Center for Robotics and Embedded Systems University of Southern California Technical Report CRES-08-004



A Note on Unscented Filtering and State Propagation in Nonlinear Systems

Jonathan Kelly

November 20, 2008 Revised November 23, 2011

Abstract

In this note, we illustrate the effect of nonlinear state propagation in the unscented Kalman filter (UKF). We consider a simple nonlinear system, consisting of a two-axis inertial measurement unit. Our intent is to show that the propagation of a set of sigma points through a nonlinear process model in the UKF can produce a counterintuitive (but correct) updated state estimate. We compare the results from the UKF with those from the well-known extended Kalman filter (EKF), to highlight how the UKF and the EKF differ.

1 Introduction

The goal in *estimating* the state of a dynamic system is to determine, as nearly as possible, the true state of the system from a series of noisy measurements (observations). State estimation for linear dynamic systems has a long history and is well understood – in practice, however, all real-world systems are *nonlinear* to some degree. Unfortunately, nonlinear dynamic systems are significantly more difficult to deal with, and the associated theory is less cohesive and developed.

The most popular and established nonlinear estimation algorithm is the extended Kalman filter (EKF), which typically uses an error state formulation, linearizing about the current nonlinear state estimate. This effectively turns a nonlinear problem into a linear problem, to which the standard Kalman filter equations may be applied. However, linearization in the EKF can lead to difficulties such as inconsistency, and even filter divergence, for highly nonlinear systems.

Recently, an alternative Kalman filter formulation known as the *unscented Kalman filter* (UKF) has attracted significant attention from the estimation and control communities. The UKF operates by propagating a set of *sigma points*, rather than simply the state mean, through the nonlinear process and measurement models. This approach has a number of

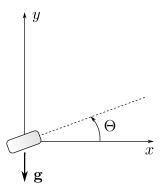


Figure 1: Simple two-dimensional IMU system. The orientation of the IMU, Θ , is measured counterclockwise relative to the positive x axis. Gravity acts directly along the negative y axis.

advantages, including, crucially, the ability to more accurately transform state mean and covariance information over time.

Our purpose in this note is to demonstrate the performance of the UKF, as compared to the EKF, for a simple nonlinear system. Specifically, we show (via simulation) that the propagation of the state vector through a nonlinear process model in the UKF can lead to a counterintuitive updated state estimate. We compare the UKF output with the output from the EKF, for the same set of inputs and measurements, to highlight the differences between the filters. Finally, we offer some conclusions and suggestions for further reading at the end of the paper.

2 A Simple Nonlinear System

We will study a simple nonlinear system, consisting of a 'two-dimensional' inertial measurement unit (IMU). A diagram of the system is shown in Figure 1. The IMU has two orthogonal accelerometers, which measure acceleration in the plane of the page, and a single gyroscope, which measures the rate of rotation about an axis oriented directly out of the page. The state vector is:

$$\mathbf{X}(t) = \begin{bmatrix} p_x(t) & p_y(t) & \Theta(t) & v_x(t) & v_y(t) \end{bmatrix}^T \tag{1}$$

where all of the quantities are functions of time, as indicated. States p_x and p_y are the x and y position, respectively, of the IMU in the global reference frame (defined by the coordinate axes in Figure 1). Likewise, states v_x and v_y are the x and y velocity of the IMU in the global frame. The state Θ is the orientation angle of the IMU body frame, measured counterclockwise relative to the positive x axis.

2.1 Process Model

The system we have described evolves in continuous time according to a series of nonlinear differential equations. We can write the state update equations in vector form as:

$$\dot{\mathbf{X}}(t) = f(\mathbf{X}(t), \mathbf{u}(t), \mathbf{n}(t)) = \begin{bmatrix} \dot{p}_x(t) \\ \dot{p}_y(t) \\ \dot{\Theta}(t) \\ \dot{v}_x(t) \\ \dot{v}_y(t) \end{bmatrix} + \mathbf{n}(t) = \begin{bmatrix} v_x(t) \\ v_y(t) \\ \omega_m(t) \\ \mathbf{C}(\Theta(t)) \begin{bmatrix} a_{mx}(t) \\ a_{my}(t) \end{bmatrix} + \mathbf{g} \end{bmatrix} + \mathbf{n}(t) \quad (2)$$

Here, $\mathbf{u}(t) = \begin{bmatrix} \omega_m(t) & a_{mx}(t) & a_{my}(t) \end{bmatrix}^T$ is a vector of measured IMU values, which we treat as control inputs; $\omega_m(t)$ is the rotation rate of the IMU, while $a_{mx}(t)$ and $a_{my}(t)$ are linear accelerations measured in the IMU frame. The matrix $\mathbf{C}(\Theta(t))$ is the instantaneous direction cosine (rotation) matrix from the IMU frame to the global frame at time t. Vector \mathbf{g} is the gravity vector. Finally, $\mathbf{n}(t)$ is a 5×1 white Gaussian process noise vector with covariance $\mathbf{E}\langle \mathbf{n}(t), \mathbf{n}(\tau) \rangle = \mathbf{Q}_c(t) \, \delta(t-\tau)$ (where $\mathbf{Q}_c(t)$ is a spectral density matrix).

For implementation, it is necessary to discretize the continuous-time system equations above. We will use discrete-time versions of both the UKF and the EKF (instead of continuous-discrete implementations — see [1] for a discussion). In the case of the EKF, we treat both the time and measurements updates discretely. In the case of the UKF, we propagate the state forward in time using a fourth-order Runge-Kutta integrator; for practical purposes, this will not affect the conclusions we draw. We also require a discretized process noise covariance matrix, \mathbf{Q}_k , for the discrete time step k. We will assume that \mathbf{Q}_k is constant (i.e. $\forall k, \mathbf{Q}_k = \mathbf{Q}$), and is given by:

$$\mathbf{Q} = \begin{bmatrix} (0.05 \text{ m})^2 & 0 & 0 & 0 & 0\\ 0 & (0.05 \text{ m})^2 & 0 & 0 & 0\\ 0 & 0 & (\pi/180 \text{ rad})^2 & 0 & 0\\ 0 & 0 & 0 & (0.01 \text{ m/s})^2 & 0\\ 0 & 0 & 0 & 0 & (0.01 \text{ m/s})^2 \end{bmatrix}$$

2.2 Measurement Model

Our observations will consist of discrete measurements of the position and orientation of the IMU, from e.g. a LIDAR sensor. At time step k, the measurement is:

$$\mathbf{z}_{k} = h(\mathbf{X}_{k}, \eta_{k}) = \begin{bmatrix} p_{x,k} \\ p_{y,k} \\ \Theta_{k} \end{bmatrix} + \eta_{k}$$
(3)

 $^{^{1}}$ We assume that the gravity vector has a nominal magnitude of 9.81 m/s 2 and acts directly along the global negative y axis.

where η_k is a 3×1 white Gaussian measurement noise process vector with covariance matrix \mathbf{R}_k . We will assume that \mathbf{R}_k is constant (i.e. $\forall k, \mathbf{R}_k = \mathbf{R}$), and is given by:

$$\mathbf{R} = \begin{bmatrix} (0.1 \text{ m})^2 & 0 & 0\\ 0 & (0.1 \text{ m})^2 & 0\\ 0 & 0 & (\pi/72 \text{ rad})^2 \end{bmatrix}$$

Although we have defined the measurement noise covariance matrix above, we will always set the actual measurement noise entering the system to zero, i.e. the term η_k will be identically zero for all k. This is a fictitious situation, of course, but will make it easier to emphasize the differences between the EKF and the UKF later in the paper.

2.3 Initialization

We assume that the IMU is positioned 1.0 meters along the positive y axis, and that the unit remains level and stationary for all time. The true system state (which does not change) is then:

$$\mathbf{X}(t) = \begin{bmatrix} 0.0 \text{ m} & 1.0 \text{ m} & 0.0 \text{ rad} & 0.0 \text{ m/s} & 0.0 \text{ m/s} \end{bmatrix}^T, \quad t \ge 0$$

We will choose the following initial state estimate² for both the EKF and the UKF:

$$\hat{\mathbf{X}}_0 = \begin{bmatrix} 0.0 \text{ m} & 0.0 \text{ m} & 0.0 \text{ rad} & 0.0 \text{ m/s} & 0.0 \text{ m/s} \end{bmatrix}^T$$

The initial state covariance matrix is:

$$\mathbf{P}_0 = \begin{bmatrix} (0.5 \,\mathrm{m})^2 & 0 & 0 & 0 & 0 \\ 0 & (0.5 \,\mathrm{m})^2 & 0 & 0 & 0 \\ 0 & 0 & (\pi/18 \,\mathrm{rad})^2 & 0 & 0 \\ 0 & 0 & 0 & (0.01 \,\mathrm{m/s})^2 & 0 \\ 0 & 0 & 0 & 0 & (0.01 \,\mathrm{m/s})^2 \end{bmatrix}$$

In our implementation, IMU data arrives at a rate of 10 Hz, and position/orientation measurements arrive at 1 Hz. Our simulation runs both filters for 10 seconds, resulting in a total of 100 time updates and 10 measurement updates.

3 Extended Kalman Filtering

The most popular nonlinear state estimator is the extended Kalman filter. In the EKF, state updates are performed by linearizing the nonlinear system about the current state mean [2]. We will use the discrete form of the EKF, in which both time and measurement updates are performed at discrete time steps:

$$\hat{\mathbf{X}}_{k}^{-} = f(\hat{\mathbf{X}}_{k-1}^{+}, \mathbf{u}_{k-1}, \mathbf{n}_{k-1})$$
(4)

$$\mathbf{P}_{k}^{-} = \mathbf{F}_{k-1} \mathbf{P}_{k-1}^{+} \mathbf{F}_{k-1}^{T} + \mathbf{Q}_{k-1}$$
 (5)

²Throughout this note, we denote estimated quantities with the '^' (hat) symbol.

where $\hat{\mathbf{X}}_{k-1}^+$ is the state estimate at the previous time step and \mathbf{P}_{k-1}^+ is the associated covariance matrix, \mathbf{u}_{k-1} is the vector of control inputs, \mathbf{n}_{k-1} is the measurement noise vector, and \mathbf{F}_{k-1} is the Jacobian matrix of partial derivatives of the process model function $f(\mathbf{X}_{k-1}, \mathbf{u}_{k-1}, \mathbf{n}_{k-1})$ with respect to the system state, evaluated at $\hat{\mathbf{X}}_{k-1}^+$.

When a position/orientation measurement arrives, we compute the Kalman gain matrix at time step k as:

$$\mathbf{K}_{k} = \mathbf{P}_{k}^{-} \mathbf{H}_{k}^{T} \left[\mathbf{H}_{k} \mathbf{P}_{k}^{-} \mathbf{H}_{k}^{T} + \mathbf{R}_{k} \right]^{-1} \tag{6}$$

where \mathbf{H}_k is the Jacobian matrix of partial derivatives of the the measurement model function $h(\mathbf{X}_k, \eta_k)$ with respect to the system state, evaluated at $\hat{\mathbf{X}}_k^-$. Using the Kalman gain matrix, we update the state estimate and state covariance matrix at time step k according to:

$$\hat{\mathbf{X}}_k^+ = \hat{\mathbf{X}}_k^- + \mathbf{K}_k \left[\mathbf{z}_k - h(\hat{\mathbf{X}}_k^-, \eta_k) \right] \tag{7}$$

$$\mathbf{P}_{k}^{+} = \left[\mathbf{I} - \mathbf{K}_{k} \mathbf{H}_{k} \right] \mathbf{P}_{k}^{-} \tag{8}$$

3.1 Results and Remarks

The final EKF state estimate (after 100 time steps) is:

$$\hat{\mathbf{X}}_{100} = \begin{bmatrix} 0.000 \text{ m} & 1.000 \text{ m} & 0.000 \text{ rad} & 0.000 \text{ m/s} & 0.001 \text{ m/s} \end{bmatrix}^T$$

Note that the true and estimated state values are identical to three decimal places, with the exception of a very small residual velocity error.

We begin our analysis by examining a plot of the estimated position \hat{p}_y of the IMU along the y axis over time, shown in Figure 2. Note that, because we have perfect measurement information, the position estimate converges rapidly to the true position of the IMU.

The estimated orientation $\hat{\Theta}_k$ of the IMU frame relative to the global frame remains identically zero for the entire simulation.³ We plot the standard deviation of the orientation estimate in Figure 3. The initial standard deviation is 10 degrees, which drops to approximately 2.5 degrees after the first measurement update. The 'sawtooth' pattern that appears starting at t = 1.0 seconds is produced by the process noise that is added at each time step.

Figure 4 shows the estimated y axis velocity of the IMU over time. Importantly, although the IMU is actually stationary throughout the simulation, \hat{v}_y is not always zero. This is because the initial measurement update induces a correlation between the state \hat{p}_y and the state \hat{v}_y (i.e. position depends on velocity). Note also, however, that the value of \hat{v}_y remains constant between measurement updates. The EKF propagates the state mean only — since the estimated orientation $\hat{\Theta}$ is always zero, and the accelerometers provide perfect measurements (as per our assertion), the net acceleration is always zero, and hence the velocity remains constant. We will return to this point in Section 4.1 when we discuss the effect of the unscented transform. For comparison, we also plot the \hat{v}_y data for the EKF in Figure 5 using the same scale increments as for the UKF (in Figure 8).

³The measured rotation rate is always zero, and the orientation of the IMU is not correlated with any of the other states in the state vector.

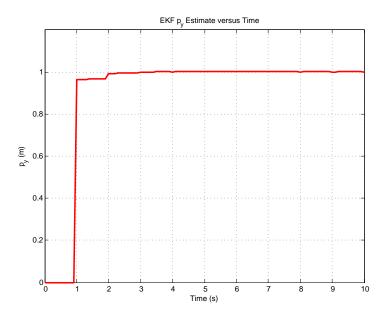


Figure 2: IMU y axis position estimate (\hat{p}_y) , for the EKF. Note the rapid convergence to the true position of the IMU.

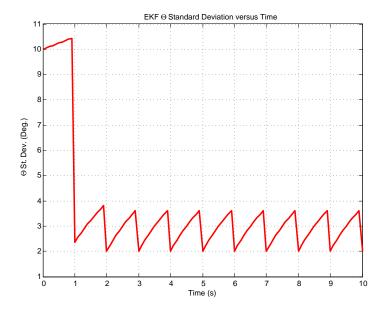


Figure 3: Standard deviation of IMU orientation estimate $(\hat{\Theta})$, for the EKF. The sawtooth pattern results from the addition of process noise.

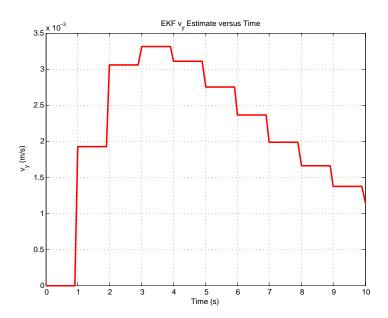


Figure 4: IMU y axis velocity estimate (\hat{v}_y) , for the EKF. The staircase form of the plot results from an initial weak correlation between states \hat{v}_y and \hat{p}_y , induced by the measurement updates.

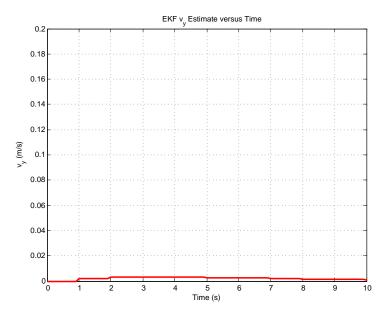


Figure 5: Scaled IMU y axis velocity estimate (\hat{v}_y) , for the EKF. This plot has the same scale increments as the \hat{v}_y plot for the UKF, shown in Figure 8.

4 Unscented Kalman Filtering

The unscented Kalman filter is a modification of the standard Kalman filter that is particularly suited for nonlinear problems. We give a brief overview of the filter here and refer the reader to [3] for a more detailed treatment.

The UKF captures the mean and covariance of a probability distribution with a set of deterministically-selected sample points called $sigma\ points$, which lie on the covariance contours of the N-dimensional state space. To update the system state, the UKF generates a set of 2N+1 sigma points — each point is then propagated through the (nonlinear) system process and measurement models to compute the posterior state mean and state covariance. This approach (the $unscented\ transform$) is attractive because it avoids many of the problems that can result from linearization in the EKF, and has third-order accuracy for Gaussian error distributions. An additional benefit is that the UKF is derivative-free — it does not require the computation of Jacobian matrices.

The most straightforward implementation of the UKF augments the state vector and state covariance matrix with process and measurement noise components:

$$\mathbf{X}_{k-1}^{a} = \begin{bmatrix} \mathbf{X}_{k-1} \\ \mathbf{n}_{k-1} \\ \eta_{k-1} \end{bmatrix} = \begin{bmatrix} \mathbf{X}_{k-1} \\ \mathbf{0}_{5\times 1} \\ \mathbf{0}_{3\times 1} \end{bmatrix}, \quad \mathbf{P}_{k-1}^{a} = \begin{bmatrix} \mathbf{P}_{k-1} & \mathbf{0}_{5\times 5} & \mathbf{0}_{5\times 3} \\ \mathbf{0}_{5\times 5} & \mathbf{Q} & \mathbf{0}_{5\times 3} \\ \mathbf{0}_{3\times 5} & \mathbf{0}_{3\times 5} & \mathbf{R} \end{bmatrix}$$
(9)

where \mathbf{n}_{k-1} and η_{k-1} are the zero mean process noise and measurement noise vectors and \mathbf{Q} and \mathbf{R} are the associated covariance matrices; the augmented state vector has size N=13 in this case. We employ the scaled form of the unscented transform [4] which requires a scaling term:

$$\kappa = \alpha^2 (N + \beta) - N \tag{10}$$

Here, α is a parameter which controls the spread of the sigma points around the state mean; α is usually set to a small positive value (0.5 in our case). The β value is a distribution-specific parameter; $\beta=2$ is optimal for Gaussian state variables. At time k-1 and for state estimate $\hat{\mathbf{X}}_{k-1}^+$, the augmented state $\hat{\mathbf{X}}_{k-1}^a$ is used to generate the set of sigma points according to:

$$\chi_{0,k-1} = \hat{\mathbf{X}}_{k-1}^a \tag{11}$$

$$\chi_{j,k-1} = \hat{\mathbf{X}}_{k-1}^a + \sqrt{(\kappa + N)\mathbf{P}_{k-1}^a}, \quad j = 1,\dots, N$$
 (12)

$$\chi_{j,k-1} = \hat{\mathbf{X}}_{k-1}^a - \sqrt{(\kappa + N)\mathbf{P}_{k-1}^a}, \quad j = N+1, \dots, 2N$$
 (13)

The term $\sqrt{(\kappa+N)\mathbf{P}_{k-1}^a}$ includes the matrix square root of \mathbf{P}_{k-1}^a , which can be found by Cholesky decomposition. Individual sigma points are propagated through the process model and combined in a weighted average to generate an updated state mean and state covariance matrix. The weight values are:

$$W_0^{(m)} = \kappa/(\kappa + N) \tag{14}$$

$$W_0^{(c)} = \kappa/(\kappa + N) + (1 - \alpha^2 + \beta)$$
(15)

$$W_0^{(m)} = \kappa/(\kappa + N) + (1 - \alpha^2 + \beta)$$

$$W_j^{(m)} = W_j^{(c)} = 1/2(\kappa + N), \quad j = 1, \dots, 2N$$
(16)

These weights are used to calculate the *a priori* state estimate and covariance matrix at step k:

$$\chi_{j,k} = f(\chi_{j,k-1}, \mathbf{u}_{k-1}) \tag{17}$$

$$\hat{\mathbf{X}}_{k}^{-} = \sum_{j=0}^{2N} W_{j}^{(m)} \chi_{j,k} \tag{18}$$

$$\mathbf{P}_{k}^{-} = \sum_{j=0}^{2N} W_{j}^{(c)} [\chi_{j,k} - \hat{\mathbf{X}}_{k}^{-}] [\chi_{j,k} - \hat{\mathbf{X}}_{k}^{-}]^{T}$$
(19)

where f is the (nonlinear) process model function.

When an observation arrives, we determine the predicted measurement value by propagating each sigma point through the (nonlinear) measurement model function h:

$$\mathcal{Z}_{j,k} = h(\chi_{j,k}) \tag{20}$$

$$\hat{\mathbf{z}}_{k}^{-} = \sum_{j=0}^{2N} W_{j}^{(m)} \mathcal{Z}_{j,k}$$
 (21)

We then perform a state update by computing the Kalman gain \mathbf{K} and the *a posteriori* state vector and state covariance matrix:

$$\mathbf{P}_{\hat{\mathbf{x}}_{k}\hat{\mathbf{z}}_{k}} = \sum_{i=0}^{2N} W_{i}^{(c)} [\chi_{i,k} - \hat{\mathbf{X}}_{k}^{-}] [\mathcal{Z}_{i,k} - \hat{\mathbf{z}}_{k}^{-}]^{T}$$
(22)

$$\mathbf{P}_{\hat{\mathbf{z}}_{k}\hat{\mathbf{z}}_{k}} = \sum_{i=0}^{2N} W_{i}^{(c)} [\mathcal{Z}_{i,k} - \hat{\mathbf{z}}_{k}^{-}] [\mathcal{Z}_{i,k} - \hat{\mathbf{z}}_{k}^{-}]^{T}$$
(23)

$$\mathbf{K}_k = \mathbf{P}_{\hat{\mathbf{x}}_k \hat{\mathbf{z}}_k} \mathbf{P}_{\hat{\mathbf{z}}_k \hat{\mathbf{z}}_k}^{-1} \tag{24}$$

$$\hat{\mathbf{X}}_k = \hat{\mathbf{X}}_k^- + \mathbf{K}_k (\mathbf{z}_i - \hat{\mathbf{z}}_i^-) \tag{25}$$

$$\mathbf{P}_k = \mathbf{P}_k^- - \mathbf{K}_k \mathbf{P}_{\hat{\mathbf{z}}_k \hat{\mathbf{z}}_k} \mathbf{K}_k^T \tag{26}$$

The matrices $\mathbf{P}_{\hat{\mathbf{x}}_k\hat{\mathbf{z}}_k}$ and $\mathbf{P}_{\hat{\mathbf{z}}_k\hat{\mathbf{z}}_k}$ are the state-measurement cross-covariance matrix and the measurement covariance matrix, respectively.

Special consideration is required when propagating angular quantities in the state vector through the unscented transform; the weighted average in Equation 18 may not be the simple barycentric mean, due to the discontinuity at $\pi/-\pi$ radians.

4.1 Results and Remarks

The final UKF state estimate (after 100 time steps) is:

$$\hat{\mathbf{X}}_{100} = \begin{bmatrix} 0.000 \text{ m} & 0.978 \text{ m} & 0.000 \text{ rad} & 0.000 \text{ m/s} & -0.080 \text{ m/s} \end{bmatrix}^T$$

In this case, the true and estimated state values coincide only for p_x , Θ and v_x . There is a 2 centimeter residual y axis position error, and an 8 centimeters per second residual y

axis velocity error at the end of the simulation. This is an interesting result, given that we stated initially that the UKF is typically *more* accurate than the EKF for most nonlinear problems. In the remainder of this section, we will explain why this difference exists.

If we examine a plot of the estimated position \hat{p}_y of the IMU along the y axis over time, shown in Figure 6, we see that it does not converge immediately to the true position, but exhibits a sawtooth pattern with decreasing amplitude. However, the standard deviation of $\hat{\Theta}$ for the UKF, shown in Figure 7, is exactly the same as for the EKF. Figure 4 shows the estimated y axis velocity of the IMU over time – although the IMU is stationary throughout the simulation, \hat{v}_y reaches a value of -0.16 meters per second, which is almost 50 times larger in magnitude, and opposite in sign, compared to the EKF estimate at the same time step.

Given this, and the facts that initial velocity estimate is zero and the measured acceleration of the IMU is always equal in magnitude to \mathbf{g} , why does the UKF estimate of the y axis velocity reach such a large negative value? And why, unlike the EKF, does the velocity not remain constant between measurement updates? The answer lies in the propagation of the uncertainty in the IMU orientation angle through the nonlinear process model.

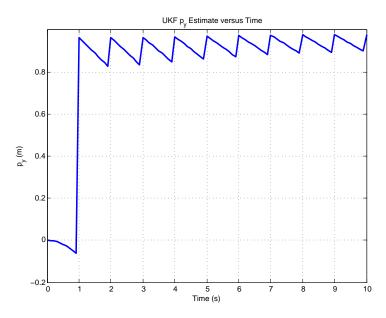


Figure 6: IMU y axis position estimate (\hat{p}_y) , for the UKF.

To illustrate how the orientation uncertainty affects the velocity estimate, consider the first time step, from t=0 seconds to t=0.1 seconds. The UKF will generate an initial set of sigma points which lie on the covariance contours in state space. For all but two of the sigma points, the value of Θ will be equal to zero (the initial state mean). The remaining sigma points, however, will have Θ values equal to ± 10 degrees (one standard deviation away from the mean). These values in radians are:

$$\theta_{4,0} = 0.276$$

 $\theta_{14,0} = -0.276$

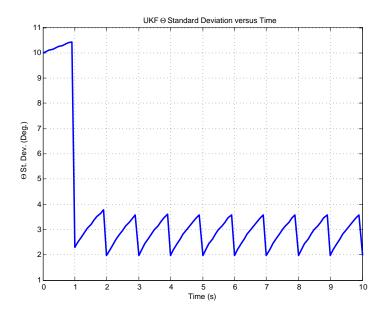


Figure 7: Standard deviation of IMU orientation estimate $(\hat{\Theta})$, for the UKF. Note that the plot is the same as for the EKF.

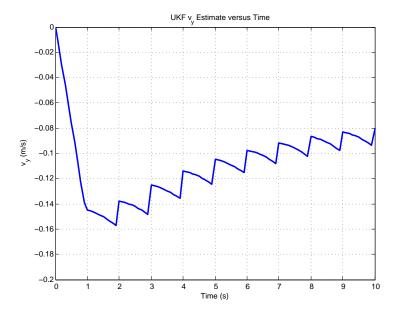


Figure 8: IMU y axis velocity estimate (\hat{v}_y) , for the UKF.

To compute the updated estimate \hat{v}_y , the UKF propagates each sigma point through the nonlinear process model. As shown in Equation 2, the measured IMU acceleration values

are first transformed by the direction cosine matrix $C(\Theta(t))$; for sigma points with $\Theta = 0$ degrees, this matrix is the identity matrix, and the measured acceleration values exactly cancel with the gravity vector. The sigma points with non-zero orientation angles ($\theta_{4,0}$ and $\theta_{14,0}$ above) produce acceleration vectors that do not exactly cancel with gravity, however. Since the acceleration vectors are symmetric about the y axis, the x components do cancel, and we are left with only the components acting along y. If we sum those components, taking gravity into consideration, we obtain:

$$\hat{v}_{y,1} = \left(W_4^{(m)}(\cos(\theta_{4,0})a_{my} + g_y) + W_{14}^{(m)}(\cos(\theta_{14,0})a_{my} + g_y)\right)\Delta t$$

$$= ((0.2)(9.44 - 9.81) + (0.2)(9.44 - 9.81))(0.1) \text{ m/s} = -0.015 \text{ m/s}$$
(27)

The result is a *net downward velocity* along the y axis of 1.5 centimeters per second at t=0.1 seconds. This velocity value continues to grow until the first measurement update. Further, it grows *nonlinearly* because we are adding process noise (to the orientation) at each time step. It is this *nonlinear propagation of orientation uncertainty* that produces the significant differences in results for the EKF and the UKF.⁴

5 Conclusions

The goal of this note was to illustrate the inherent differences between nonlinear state updates in the EKF and the UKF. We noted in Section 4.1 that the UKF produces a counterintuitive estimate of v_y , with a relatively large negative velocity when the IMU is, in fact, stationary. We also noted, paradoxically, that the EKF has a lower velocity error and a lower overall position error at the end of the simulation. However, we assert here that the estimate produced by the UKF is in fact *more correct*, because it properly accounts for the uncertainty in the orientation of the IMU.

The significant difference between the estimates produced by the EKF and UKF is an indication that the choice of estimation algorithm must be made carefully, and only after the problem under consideration has been well-studied. We demonstrated the effect of the unscented transform on the state estimate for a very simple nonlinear system – however, for more complex (i.e. real-world) systems, this type of analysis is usually much more difficult.

To the reader interested in pursuing these topics in further detail, we offer several suggestions. Crassidis and Junkins [1] provides a clear explanation of the linear Kalman filter and the EKF, while Simon [2] gives a good overview of the UKF. For a discussion of nonlinear systems in general, including nonlinear control theory, see for example Sastry [5].

References

[1] J. L. Crassidis and J. L. Junkins, *Optimal Estimation of Dynamic Systems*, ser. Applied Mathematics and Nonlinear Science. Boca Raton, Florida, USA: Chapman & Hall/CRC, April 2004, vol. 2.

⁴Note that we have selected a relatively large initial standard deviation for Θ . In practice, the uncertainty is unlikely to be this large – we chose the value specifically to illustrate the nonlinear propagation effect.

- [2] D. Simon, Optimal State Estimation: Kalman, H_{∞} , and Nonlinear Approaches, 1st ed. Hoboken, New Jersey, USA: Wiley-Interscience, June 2006.
- [3] S. J. Julier and J. K. Uhlmann, "Unscented Filtering and Nonlinear Estimation," *Proceedings of the IEEE*, vol. 92, no. 3, pp. 401–422, March 2004.
- [4] S. J. Julier, "The Scaled Unscented Transform," in *Proceedings of the American Control Conference (ACC'02)*, vol. 6, Anchorage, Alaska, USA, May 2002, pp. 4555–4559.
- [5] S. Sastry, *Nonlinear Systems: Analysis, Stability and Control*, ser. Interdisciplinary Applied Mathematics, J. E. Marsden, L. Sirovich, and S. Wiggins, Eds. New York: Springer, June 1999, vol. 10.