gyroscope. This model is often used for IMUs and is applicable to most applications in robotics.

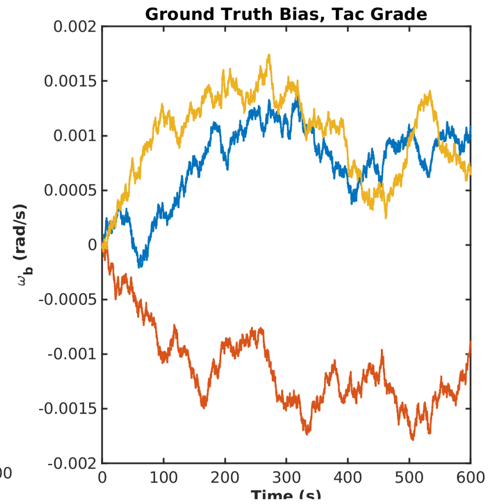
The IMU is simulated for 600 seconds, with a ground true angular acceleration generated by a random polynomial. The scale factor of the acceleration is adjusted to limit the max to about 0.5 rad/s to avoid a cluttered graph. We sample at 20 Hz and have two sets of IMU parameters that describe the covariance of the three white noises in the simulation, one is labeled *mem*, for the MEMs grade IMUs which are low cost and commonly used in smartphones, another is labeled *tac*, for TAC (tacticle) grade IMUs, which are used for scientific and military applications. The parameters are shown in 1.

Table 1
Simulated IMU Parameters

	Bias Drift (rad/$s^{\frac{3}{2}}$)	Gyro Noise (rad/$s^{\frac{1}{2}}$)	Accel Noise (m/$s^{\frac{5}{2}}$)
MEMs grade	2e-4	4.65e-5	5e-5
TAC grade	2e-3	2.03e-4	5e-4



(a) MEMs Grade



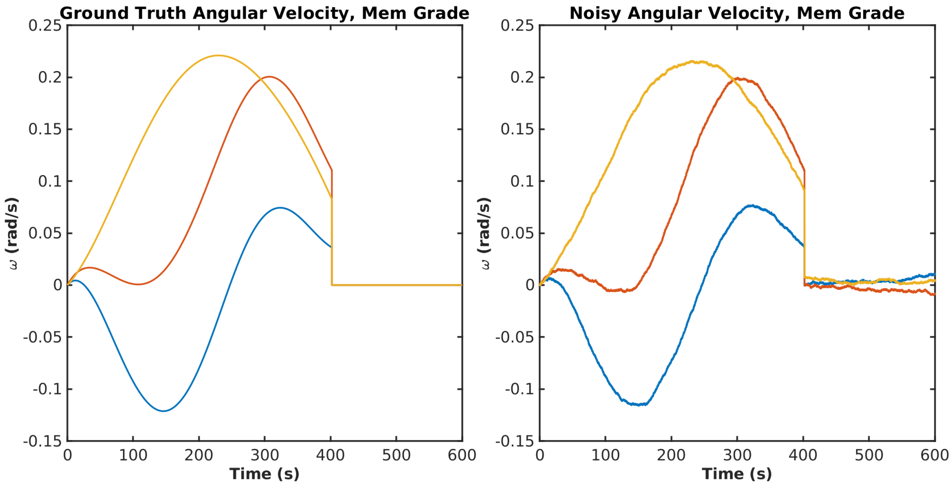
(b) TAC Grade

Figure 1. *Simulated Gyro Bias Drift.*

Figures 1, 2a, and 2b describe the generated bias, ground truth angular velocity and noisy and biased angular velocity. Notice that the bias acts like a random walk process which is consistent with the behavior of real IMUs. For both sets of generated data, we used the same ground truth angular velocity, to ensure the differences in measurements are comparable both absolutely and relative to the magnitude of the actual movement. Figure 3 shows the attitude of the vehicle. Notice that there a lot of transitions due to angle wraparounds, which is very difficult for additive formulations of the Kalman filters.

We ran the UKF filter, as well as the following filters from `python-ahrs` to compare the performance of our filter: angular integration, Madgwick filter[17], EKF, and the Complementary filter. For each filter we tested it with both the correct initial attitude of no rotation and with an incorrect rotation (45°roll, 45 °pitch, and 90 °yaw). A selection of filter outputs are plotted Figures 4a to 4d, along with the ground truth attitude.

Note that there the yaw drift for all of the filters in Figure 4 are much greater than other



(a) GT Angular Velocity (b) Noisy Angular Velocity

Figure 2. *Simulated Angular Velocity.*

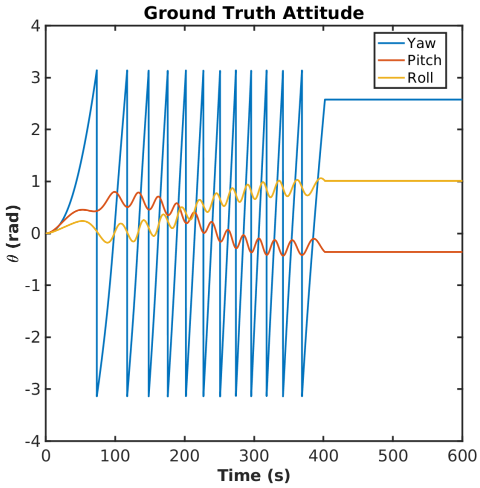


Figure 3. *Ground Truth Attitude*

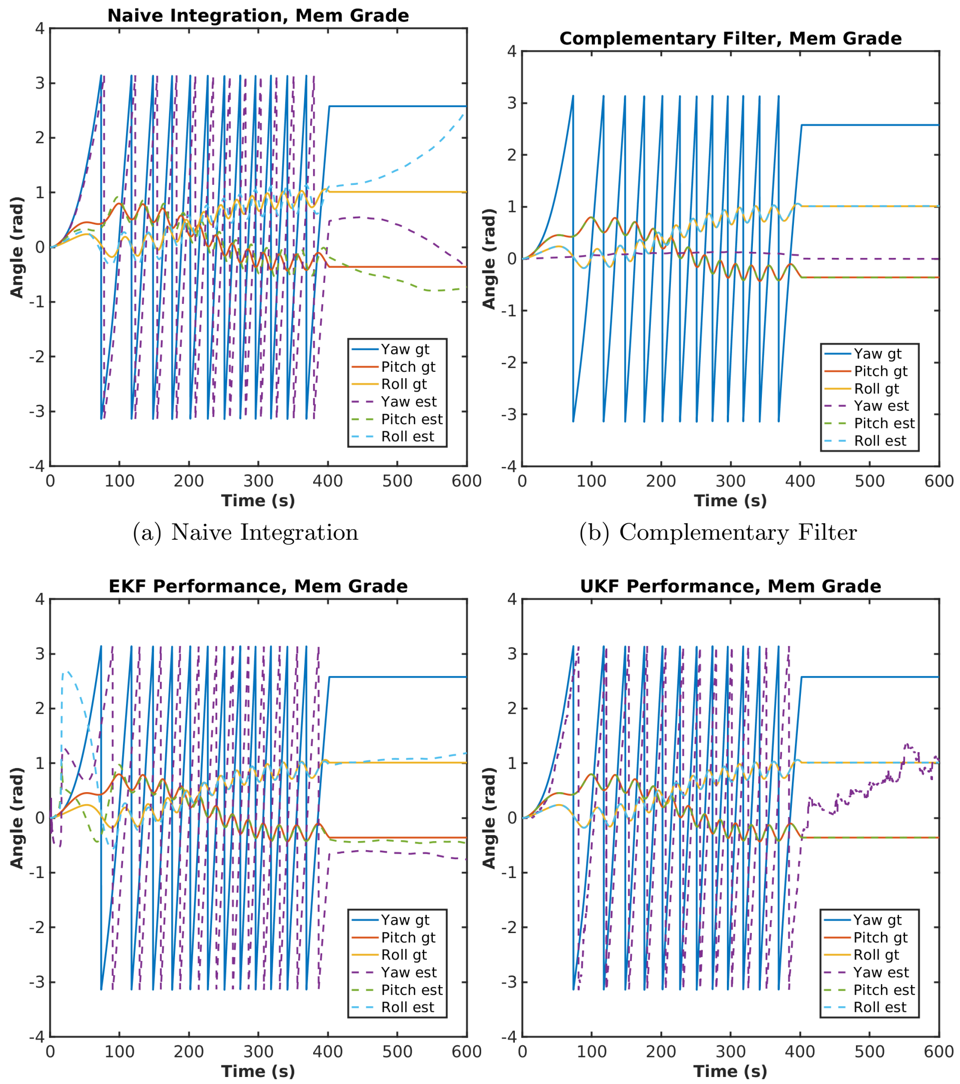
axis, due to the lack of correction by the accelerometer.

Table 2 summarizes the errors of the filters over all trials, with the error function defined as

$$\tilde{\theta} = \frac{1}{N}\sum_{i=0}^{N}\sqrt{(x_i - x_i^*)^2}$$

where x_i is the filter output at time-step i and x_i^* is the ground truth attitude. Only the roll and pitch is evaluated since yaw is not correctable with the accelerometer measurement.

For the trials with incorrect initialization, it is very apparent that the EKF and Madgwick filter takes more time to converge to the correct solution while the Complementary filter



(a) Naive Integration (b) Complementary Filter

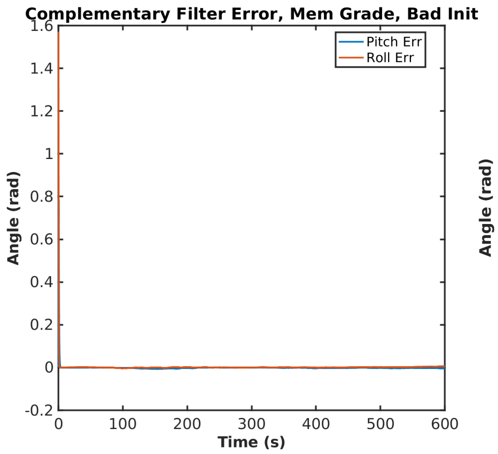
(c) EKF (d) UKF

Figure 4. *Filter Outputs for a MEM grade IMU.*

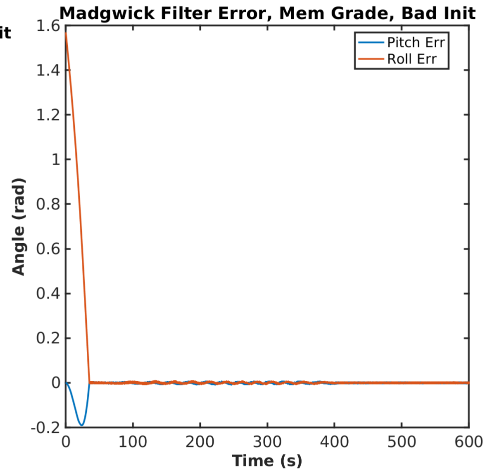
Table 2
Mean Roll and Pitch Error (rad)

	MEM Grade	MEM Grade Bad Init	TAC Grade	TAC Grade Bad Init
Naive Integration	0.512	1.502	0.387	1.474
Complementary Filter	0.047	**0.050**	**0.039**	**0.041**
Madgwick Filter	0.051	0.101	0.100	0.148
EKF	0.355	0.308	0.099	0.206
UKF	**0.029**	0.065	0.201	0.111

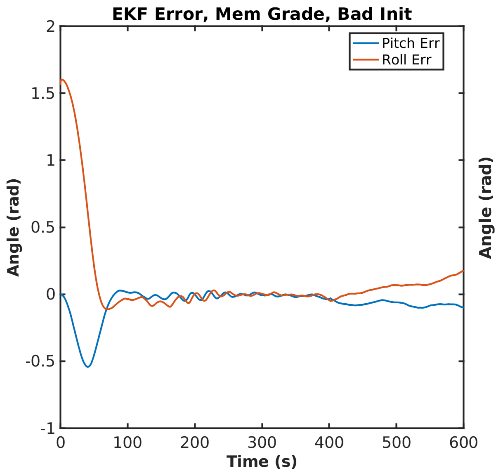
and the UKF converged more rapidly. Plots of the errors for the incorrect initialization case is shown in Figure 5. The difference in convergence style is also interesting, with the UKF appearing more unstable while the Madgwick and Complementary filter seemed like a controlled convergence. The convergence speed of the Madgwick filter can be tuned with a change in parameters, allowing it to trust the accelerometer more.



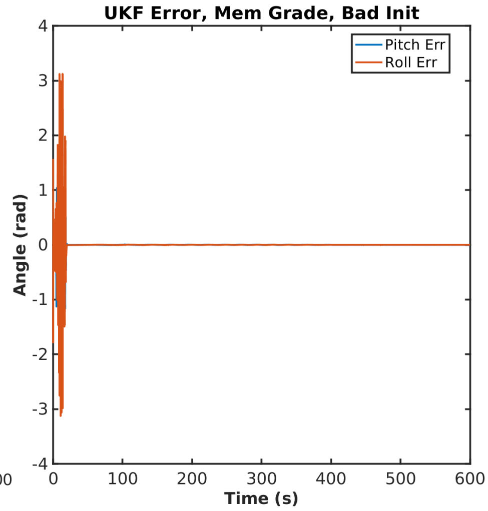
(a) Complementary Filter



(b) Madgwick Filter



(c) EKF



(d) UKF

Figure 5. *Filter Errors with Incorrect Initialization.*

6. Discussion of Filter Performance and Stability. From the data in Table 2, we can see that the UKF performs very well compared to other filters that are designed for attitude estimation, especially considering the fact that our implementation is not well tuned and may contain small bugs. It is interesting to see that a better sensor did not give a better performance for the UKF, which may suggest that either the uncertainty created by errors in the sampling process or small bugs have decreased the accuracy of this filter. More time is needed to verify those claims. In general, an incorrect initialization lead to a worse solution from all filters. The most drastic difference is from the EKF, which may be attributed to very incorrect linearizations in the filter. We believe that the UKF did not perform as badly because it uses the Unscented transform to directly sample the distribution, which does not depend on the quality of an initial guess.

Another interesting thing to note is that the Madgwick filter and the Complementary filter converged much quicker to the correct position than the UKF and EKF. Perhaps the simplicity in their formulation makes them especially well suited for bad initial guesses. Finally, the naive integration performed very poorly when there is a bad initialization, as expected.

I did not get to work on the stability analysis of the UKF. But it is worth noting that the covariance matrix fails to remain PSD when the covariances of the sensors reach below 1e-6 with a sample rate of 20 Hz. It may be an issue with my implementation, but as is, it will not be able to property filter readings from a navigation grade IMU due to numerical issues.

7. Conclusions. Our experiments have shown that the UKF is a powerful filter that gives superior results to the EKF when it comes to nonlinear state estimation problems. For the problem of attitude estimation in particular, it remains competitive with the complementary filter. If one is deciding between the two for an attitude estimator, I would recommend the complementary filter for its simplicity, numerical stability, and good performance. However, its filter design is not as flexible as the UKF, and therefore will be less useful when one needs to integrate other forms of measurements.

REFERENCES

[1] T. D. Barfoot, *State Estimation for Robotics*, Cambridge University Press, 1 ed., July 2017, https://doi.org/10.1017/9781316671528, https://www.cambridge.org/core/product/identifier/9781316671528/type/book (accessed 2023-05-10).

[2] A. Barrau and S. Bonnabel, *The Invariant Extended Kalman Filter as a Stable Observer*, IEEE Transactions on Automatic Control, 62 (2017), pp. 1797–1812, https://doi.org/10.1109/TAC.2016.2594085. Conference Name: IEEE Transactions on Automatic Control.

[3] P. Braca, R. Grasso, M. Vespe, S. Maresca, and J. Horstmann, *Application of the JPDA-UKF to HFSW radars for maritime situational awareness*, in 2012 15th International Conference on Information Fusion, July 2012, pp. 2585–2592.

[4] S. E. Cohn, N. S. Sivakumaran, and R. Todling, *A Fixed-Lag Kalman Smoother for Retrospective Data Assimilation*, Monthly Weather Review, 122 (1994), pp. 2838–2867, https://doi.org/10.1175/1520-0493(1994)122⟨2838:AFLKSF⟩2.0.CO;2, https://journals.ametsoc.org/view/journals/mwre/122/12/1520-0493_1994_122_2838_aflksf_2_0_co_2.xml (accessed 2023-05-10). Publisher: American Meteorological Society Section: Monthly Weather Review.

[5] J. L. Crassidis and F. L. Markley, *Unscented Filtering for Spacecraft Attitude Estimation*.

[6] F. Dellaert, *Factor Graphs and GTSAM: A Hands-on Introduction*.

[7] M. Euston, P. Coote, R. Mahony, J. Kim, and T. Hamel, *A complementary filter for attitude*

estimation of a fixed-wing UAV, in 2008 IEEE/RSJ International Conference on Intelligent Robots and Systems, Sept. 2008, pp. 340–345, https://doi.org/10.1109/IROS.2008.4650766. ISSN: 2153-0866.

[8] A. Germani, C. Manes, and P. Palumbo, *Polynomial extended Kalman filter*, IEEE Transactions on Automatic Control, 50 (2005), pp. 2059–2064, https://doi.org/10.1109/TAC.2005.860256. Conference Name: IEEE Transactions on Automatic Control.

[9] F. Gustafsson and G. Hendeby, *Some Relations Between Extended and Unscented Kalman Filters*, IEEE Transactions on Signal Processing, 60 (2012), pp. 545–555, https://doi.org/10.1109/TSP.2011.2172431, http://ieeexplore.ieee.org/document/6051522/ (accessed 2023-05-03).

[10] G. P. Huang, A. I. Mourikis, and S. I. Roumeliotis, *On the complexity and consistency of UKF-based SLAM*, in 2009 IEEE International Conference on Robotics and Automation, May 2009, pp. 4401–4408, https://doi.org/10.1109/ROBOT.2009.5152793. ISSN: 1050-4729.

[11] S. J. Julier and J. K. Uhlmann, *New extension of the Kalman filter to nonlinear systems*, Orlando, FL, USA, July 1997, p. 182, https://doi.org/10.1117/12.280797, http://proceedings.spiedigitallibrary.org/proceeding.aspx?doi=10.1117/12.280797 (accessed 2023-04-07).

[12] M. Kaess, H. Johannsson, R. Roberts, V. Ila, J. J. Leonard, and F. Dellaert, *iSAM2: Incremental smoothing and mapping using the Bayes tree*, The International Journal of Robotics Research, 31 (2012), pp. 216–235, https://doi.org/10.1177/0278364911430419, http://journals.sagepub.com/doi/10.1177/0278364911430419 (accessed 2023-05-10).

[13] M. Kaess, A. Ranganathan, and F. Dellaert, *iSAM: Incremental Smoothing and Mapping*, IEEE Transactions on Robotics, 24 (2008), pp. 1365–1378, https://doi.org/10.1109/TRO.2008.2006706, http://ieeexplore.ieee.org/document/4682731/ (accessed 2023-05-10).

[14] R. E. Kalman, *A New Approach to Linear Filtering and Prediction Problems*, Journal of Basic Engineering, 82 (1960), pp. 35–45, https://doi.org/10.1115/1.3662552, https://asmedigitalcollection.asme.org/fluidsengineering/article/82/1/35/397706/A-New-Approach-to-Linear-Filtering-and-Prediction (accessed 2023-04-07).

[15] E. Kraft, *A quaternion-based unscented Kalman filter for orientation tracking*, in Sixth International Conference of Information Fusion, 2003. Proceedings of the, Cairns, Queensland, Australia, 2003, IEEE, pp. 47–54, https://doi.org/10.1109/ICIF.2003.177425, http://ieeexplore.ieee.org/document/1257247/ (accessed 2023-05-02).

[16] Z. Li, J. E. O'Doherty, T. L. Hanson, M. A. Lebedev, C. S. Henriquez, and M. A. L. Nicolelis, *Unscented Kalman Filter for Brain-Machine Interfaces*, PLoS ONE, 4 (2009), p. e6243, https://doi.org/10.1371/journal.pone.0006243, https://dx.plos.org/10.1371/journal.pone.0006243 (accessed 2023-05-10).

[17] S. O. H. Madgwick, *An efficient orientation filter for inertial and inertial/magnetic sensor arrays*.

[18] E. Manitsas, R. Singh, B. C. Pal, and G. Strbac, *Distribution System State Estimation Using an Artificial Neural Network Approach for Pseudo Measurement Modeling*, IEEE Transactions on Power Systems, 27 (2012), pp. 1888–1896, https://doi.org/10.1109/TPWRS.2012.2187804, http://ieeexplore.ieee.org/document/6176289/ (accessed 2023-05-10).

[19] F. L. Markley, *Multiplicative vs. Additive Filtering for Spacecraft Attitude Determination*.

[20] H. M. T. Menegaz, J. Y. Ishihara, G. A. Borges, and A. N. Vargas, *A Systematization of the Unscented Kalman Filter Theory*, IEEE Transactions on Automatic Control, 60 (2015), pp. 2583–2598, https://doi.org/10.1109/TAC.2015.2404511, http://ieeexplore.ieee.org/document/7042740/ (accessed 2023-05-03).

[21] A. Sagan, Y. Liu, and A. Bernstein, *Decentralized Low-Rank State Estimation for Power Distribution Systems*, IEEE Transactions on Smart Grid, 12 (2021), pp. 3097–3106, https://doi.org/10.1109/TSG.2021.3058609. Conference Name: IEEE Transactions on Smart Grid.

[22] E.-H. Shin and N. El-Sheimy, *An unscented Kalman filter for in-motion alignment of low-cost IMUs*, in PLANS 2004. Position Location and Navigation Symposium (IEEE Cat. No.04CH37556), Apr. 2004, pp. 273–279, https://doi.org/10.1109/PLANS.2004.1309005.

[23] A. M. Sjøberg and O. Egeland, *Lie Algebraic Unscented Kalman Filter for Pose Estimation*, Apr. 2020, http://arxiv.org/abs/2005.00385 (accessed 2023-05-01). arXiv:2005.00385 [cs, eess, math].

[24] H. W. SORENSON‡ and A. R. STUBBERUD§, *Non-linear filtering by approximation of the a posteriori density*, International Journal of Control, 8 (1968), pp. 33–51, https://doi.org/10.1080/00207176808905650, https://doi.org/10.1080/00207176808905650 (accessed 2023-05-03). Publisher:

Taylor & Francis _eprint: https://doi.org/10.1080/00207176808905650.

[25] M. St-Pierre and D. Gingras, *Comparison between the unscented Kalman filter and the extended Kalman filter for the position estimation module of an integrated navigation information system*, in IEEE Intelligent Vehicles Symposium, 2004, June 2004, pp. 831–835, https://doi.org/10.1109/IVS.2004.1336492.

[26] R. van der Merwe, A. Doucet, N. de Freitas, and E. Wan, *The Unscented Particle Filter*, in Advances in Neural Information Processing Systems, vol. 13, MIT Press, 2000, https://proceedings.neurips.cc/paper/2000/hash/f5c3dd7514bf620a1b85450d2ae374b1-Abstract.html (accessed 2023-05-03).

[27] R. Van der Merwe and E. Wan, *The square-root unscented Kalman filter for state and parameter-estimation*, in 2001 IEEE International Conference on Acoustics, Speech, and Signal Processing. Proceedings (Cat. No.01CH37221), vol. 6, Salt Lake City, UT, USA, 2001, IEEE, pp. 3461–3464, https://doi.org/10.1109/ICASSP.2001.940586, http://ieeexplore.ieee.org/document/940586/ (accessed 2023-04-07).

[28] M. Vandyke, J. Schwartz, and C. Hall, *Unscented Kalman Filtering for spacecraft attitude state and parameter estimation*, Advances in the Astronautical Sciences, 119 (2004).

[29] H. Wang, G. Fu, J. Li, Z. Yan, and X. Bian, *An Adaptive UKF Based SLAM Method for Unmanned Underwater Vehicle*, Mathematical Problems in Engineering, 2013 (2013), pp. 1–12, https://doi.org/10.1155/2013/605981, http://www.hindawi.com/journals/mpe/2013/605981/ (accessed 2023-05-03).

[30] J. N. Yang, S. Lin, H. Huang, and L. Zhou, *An adaptive extended Kalman filter for structural damage identification*, Structural Control and Health Monitoring, 13 (2006), pp. 849–867, https://doi.org/10.1002/stc.84, https://onlinelibrary.wiley.com/doi/abs/10.1002/stc.84 (accessed 2023-05-10). _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/stc.84.

[31] J. Zhao and L. Mili, *Robust Unscented Kalman Filter for Power System Dynamic State Estimation With Unknown Noise Statistics*, IEEE Transactions on Smart Grid, 10 (2019), pp. 1215–1224, https://doi.org/10.1109/TSG.2017.2761452. Conference Name: IEEE Transactions on Smart Grid.