Unscented Kalman filter with stable embedding for simple, accurate and computationally efficient state estimation of systems on manifolds in Euclidean space

Jae-Hyeon Park | Dong Eui Chang*

School of Electrical Engineering, Korea Advanced Institute of Science and Technology, Daejeon, Korea

Correspondence
Dong Eui Chang, School of Electrical Engineering, Korea Advanced Institute of Science and Technology, 291 Daehak-ro, Yuseong-gu, Daejeon 34141, Republic of Korea. Email: dechang@kaist.ac.kr

Summary
This paper proposes a simple, accurate and computationally efficient method to apply the ordinary unscented Kalman filter developed in Euclidean space to systems whose dynamics evolve on manifolds. We use the mathematical theory called stable embedding to make a variant of unscented Kalman filter that keeps state estimates in close proximity to the manifold while exhibiting excellent estimation performance. We confirm the performance of our devised filter by applying it to the satellite system model and comparing the performance with other unscented Kalman filters devised specifically for systems on manifolds. Our devised filter has a low estimation error, keeps the state estimates in close proximity to the manifold as expected, and consumes a minor amount of computation time. Also our devised filter is simple and easy to use because our filter directly employs the off-the-shelf standard unscented Kalman filter devised in Euclidean space without any particular manifold-structure-preserving discretization method or coordinate transformation.

KEYWORDS:
stable embedding, unscented Kalman filter, attitude estimation, satellite, manifold

1 INTRODUCTION

Filtering and state estimation is an important research field that is essential in various areas of industry such as process control, tracking, navigation, and robot control. Recently, researchers are designing filtering and state estimation techniques specifically devised for systems on manifolds. This is because many systems are inherently defined on manifolds, and applying filters devised in Euclidean space directly to the systems on manifolds may cause inaccuracy and divergence in state estimation. Examples of systems on manifolds are quadrotors, winged unmanned aerial vehicles, satellites, robot arms, and arrival tracking in antenna array processing. One of the major drawbacks of working with systems on manifolds is that it does not well accommodate many tools available for systems in Euclidean space. There have been efforts to generalize the unscented Kalman filter (UKF) to the filtering and state estimation of systems on manifolds because it is one of the popular filtering techniques for nonlinear systems.

The UKF was first devised for accurate filtering and state estimation of nonlinear systems on Euclidean space. It became one of the major estimation techniques because it can approximate the statistics of state estimates up to the second order. The detailed explanation of UKF devised in Euclidean space can be found in the paper by Wan and Van der Merwe and we will call it the standard UKF to distinguish it from the variants of UKF later introduced. The direct application of the standard UKF
to systems on manifolds may not give accurate estimates, because the standard UKF is derived in Euclidean space, so state estimates deviate from the manifolds during UKF process. There are variants of UKF that have been developed for filtering and state estimation of systems on manifolds. Sipos devised an UKF for systems on Lie group, which is a special family of manifolds with group structure. This UKF used the property of Lie algebra which is a tangent space at the identity of Lie group and defined sigma points of unscented transform on the Lie algebra. Hauber et al. used an UKF for systems on Riemannian manifold, which is a family of differentiable manifolds with a Riemannian metric defined. This UKF defined sigma points of unscented transform on the tangent space of the manifold and employed geodesics and parallel transformations. These UKFs are practical since many systems of robotics are defined on sets that are both Lie group and Riemannian manifold. i.e. special orthogonal group, special Euclidean group, and n-sphere. Brossard et al. used a similar framework of UKF on manifolds as Hauber et al. and made an open-source code that can be used for practitioners not being familiar with manifolds and Lie groups. Cantelobre et al. applied the method of Brossard et al. on position estimation of autonomous underwater vehicles. Sjøberg et al. and developed an UKF for matrix Lie groups. These variants of UKF require special discretized form of systems and manifold-structure-preserving computation steps which can be computationally expensive.

In this article, we propose a method to directly apply the standard UKF devised in Euclidean space to systems on manifolds. We use the mathematical theory called stable embedding that was first founded in the paper by Chang et al. Stable embedding was previously applied to extended Kalman filter (EKF) of systems defined on special orthogonal group and the devised filter was computationally efficient. We extend systems on manifolds to an ambient Euclidean space in such a way that the manifold becomes an invariant attractor or an asymptotically stable invariant set for the extended system in the Euclidean space. Then we discretize the modified system using the ordinary Euler discretization method and apply the standard UKF derived in Euclidean space to the resultant discrete-time system. The reason why we use the Euler method is that it is the simplest one, but one could also use any off-the-shelf discretization method such as the Runge-Kutta 4th-order method which would only increase the accuracy of our proposed UKF. By inheriting the stability property of the continuous-time extended system before the discretization, this UKF has high estimation accuracy thanks to state estimates staying in close proximity of the manifold. We use the satellite attitude system model to show the excellent performance of our devised UKF. The satellite system model has its attitude quaternion state defined on 3-sphere \( S^3 = \{q \in \mathbb{H} \mid \|q\| = 1\} \), where \( \mathbb{H} \) is the set of quaternions which is isomorphic to \( \mathbb{R}^4 \) as a vector space. We have chosen the system defined on \( S^3 \), because the state space \( S^3 \) is both a Lie group and a Riemannian manifold. We compare our UKF with other UKFs designed for systems on manifolds. There are also Kalman filters available that apply to continuous-time systems directly. They require integration of differential momentum equations which can be computationally complex. Hence, applying Kalman filters to discrete form of continuous-time systems is frequently used in real applications due to fast computation. Since, one of the merits of our proposed UKF is fast computation, we choose to discretize continuous-time systems and then apply the UKFs to them.

This paper is organized as follows. Section 2 briefly explains the existing methods of UKF for systems on manifolds. Section 2 describes our method of UKF for systems on manifolds. Section 4 shows the performance of our method with a simulation on satellite attitude estimation. The conclusion is presented in Section 5. An appendix is provided at the end of the paper to give a brief review of the standard UKF and a guideline for choosing a value of the stable embedding constant later introduced.

2 REVIEW OF EXISTING METHODS OF UKF FOR SYSTEMS ON MANIFOLDS

Before we explain our method, we briefly introduce existing methods of applying UKF to systems on manifolds. Every function and manifold is assumed to be smooth in this paper unless stated otherwise. Let \( M \) be a manifold with dimension \( n_M \) in Euclidean space \( \mathbb{R}^{n_x} \) where \( n_M < n_x \). A general continuous-time system on \( M \) is given by

\[
\begin{align*}
x' &= f(x, u), & x \in M \subseteq \mathbb{R}^{n_x}, u \in \mathbb{R}^{n_u} \\
y &= g(x, u), & y \in \mathbb{R}^{n_y}
\end{align*}
\]

where \( f : M \times \mathbb{R}^{n_u} \to TM \) is the state differential equation, \( x \) is the state, \( u \) is the input, \( g : M \times \mathbb{R}^{n_u} \to \mathbb{R}^{n_y} \) is the measurement equation, \( y \in \mathbb{R}^{n_y} \) is the measurement. Here, \( TM \) denotes the tangent bundle of the manifold \( M \).

Suppose that \( f \) can be extended to an open neighborhood of the manifold \( M \) in \( \mathbb{R}^{n_x} \). If we were to naively apply the standard UKF devised in Euclidean space, we would discretize the system using the ordinary Euler discretization without considering the manifold \( M \), and apply the standard UKF. In Appendix A we give a brief explanation of the standard UKF described in the paper by Wan and Van der Merwe and the typical values of parameters used in the standard UKF. Since the manifold \( M \) is not
closed on Euclidean addition and subtraction, there arises the problem that $M$ may not be stable for the discretized system, i.e.,
discrete-time trajectories of the discretized system may diverge off the manifold $M$. Some researchers use the forced projection
to compensate for the state estimates deviating from the manifold. This method projects the state estimates $\hat{x}_{k\mid k}$ at the end
of each UKF update to the manifold $M$ by solving the following optimization problem $\hat{x}_{\text{new},k\mid k} = \arg\max_{x \in M} \|x - \hat{x}_{k\mid k}\|^2$. This
least-squares optimization problem can be solved by using an appropriate gradient descent algorithm regarding the manifold,
but it can be computationally expensive. Detailed explanation of various optimization algorithms can be found in the book by
Chong and Zak.\[13\]

Some researchers transform the coordinates of state estimate points on manifolds to vector spaces, apply the usual UKF
correction steps on the vector spaces and transform the coordinates of points in vector spaces back to the manifolds to get the
next state estimates, because many tools of Euclidean space are available in vector spaces. The UKF from the paper by Sipos[9]
is devised specifically for the case when the manifold $M$ is a Lie group. First, discretize the continuous-time system (1) defined
on the manifold $M$ into a specific form so that the states of discretized system stay on $M$ as follows:

$$x_{k+1} = f_D(x_k, u_k), \ x_k \in M \ \forall k \in \mathbb{N}_{\geq 0}.$$  \hspace{1cm} (2)

Then apply the variant of UKF that defines sigma points on the Lie algebra of the Lie group, where it is noted that the Lie
algebra is a vector space. This UKF needs computation of several coordinate transformations between the Lie group and its
Lie algebra at each step of estimation, which is computationally intensive and expensive; refer to the book by Hall[19] on Lie
groups and Lie algebras. The UKF from the paper by Hauberg et al.[9] is devised for Riemannian manifolds; refer to the book
by Carmo[20] on Riemannian manifolds. They first discretize the continuous-time system (1) defined on the manifold $M$ into a specific form so that
the states of the discretized system stay on $M$ just like the system (2). Then apply the variant of UKF that defines sigma points on the tangent spaces at the state estimates on the manifold, where it is noted that the tangent spaces of
a manifold are vector spaces. This UKF also needs computation of several coordinate transformations between the manifold
and its tangent spaces at each step of estimation, which is computationally intensive and expensive. In summary, the above
two methods proposed in the paper by Sipos[9] and the paper by Hauberg et al.[9] both need coordinate changes at each step of
estimation which can be computationally expensive.

3 \ | \ THE UKF WITH STABLE EMBEDDING

Here, we devise a variant of UKF that can apply the standard UKF derived in Euclidean space to systems on manifolds
without any additional manifold-structure-preserving discretization or coordinate transformations. Our method is simple and computationally
efficient as good as the standard UKF. We use the mathematical theory called stable embedding in the paper by Chang et al.[13] First, extend the system (1) from $M$ to $\mathbb{R}^n$, as follows:

$$\dot{x} = f_E(x, u), \ x \in \mathbb{R}^n, u \in \mathbb{R}^n$$ \hspace{1cm} (3a)

$$y = g_E(x, u), \ x \in \mathbb{R}^n, u \in \mathbb{R}^n,$$ \hspace{1cm} (3b)

such that

$$f_E(x, u) = f(x, u), \ \forall x \in M, u \in \mathbb{R}^n$$ \hspace{1cm} (4a)

$$g_E(x, u) = g(x, u), \ \forall x \in M, u \in \mathbb{R}^n,$$ \hspace{1cm} (4b)

where $f_E : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}^n$ and $g_E : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}^n$. Condition (1) implies that the extended system (3) is identical to the
original system (1) when $x \in M$. Thus, $M$ is an invariant set of the extended system (3). Then, we modify the system (3) to
make the manifold $M$ asymptotically stable, i.e. all flows of the system starting from a fixed neighborhood of $M$ asymptotically
converge to $M$ as time tends to infinity. Suppose there is a function $V : \mathbb{R}^n \to \mathbb{R}_{\geq 0}$ such that

$$M = V^{-1}(0),$$ \hspace{1cm} (5)

and

$$\langle \nabla V(x), f_E(x, u) \rangle = 0, \forall x \in \mathbb{R}^n, u \in \mathbb{R}^n,$$ \hspace{1cm} (6)
where $V^{-1}(0) = \{ x \in \mathbb{R}^n | V(x) = 0 \}$, and $\langle , \rangle$ is the Euclidean inner product. Consider the following modified system

\begin{align}
\dot{x} &= f_E(x, u) - a \nabla V(x), \quad x \in \mathbb{R}^n, u \in \mathbb{R}^n, \\
y &= g_E(x, u), \quad x \in \mathbb{R}^n, u \in \mathbb{R}^n,
\end{align}

(7a)

(7b)

where $a > 0$ is a constant parameter called stable embedding constant. Since the function $V$ attains its minimum value of 0 on $M$, $\nabla V(x) = 0$ for all $x \in M$. So the system (7) coincides with the original system (1) on $M$ and $M$ is an invariant set of (7). Since $-\nabla V$ points to the direction of decreasing $V$, it can be shown that $M$ is asymptotically stable for the system (7) on certain conditions. More concretely speaking, along each trajectory of the system (7),

$$
\frac{dV}{dt} = \langle \nabla V, f_E(x, u) - a \nabla V \rangle = -a\|\nabla V\|^2 \leq 0
$$

by (6). Since $V$ attains its global minimum value 0, $\nabla V(x) = 0$ in a neighborhood of $M$ if and only if $x \in M$. Hence, by Lyapunov’s stability theory, the manifold $M$ is asymptotically stable for the system (7). Also, the above inequality shows that the value of parameter $a$ determines the rate of convergence to $M$; refer to Theorem 1 in the paper by Chang for more on the stability conditions for $M$. An easy way to construct the function $V$ is to set $V(x) = \| \ell(x) - c \|^2$ when the manifold $M$ is defined by a level set $\ell(x) = c$ of some function $\ell(x)$.

We now add additive noises on the system (7) to produce a stochastic continuous-time system as follows:

\begin{align}
\dot{x} &= f_E(x, u) - a \nabla V(x) + w_C, \quad x \in \mathbb{R}^n, u \in \mathbb{R}^n, w_C \in \mathbb{R}^n, \\
y &= g_E(x, u) + v_C, \quad x \in \mathbb{R}^n, u \in \mathbb{R}^n, v_C \in \mathbb{R}^n,
\end{align}

(8a)

(8b)

where $x(0) \sim (x_0, P_0)$ is the initial condition of $x$, $w_C \sim (0, P_{w,C})$ is the process noise, and $v_C \sim (0, P_{v,C})$ is the measurement noise. We can apply additive noises because the system (7) is globally defined in the Euclidean space. Here, $(\mu, P)$ stands for a white noise with mean $\mu$ and covariance $P$. It is assumed that $x(0)$, $w_C$, and $v_C$ are mutually uncorrelated white noises.

We discretize the system (8) using the ordinary Euler discretization with time step $h > 0$ to apply the standard UKF derived in Euclidean space as follows:

\begin{align}
x_{k+1} &= x_k + h(f_E(x_k, u_k) - a \nabla V(x_k)) + w_D, \\
y_k &= g_E(x_k, u_k) + v_D,
\end{align}

(9a)

(9b)

where $x_k = x(kh)$, $u_k = u(kh)$, and $y_k = y(kh)$ for $k \in \mathbb{N}_{\geq 0}$. The stochastic properties are as follows: $x_0 \sim (x_0, P_0)$, $w_D \sim (0, P_{w,D})$ where $P_{w,D} \approx h P_{w,C}$, and $v_D \sim (0, P_{v,D})$ where $P_{v,D} \approx P_{v,C}/h$. A detailed explanation of discretization of a stochastic continuous-time system can be found in the book by Lewis et al. p. 84 Then we apply the standard UKF derived in Euclidean space, described in Appendix A to the system (9) directly, which is valid since the system (9) is globally defined in Euclidean space.

We call the above procedure as the UKF with stable embedding (UKF-SE). The big merit of this method is that all trajectories of the discretized form of the continuous-time system (7) starting from some neighborhood of $M$ stay in close proximity of $M$. In the following section, we apply UKF-SE to the satellite system model.

4 | SIMULATION

In this section, we conduct numerical simulations with MATLAB to compare performance of UKF-SE with other UKFs. The computation platform used for conducting these numerical studies is: CPU - Intel(R) Core(TM) i7-8700 @ 3.20GH, memory - 16GB RAM, and software - MATLAB R2018b. We use the satellite system model from the paper by Kristiansen and Nicklasson and the paper by Hajiyev and Soken for applying various UKFs. Most of the constants and satellite parameters used in the system are from the paper by Hajiyev and Soken. We organize the parameter values used in our simulation in Table I. We denote the Earth-centered inertial reference frame by $I$, the orbit reference frame by $O$, and the body reference frame by $B$ as in the paper by Kristiansen and Nicklasson. The stochastic continuous-time satellite system model is given as follows:

\begin{align}
q_{BO} &= \frac{1}{2} \wedge(q_{BO})(0; \Omega_{BO}), \\
\dot{\Omega}_{BI} &= \wedge^{-1}((\Omega_{BI} \times \Omega_{BI}) + \wedge^{-1}\tau_g + w_{\Omega,C},
\end{align}

(10a)

(10b)
where $\mathbf{q}_{BO} \in S^3$ is the satellite attitude quaternion with respect to the orbit frame; $\Omega_{BI} \in \mathbb{R}^3$ is the satellite angular velocity with respect to the inertial frame; $\Lambda(\mathbf{q}_{BO}) \in \mathbb{R}^{3 \times 4}$ with $\mathbf{q}_{BO} = (q_0, q_1, q_2, q_3)$ is given as

$$
\Lambda(\mathbf{q}_{BO}) = \begin{bmatrix}
q_0 - q_1 - q_2 - q_3 \\
q_1 q_0 - q_3 q_2 \\
q_2 q_3 - q_0 q_1 \\
q_3 q_2 - q_1 q_0
\end{bmatrix} ;
$$

$\Omega_{BO} \in \mathbb{R}^3$ is the satellite angular velocity with respect to the orbit frame given as

$$
\Omega_{BO} = \Omega_{BI} + A_{BO} \begin{bmatrix} 0 & -a_0 \end{bmatrix}^T ;
$$

$\tau_g$ is the gravity gradient torque given as

$$
\tau_g = 3a_0^2 A_{BO,3} \times \Lambda(\mathbf{q}_{BO,3}) ;
$$

$\mathbb{I}$ is the moment of the inertia matrix of the satellite; $w_{\Omega LC} \in \mathbb{R}^3$ is the process noise which follows the Gaussian distribution $N(0, \sigma_{\Omega LC}^2 \cdot I_3)$ with mean 0 and covariance $\sigma_{\Omega LC}^2 \cdot I_3$; $A_{BO} \in \mathbb{R}^{3 \times 3}$ is the rotation matrix of $\mathbf{q}_{BO} = (q_0, q_1, q_2, q_3)$ given as

$$
A_{BO} = \begin{bmatrix}
(q_0^2 + q_1^2 - q_2^2 - q_3^2) & 2(q_1 q_2 - q_0 q_3) & 2(q_0 q_3 + q_1 q_2) \\
2(q_1 q_2 + q_0 q_3) & (q_0^2 - q_1^2 + q_2^2 - q_3^2) & 2(q_2 q_3 - q_0 q_1) \\
2(q_1 q_3 - q_0 q_2) & 2(q_2 q_3 + q_0 q_1) & (q_0^2 - q_1^2 - q_2^2 + q_3^2)
\end{bmatrix} ;
$$

$\omega_0 \approx (\mu/r_0^3)^{1/2}$ is the angular velocity of the orbit with respect to the inertial frame which is a constant; $\mu$ is the Earth’s gravitational coefficient; $r_0$ is the distance between the satellite and the Earth; and $A_{BO,3}(t)$ is the 3rd column of $A_{BO}$. The initial state for the satellite system is $(\mathbf{q}_{BO}(0), \Omega_{BI}(0)) \sim N((\mathbf{q}_{BO,0}, \Omega_{BI,0}), P_0)$ where $\mathbf{q}_{BO,0} = (\cos(\pi/4), \sin(\pi/4), 0, 0)$, $\Omega_{BI,0} = (-0.01, 0.01, 0.01)[\text{rad s}^{-1}]$, and $P_0 = \text{diagonal}(10^{-8} \cdot I_4, 0.012 \cdot I_3)$.

The satellite measurement sensors are composed of a magnetometer and a rate gyro. The magnetometer measurement model is given as follows:

$$
\begin{bmatrix} H_x \\ H_y \\ H_z \end{bmatrix} = A_{BO} \begin{bmatrix} H_1 \\ H_2 \\ H_3 \end{bmatrix} + v_{\text{mag,C}} ,
$$

(11)

where $H_x$, $H_y$, and $H_z$ are the measured Earth’s magnetic field vector components in the body frame. Here, $H_1$, $H_2$, and $H_3$ are the Earth’s magnetic field vector components in the orbit frame as a function of time given as

$$
H_1 = \frac{M_e}{r_0^3} \left\{ \cos(\omega_t t, \cos(\epsilon) \sin(i) - \sin(\epsilon) \cos(i) \cos(\omega_t t)] - \sin(\omega_t t) \sin(\epsilon) \sin(\omega_t t) \right\} ,
$$

$$
H_2 = -\frac{M_e}{r_0^3} \left\{ \cos(\epsilon) \cos(i) + \sin(\epsilon) \sin(i) \cos(\omega_t t) \right\} ,
$$

$$
H_3 = \frac{2M_e}{r_0^3} \left\{ \sin(\omega_t t) \cos(\epsilon) \sin(i) - \sin(\epsilon) \cos(i) \cos(\omega_t t) \right\} - 2 \sin(\omega_t t) \sin(\epsilon) \sin(\omega_t t) \right\} ,
$$

where $M_e$ is the magnetic dipole moment of the Earth, $\epsilon$ is the magnetic dipole tilt, $i$ is the orbit inclination, and $\omega_t$ is the spin rate of the Earth. In $\mathbb{H}$, $v_{\text{mag,C}} \in \mathbb{R}^3$ is the magnetometer measurement noise which follows $N(0, \sigma_{\text{mag,C}}^2 \cdot I_3)$. The rate gyros measurement model is given as follows:

$$
\Omega_{BI,\text{meas}} = \Omega_{BI} + v_{\text{rate,C}} ,
$$

(12)

where $\Omega_{BI,\text{meas}}$ is the measured angular rate of the satellite, and $v_{\text{rate,C}} \in \mathbb{R}^3$ is the rate gyros measurement noise which follows $N(0, \sigma_{\text{rate,C}}^2 \cdot I_3)$. All the measurements are measured with a sampling frequency $f$.

The satellite system has a state $(\mathbf{q}_{BO}, \Omega_{BI}) \in S^3 \times \mathbb{R}^3$ which is 6 dimensional manifold in $\mathbb{H} \times \mathbb{R}^3$. We do not consider $\Omega_{BO}$ as a state because it does not have a differential equation. We first extend the system from $S^3 \times \mathbb{R}^3$ to $\mathbb{H} \times \mathbb{R}^3$. Define a function $V(\mathbf{q}_{BO}, \Omega_{BI}) : \mathbb{H} \times \mathbb{R}^3 \to \mathbb{R}_{\geq 0}$ as

$$
V(\mathbf{q}_{BO}, \Omega_{BI}) = \frac{1}{4} (\|\mathbf{q}_{BO}\|^2 - 1)^2 .
$$

It is easy to see that $S^3 \times \mathbb{R}^3 = V^{-1}(0)$. The derivative of $V$ is calculated as $\nabla_{\mathbf{q}_{BO}} V = (\|\mathbf{q}_{BO}\|^2 - 1) \mathbf{q}_{BO}$, and $\nabla_{\Omega_{BI}} V = 0$. This $V$ satisfies the conditions and for $V$ in Section 3. We add $-\alpha \nabla V$ where $\alpha > 0$ to the extended system to get the final
While implementing UKF loops, we use the covariance matrix of the initial state modified system as follows:

\[ \begin{align*}
\hat{q}_{BO} &= \frac{1}{2} N(q_{BO}[0; \Omega_{BO}] - \alpha(\|q_{BO}\|^2 - 1)q_{BO},
\end{align*} \tag{13a} \]

\[ \begin{align*}
\hat{\Omega}_{BI} &= \frac{1}{h}(\|\Omega_{BI}\| \times \Omega_{BI} + I_1^{-1} r_g + w_{\Omega C}),
\end{align*} \tag{13b} \]

where \((q_{BO}, \Omega_{BI}) \in \mathbb{H} \times \mathbb{R}^3\). This is the stably embedded version \(^7\) of the satellite system model \(^{10}\), i.e. \(S^3 \times \mathbb{R}^3\) is an invariant manifold of the modified system and is exponentially stable for the modified system; refer to Lemma 1 in Ko et al.'s paper.\(^{24}\)

Then, we may apply the standard UKF derived in Euclidean space directly to the system model \(^{13}\) and the measurement models given in \(^{11}\) and \(^{12}\).

Since the satellite system in nature is a continuous-time system, we generate a continuous-time trajectory of the satellite using \(^{10}\). We integrate \(^{10}\) for 200[s] and take measurements at discrete time steps of \(t = kh, k \in \mathbb{N}_{\geq 0}\) with a time step of \(h = \frac{1}{f} = 0.1[\text{s}]\). For each type of UKF, we conduct one thousand Monte Carlo runs with each run simulated for 200 seconds.

While implementing UKF loops, we use the covariance matrix of the initial state \(P_0 = \text{diag}(10^{-8} \cdot I_3, 0.01 \cdot I_3)\) and the covariance matrix of the process noise \(P_{w,D} = \text{diag}(10^{-8} \cdot I_3, h \sigma^2_{\Omega C} \cdot I_3)\). We calculate the mean squared error (MSE) of attitude estimates, and mean distance (MD) between attitude estimates and \(S^3\) which are calculated as follows, respectively:

\[ \begin{align*}
MSE_q &= E[\|q(kh) - \hat{q}_k\|^2] \approx \frac{1}{N} \sum_{i=1}^N \|q_i^{(i)}(kh) - \hat{q}_k^{(i)}\|^2, \tag{14a} \\
MD_{q,S^3} &= E[\|\hat{q}_k\| - 1] \approx \frac{1}{N} \sum_{i=1}^N \|\hat{q}_k^{(i)}\| - 1, \tag{14b} 
\end{align*} \]

where \(q(kh)\) is the real attitude at time \(t = kh\) in the continuous-time trajectory, \(\hat{q}_k\) is the attitude estimate at time \(t = kh\) using each UKF, \(E[\cdot]\) is defined as the expectation with respect to \(N = 1000\) simulations, and the superscript \((i)\) stands for the \(i\)-th trial of the \(N\) simulations. \(MSE_q\) represents estimation error of the attitude estimates. \(MD_{q,S^3}\) represents how close the attitude estimates are from \(S^3\) which is the unit quaternion set. The best \(\alpha\) with a minimum estimation error is dependent on the system and the time step, and a general guideline for choosing a value of \(\alpha\) is provided in Appendix\(^{3}\).

We conduct comparison simulation of our UKF-SE with other existing methods of applying UKFs to systems on manifolds introduced in Section\(^{2}\). We apply the following types of filters on the sampled discrete-time signals of the satellite system model:

(i) The standard UKF: Extend the satellite system model \(^{10}\) from \(S^3 \times \mathbb{R}^3\) to \(\mathbb{H} \times \mathbb{R}^3\) and apply the standard UKF discussed in Appendix\(^{4}\).

(ii) Projection by optimization: This is the forced projection method introduced in Section\(^{2}\). We project the attitude estimates \(\hat{q}_{k|k}\) to the manifold \(S^3\) at the end of each estimation step of (i) by solving \(\hat{q}_{new,k|k} = \arg \max_{q \in S^3} \|q - \hat{q}_{k|k}\|^2\). We use the Lagrangian multiplier method with the equality constraint \(\|q\|^2 - 1 = 0, \ q \in \mathbb{H}\) to solve this optimization problem.\(^{15}\) We use the MATLAB command ‘fmincon’ to implement the optimization by Lagrangian algorithm.

### TABLE 1 Parameter values used in our simulation

| Notation | Parameter | Value |
|----------|-----------|-------|
| \(I\)   | the moment of the inertia matrix of the satellite | \(\text{diag}(2.1, 2.0, 1.9) \times 10^{-3} [\text{kg m}^2]\) |
| \(\sigma_{\Omega C}\) | the standard deviation of the process noise | \(0.01 [\text{rad s}^{-1}]\) |
| \(\mu\) | the Earth’s gravitational coefficient | \(3.98601 \times 10^{14} [\text{m}^3/\text{s}^2]\) |
| \(r_0\) | the distance between the satellite and the Earth | \(6928140 [\text{m}]\) |
| \(M_e\) | the magnetic dipole moment of the Earth | \(7.943 \times 10^{15} [\text{Wb m}]\) |
| \(\varepsilon\) | the magnetic dipole tilt | \(11.7^\circ\) |
| \(i\) | the orbit inclination | \(97^\circ\) |
| \(\omega_e\) | the spin rate of the Earth | \(7.29 \times 10^{-5} [\text{rad s}^{-1}]\) |
| \(\sigma_{\text{mag,C}}\) | the standard deviation of the magnetometer measurement noise | \(200 [\text{nT}]\) |
| \(\sigma_{\text{rate,C}}\) | the standard deviation of the rate gyro measurement noise | \(1.84 \times 10^{-4} [\text{rad s}^{-1}]\) |
| \(f\) | the sampling frequency of the measurement | \(10 [\text{Hz}]\) |
(iii) Projection by formula: This is for the specific case when the system is defined on \(S^3\). After the update is over in (i), we project the state estimate \(q_{k|k}\) to the manifold \(S^3\) by using the formula \(q_{\text{proj},k|k} = \frac{q_{k|k}}{\|q_{k|k}\|}\).

(iv) UKF-SE: This is our proposed UKF. From (B3) in Appendix B and \(h = 0.1\), \(\alpha\) should satisfy \(0 < \alpha < \frac{20}{3}\). We choose \(\alpha = 2\) for our UKF-SE.

(v) Lie algebra: This UKF is devised specifically for the systems defined on Lie groups\(\mathbb{G}\) as introduced in Section 2. The specific discretized form \(\mathbb{G}\) for the quaternion attitude system is introduced in the paper by Kraft.\(\footnote{25}\) This UKF needs several coordinate transformations between \(S^3\) and its Lie algebra at each estimation step. The Lie algebra of \(S^3\) is isomorphic to \(\mathbb{R}^3\). Given a unit quaternion \(q = q_{\text{real}} + q_{\text{imag}} \in S^3\) where \(q_{\text{real}}\) is the real part and \(q_{\text{imag}}\) is the imaginary part of the unit quaternion, the coordinate transformation of \(q\) into the Lie algebra is given as

\[
v = \arctan \left( \frac{\|q_{\text{imag}}\|}{q_{\text{real}}} \right), (15)
\]

where \(v \in \mathbb{R}^3\). Given an element of the Lie algebra \(v \in \mathbb{R}^3\), the coordinate transformation of \(v\) into \(S^3\) is given as

\[
q = \cos(\|v\|) + \sin(\|v\|) \frac{v}{\|v\|}, \quad (16)
\]

where \(q \in S^3\), and \(v\) is considered as a pure imaginary quaternion. The map \(15\) is the inverse map of \(16\); see Sipos\(\footnote{26}\) for more detail.

(vi) Tangent space: This UKF is devised for the systems defined on Riemannian manifolds\(\mathbb{M}\) as introduced in Section 2. It needs several coordinate transformations between \(S^3\) and its tangent spaces at each estimation step. We introduce the coordinate transformation formulas of the Riemannian manifold \(S^3\), logarithm map and exponential map, from the paper by Hauberg.\(\footnote{26}\) Let us choose a base point \(\mu \in S^3\) and denote the tangent space of \(S^3\) at \(\mu\) by \(T_\mu S^3 = \{u \in \mathbb{R}^4 \mid \langle u, \mu \rangle = 0\}\). A point \(q \in S^3\) can be mapped to \(T_\mu S^3\) via the logarithm map,

\[
\log_\mu(q) = (q - \mu(q^T \mu)) \frac{\theta}{\sin(\theta)},
\]

\[
\theta = \arccos(q^T \mu), (17)
\]

with the convention \(0/\sin(0) = 1\). A tangent vector \(u \in T_\mu S^3\) can be mapped to \(S^3\) via the exponential map,

\[
\exp_\mu(u) = \mu \cos(\|u\|) + \sin(\|u\|) \frac{u}{\|u\|}. (18)
\]

We use the MVIRT toolbox\(\footnote{27}\) to compute these coordinate transformations between \(S^3\) and its tangent spaces, and weighted interpolation of points on \(S^3\).

**TABLE 2** Average \(MSE_q\), average \(MD_{q,S^3}\), and computation time of 1000 simulations.

| Filters             | Average \(MSE_q\)       | Average \(MD_{q,S^3}\) | Computation time [s] |
|---------------------|-------------------------|-------------------------|----------------------|
| The standard UKF    | \(5.513 \times 10^{-7}\) | \(4.026 \times 10^{-4}\) | 798.5                |
| Projection by optimization | \(3.873 \times 10^{-7}\) | \(2.310 \times 10^{-4}\) | 5882.6               |
| Projection by formula | \(2.972 \times 10^{-7}\) | \(3.523 \times 10^{-17}\) | 842.4                |
| UKF-SE (ours)       | \(2.759 \times 10^{-7}\) | \(1.405 \times 10^{-6}\) | 869.4                |
| Lie algebra         | \(1.797 \times 10^{-4}\) | \(5.308 \times 10^{-15}\) | 3315.1               |
| Tangent space       | \(3.143 \times 10^{-7}\) | \(3.416 \times 10^{-17}\) | 5832.2               |

We plot \(MSE_q\) of each UKF as a semi-log plot on Figure 1. The Lie algebra filter has \(MSE_q\) of order \(10^{-4}\) which is relatively larger than other filters, because we need to transform the points on \(S^3\) to coordinates of \(T_1 S^3\) and vice versa at each UKF loop.
If points on $S^3$ are far away from the identity, then they are far away from $T_1 S^3$, which naturally incurs large errors. So, we do not plot the Lie algebra filter result. We also do not plot MSE of the angular velocity because there is a negligible difference on MSE of $\Omega_{BI}$ estimates among the UKF’s. We calculate the average $MSE_{q}$ along time from $t = 150 [s]$ to $t = 200 [s]$ and record them on the 2nd column of Table 2. The UKF-SE has the lowest estimation error for estimating attitude among the compared UKF’s as shown on Figure 1 and Table 2 which shows numerically that UKF-SE is the most accurate UKF for systems on manifolds. We calculate the average $MD_{q,S^3}$ along time from $t = 150 [s]$ to $t = 200 [s]$ and write on the 3rd column of Table 2. The average $MD_{q,S^3}$ of UKF-SE is 0.0035 times the average $MD_{q,S^3}$ of the standard UKF, which shows that UKF-SE keeps the attitude estimates in close proximity to the manifold due to the stable embedding. The average $MD_{q,S^3}$’s of the projection by formula filter, the Lie algebra filter, and the Tangent space filter are the lowest among the UKF’s because they are composed in a specific way that constrains the attitude estimates on the manifold. However the average $MD_{q,S^3}$’s of them are not exactly 0 because of numerical rounding errors.

We also calculate the computation time of each UKF for 1000 Monte Carlo runs and record them on the 4th column of Table 2. The number of sigma points is 20 for all the simulated UKFs in our paper. We use the concept of precision operations written in the paper by Brent to perform a thorough computing complexity analysis of the UKFs. Let $M(n)$ be the number of single-precision operations required to multiply $n$-bit integers. Elementary operations, such as addition, multiplication, and reciprocals can be evaluated in $O(M(n))$ operations, with a relative error $O(2^{-n})$. Trigonometric functions, such as sin, cos, arctan can be evaluated in $O(M(n) \log(n))$ operations, with a relative error $O(2^{-n})$. The UKF-SE shows 1.004 times slower performance than the standard UKF. The reason for UKF-SE taking more time than the standard UKF is that UKF-SE carries out the additional subtraction of the stable embedding term $\alpha(\|q_k\|^2 - 1)q_k$ for each sigma point in each UKF loop in comparison with the standard UKF. The subtraction of the stable embedding term has 17 elementary operations, and there are 20 sigma points. Hence, the UKF-SE computes $O(340 M(n))$ more operations than the standard UKF in each UKF loop. The projection by optimization filter shows the slowest computation time among the simulated filters. The reason for this is that the projection by optimization filter solves an optimization problem in each UKF loop. The UKF-SE shows a minor difference in computation time from the projection by formula filter. The projection by formula filter calculates an additional normalization formula at the end of each UKF loop in comparison with the standard UKF, which is composed of 11 elementary operations. Therefore, the projection by formula filter computes $O(11 M(n))$ more operations than the standard UKF in each UKF loop. It is however noted that a general nonlinear manifold, unlike the 3-sphere $S^3$, may not even have a closed-form of normalization formula or, may only have computationally expensive normalization algorithms such as polar decomposition for special orthogonal group. The Lie algebra filter has 3.81 times slower performance than UKF-SE because the Lie algebra filter computes additional coordinate transformations between the Lie group and its Lie algebra, (15) and (16), for each sigma point at each UKF loop in comparison.
with the standard UKF. Computation of (15) and (16) is composed of 44 elementary operations and 3 trigonometric operations. Hence, the Lie algebra filter computes $O((880 + 60 \log(n))M(n))$ more operations than the standard UKF at each UKF loop. The tangent space filter has 6.71 times slower performance than UKF-SE because the tangent space filter computes additional coordinate transformations between the Riemannian manifold and its tangent spaces, (17) and (18), for each sigma point at each UKF loop in comparison with the standard UKF. Computation of (17) and (18) is composed of 65 elementary operations and 4 trigonometric operations. Therefore, the tangent space filter computes $O((1300 + 80 \log(n))M(n))$ more operations than the standard UKF in each UKF loop.

5 | CONCLUSIONS

We have devised a simple, accurate, and computationally efficient unscented Kalman filter (UKF) for the state estimation of systems on manifolds. The idea is to extend the given system to an ambient Euclidean space, apply the special term which makes the modified system asymptotically stable on the manifold, and apply the standard UKF derived in Euclidean space. Our proposed UKF is simple because it uses the off-the-shelf standard UKF devised in Euclidean space directly while keeping the state estimates in close proximity to the manifold. We apply this filter to the satellite system model which has attitude state defined on the unit quaternion space. We investigate the performance of our devised UKF by comparing with existing methods of applying UKF to systems on manifolds. Simulation on the satellite system model shows that our UKF is the most accurate compared to the other UKFs. Our UKF also has the minor computation time comparable to the standard UKF devised in Euclidean space, because our filter does not need either any manifold-structure-preserving discretization method or coordinate transformations of other UKFs. We plan to expand our stable embedding technique to invariant Kalman filters.

ACKNOWLEDGMENTS

This work has been supported by the NRF grant funded by the Korea government (MSIT) (2021R1A2C2010585).

How to cite this article:

APPENDIX

A THE STANDARD UKF

The UKF uses the concept of unscented transform to find stochastic properties of state estimates. The unscented transform uses sigma points which capture the mean and the covariance of states accurate to the second order. We briefly summarize the standard UKF algorithm devised in Euclidean space described in the paper by Wan and Van der Merwe. A discrete-time stochastic system is given as follows:

$x_{k+1} = f_D(x_k, u_k, w_k),\]
$y_k = g_D(x_k, u_k, v_k),$

where $k \in \mathbb{N}_{\geq 0}$, the state is $x_k \in \mathbb{R}^{n_x},$ the input is $u_k \in \mathbb{R}^{n_u},$ the output is $y_k \in \mathbb{R}^{n_y},$ the process noise is $w_k \in \mathbb{R}^{n_w},$ the measurement noise is $v_k \in \mathbb{R}^{n_v},$ $x_0 \sim (x_0, P_0),$ $w_k \sim (0, P_{w,D}),$ and $v_k \sim (0, P_{v,D}).$ It is assumed that $x_0,$ $w_k,$ and $v_k$ are mutually uncorrelated white noises. We want state estimates $\hat{x}_{k|k}$ for the system above. First, we initialize the state estimate as $\hat{x}_{0|0} = \bar{x}_0$ and the covariance estimate as $P_{0|0} = P_0.$ Then, we define the augmented state $\hat{x}_{k|k}^a = [\hat{x}_k^T, w_k^T, v_k^T]^T \in \mathbb{R}^L$ for $k \in \mathbb{N}_{\geq 0}$ where $L = n_x + n_w + n_v.$ The augmented state estimate is given as $\hat{\gamma}_{k|k} = [\hat{x}_{k|k}^T, 0, 0]^T \in \mathbb{R}^L.$ The augmented covariance estimate is given as $P_{k|k}^a = \text{diagonal}(P_{k|k}, P_{w,D}, P_{v,D}).$ We iteratively generate sigma points, do time update and do measurement update at
each $k \in \mathbb{N}_{\geq 1}$. The process for generating sigma points is as follows:

\[
\begin{align*}
\sigma_i^x &= [\sigma_i^x, \sigma_i^y, \sigma_i^w] \quad \text{where } \sigma_i^x \in \mathbb{R}^{n_x}, \sigma_i^y \in \mathbb{R}^{n_y}, \sigma_i^w \in \mathbb{R}^{n_w} \text{ and } i = 0, 1, 2, \ldots, 2L, \\
\sigma_0^x &= \hat{x}_{k-1|k-1}, \\
\sigma_i^y &= \hat{x}_{k-1|k-1} + \sqrt{L + \lambda} \left[ \sqrt{P_{k-1|k-1}^{a}} \right]_i \quad i = 1, \ldots, L, \\
\sigma_i^w &= \hat{x}_{k-1|k-1} - \sqrt{L + \lambda} \left[ \sqrt{P_{k-1|k-1}^{a}} \right]_{i-L} \quad i = L + 1, \ldots, 2L, \\
W_{i}^{(m)} &= \lambda / (L + \lambda), \\
W_{i}^{(c)} &= \lambda / (L + \lambda) + (1 - \gamma^2 + \beta), \\
W_i^{(m)} &= W_i^{(c)} = \lambda / (L + \lambda) \quad i = 1, \ldots, 2L,
\end{align*}
\]

where $\left[ \sqrt{P_{k-1|k-1}^{a}} \right]_i$ is the $i^{th}$ column of the matrix, and $\lambda = \gamma^2(L + \kappa) - L$. The parameters are adopted from the paper by Wan and Van der Merwe which are $\gamma = 0.01$, $\kappa = 0$, and $\beta = 2$. The process for doing time update is as follows:

\[
\begin{align*}
\sigma_{i, k|k-1}^x &= f_D(\sigma_i^x, u_{k-1}, \sigma_i^w) \quad i = 0, \ldots, 2L, \\
\hat{x}_{k|k-1} &= \sum_{i=0}^{2L} W_i^{(m)} \sigma_{i, k|k-1}^x, \\
P_{k|k-1} &= \sum_{i=0}^{2L} W_i^{(c)}[\sigma_{i, k|k-1}^x - \hat{x}_{k|k-1}] [\sigma_{i, k|k-1}^x - \hat{x}_{k|k-1}]^T, \\
y_{i, k|k-1} &= g_D(\sigma_i^y, u_{k-1}, \sigma_i^w) \quad i = 0, \ldots, 2L, \\
\hat{y}_{k|k-1} &= \sum_{i=0}^{2L} W_i^{(m)} y_{i, k|k-1}.
\end{align*}
\]

The process for doing measurement update is as follows:

\[
\begin{align*}
P_{xy} &= \sum_{i=0}^{2L} W_i^{(c)} [\sigma_{i, k|k-1}^x - \hat{x}_{k|k-1}] [y_{i, k|k-1} - \hat{y}_{k|k-1}]^T, \\
P_{yy} &= \sum_{i=0}^{2L} W_i^{(c)} [\sigma_{i, k|k-1}^x - \hat{x}_{k|k-1}] [\sigma_{i, k|k-1}^x - \hat{x}_{k|k-1}]^T, \\
K &= P_{xy} P_{yy}^{-1}, \\
\hat{x}_{k|k} &= \hat{x}_{k|k-1} + K (y_k - \hat{y}_{k|k-1}), \\
P_{k|k} &= P_{k|k-1} - K P_{yy} K^T,
\end{align*}
\]

where $y_k$ is the measurement and $\hat{x}_{k|k}$ is the state estimate at time step $k$.

**B A GUIDELINE FOR CHOOSING A VALUE OF $\alpha$**

We are given the following equation with stable embedding \((13)\) from Section \([4]\):

\[
\dot{q}_{BO} = \frac{1}{2} \Lambda(q_{BO}|0; \Omega_{BO}) - \alpha \|q_{BO}\|^2 - 1)q_{BO}.
\]

We discretize the above equation using the ordinary Euler discretization with time step $h > 0$ as below

\[
\begin{align*}
q_{BO,k+1} &= q_{BO,k} + h \left( \frac{1}{2} \Lambda(q_{BO,k}|0; \Omega_{BO,k}) - \alpha \|q_{BO,k}\|^2 - 1)q_{BO,k} \right), \\
&= (1 - ah\|q_{BO,k}\|^2 - 1)q_{BO,k} + \frac{h}{2} \Lambda(q_{BO,k}|0; \Omega_{BO,k}).
\end{align*}
\]

**Theorem 1.** If we define the error state as below

\[
e_k = \|q_{BO,k}\|^2 - 1,
\]
then the discrete error dynamics of (B1) is given as
\[ e_{k+1} = -1 + (1 + e_k) \left( (1 - ahe_k)^2 + \frac{h^2 \|\Omega_{BO,k}\|^2}{4} \right). \]  
(B2)

**Proof.** It is easy to show the following identities:
\[ \langle q_{BO,k}, \Lambda(q_{BO,k})[0;\Omega_{BO,k}] \rangle = 0, \]
\[ \|\Lambda(q_{BO,k})[0;\Omega_{BO,k}]\|^2 = \|q_{BO,k}\|^2 \Omega_{BO,k}. \]
With the above identities and the definition of the error \( e_k \),
\[ e_{k+1} = \|q_{BO,k+1}\|^2 - 1, \]
\[ = (1 - ah\|q_{BO,k}\|^2 - 1)^2 \|q_{BO,k}\|^2 + \frac{h^2}{4} \|q_{BO,k}\|^2 \Omega_{BO,k}^2 - 1, \]
which simplifies to (B2).

The next theorem gives a general guideline for choosing a value of \( \alpha \).

**Theorem 2.** Given \( |e_k| \leq \epsilon < 1 \), and \( \|\Omega_{BO,k}\| \leq U \forall k \), if we choose \( \alpha, h, \) and \( \beta \) that satisfy
\[ 0 < ah < \frac{2}{3}, \]  
(B3)
\[ 0 < \beta < ah(1 - \epsilon)(2 - ah)(2 - 2ah - ah(2 - ah)e + (ah)^2 \epsilon^2) < 1, \]  
(B4)
and
\[ \delta(h, ah, \beta) < \epsilon, \]  
(B5)
where
\[ \delta(h, ah, \beta) = h \sqrt{(1 - 2ah(1 - \epsilon) + (ah)^2 \epsilon(1 + \epsilon))e(1 + \epsilon)\frac{U^2}{2} + (1 + \epsilon)^2 \frac{h^2U^4}{16}} \]
\[ ah(1 - \epsilon)(2 - ah)(2 - 2ah - ah(2 - ah)e + (ah)^2 \epsilon^2) - \beta, \]
then the set
\[ S_\epsilon = \{ e_k \in \mathbb{R} | |e_k| < \delta(h, ah, \beta) \} \]  
(B6)
becomes a positively invariant attractor of the discrete error dynamics (B2) with a region of attraction
\[ S_\epsilon = \{ e_k \in \mathbb{R} | |e_k| \leq \epsilon \}. \]

**Proof.** From the condition (B3), \( ah \) always satisfies the below inequality
\[ 0 < ah < \min \left( \frac{2}{2e + 1}, \frac{1 + \epsilon - \sqrt{1 - \epsilon^2}}{e(1 + \epsilon)} \right), \]
because the infimum of the right side is \( \frac{2}{3} \) for \( 0 < \epsilon < 1 \). With \( |e_k| \leq \epsilon < 1 \), the following inequalities hold
\[ 0 < 1 - \epsilon \leq 1 + e_k < 1 + \epsilon, \]
\[ 0 < 2 - ahe \leq 2 - ah e_k, \]
\[ 0 < 2 - 2ah - ah(2 - ah)e + (ah)^2 \epsilon^2 \leq 2 - 2ah - ah(2 - ah)e_k + (ah)^2 e_k^2, \]  
(B7)
Choose a Lyapunov function candidate $V(e_k) = e_k^2$. From (B2) and (B7),

$$V(e_{k+1}) - V(e_k) = e_{k+1}^2 - e_k^2$$

$$= -ah(1 + e_k)(2 - ahe_k) \left(2 - 2ah - ah(2 - ah)e_k + (ah)^2 e_k^2\right) e_k^2$$

$$+ (1 - 2ah(1 + e_k) + (ah)^2 e_k(1 + e_k))e_k(1 + e_k)$$

$$+ (1 + e_k)^2 h^4 \left\| \Omega_{BO,k} \right\|^2 \frac{2}{16}$$

$$\leq -ah(1 - e)(2 - ahe)(2 - 2ah - ah(2 - ah)e + (ah)^2 e_k^2))$$

$$+ (1 - 2ah(1 - e) + (ah)^2 e(1 + e))(1 + e)$$

$$+ (1 + e)^2 h^4 U^4 \frac{2}{16}$$

(B8)

since $\left\| \Omega_{BO,k} \right\| \leq U$.

If $|e_k|$ satisfies the inequality

$$\delta(h, ah, \beta) < |e_k| \leq \epsilon,$$

which is feasible by the condition (B5), then from (B8), $V(e_{k+1}) - V(e_k) \leq -\beta e_k^2$. Consequently, $V(e_{k+1}) \leq (1 - \beta)V(e_k)$ when $e_k$ satisfies (B9). Therefore, $e_k$ decreases exponentially in magnitude so as to get into the set $S_\delta$ in finite time, since $0 < \beta < 1$ from the condition (B4).

If $|e_k|$ satisfies the inequality

$$\delta(h, ah, 0) < |e_k| \leq \delta(h, ah, \beta),$$

(B10)

where

$$\delta(h, ah, 0) = \delta(h, ah, 0)|_{\beta=0} = h \left\{ (1 - 2ah(1 - e) + (ah)^2 e(1 + e))(1 + e) \frac{U^2}{2} + (1 + e)^2 h^4 U^4 \frac{2}{16} \right\}$$

then $V(e_{k+1}) - V(e_k) \leq 0$ from (B8), which means $|e_{k+1}| \leq |e_k|$. Hence, $e_{k+1} \in S_\delta$ if $e_k$ satisfies (B10).

If $|e_k|$ satisfies the inequality

$$|e_k| \leq \delta(h, ah, 0),$$

(B11)

then by moving $V(e_k)$ to the right side in (B8) and using (B11),

$$V(e_{k+1}) \leq \left(1 - ah(1 - e)(2 - ahe)(2 - 2ah - ah(2 - ah)e + (ah)^2 e_k^2)\right) e_k^2$$

$$+ (1 - 2ah(1 - e) + (ah)^2 e(1 + e))(1 + e) h^4 U^4 \frac{2}{16}$$

$$+ (1 - ah(1 - e)(2 - ahe)(2 - 2ah - ah(2 - ah)e + (ah)^2 e_k^2)) \cdot 0^2$$

$$+ (1 - 2ah(1 - e) + (ah)^2 e(1 + e))(1 + e) h^4 U^4 \frac{2}{16}$$

$$= \delta(h, ah, 0)^2,$$

(B12)

which implies

$$|e_{k+1}| \leq \delta(h, ah, 0),$$

if $e_k$ satisfies (B11). Therefore, $e_{k+1} \in S_\delta$ if $e_k$ satisfies (B11). To sum up, every trajectory of the discrete error dynamics (B2) starting in the set $S_\delta$ remains in $S_\delta$ for all forward time, and every trajectory starting in $S_\delta$ enters the set $S_\delta$ in finite time and remains in $S_\delta$ from then on. In other words, $S_\delta$ is a positively invariant attractor of the error dynamics with $S_\epsilon$ a region of attraction.

**Remark 1.** For any $0 < \epsilon < 1$ and $U > 0$, we can find $a$, $h$, and $\beta$ that satisfy the conditions (B3), (B4), and (B5). First, we can always find a value for $ah$ that satisfies the third inequality in (B4) by choosing a sufficiently small $ah$, for example. Choose a value of $\beta$ that satisfies the first and second inequalities of (B4). Once $ah$ and $\beta$ are fixed, we can always choose a value of $h$ that satisfies (B5). Then, find a value of $\alpha$ that satisfies (B3).

**References**

1. Fresk E, Nikolakopoulos G. Full quaternion based attitude control for a quadrotor. In: European Control Conference. IEEE; 2013; Zurich, Switzerland: 3864–3869.
2. Guo A, Zhou Z, Zhu X, Bai F. Low-Cost Sensors State Estimation Algorithm for a Small Hand-Launched Solar-Powered UAV. *Sensors* 2019; 19(21): 4627.

3. Soken HE, Hajiyev C, Sakai Si. Robust Kalman filtering for small satellite attitude estimation in the presence of measurement faults. *European Journal of Control* 2014; 20(2): 64–72.

4. Jeon S, Tomizuka M, Katou T. Kinematic Kalman filter (KKF) for robot end-effector sensing. *Journal of Dynamic Systems, Measurement, and Control* 2009; 131(2).

5. Rentmeesters Q, Absil PA, Van Dooren P. A filtering technique on the Grassmann manifold. In: Proceedings of the 19th International Symposium on Mathematical Theory of Networks and Systems. ; 2010; Budapest, Hungary.

6. Julier SJ, Uhlmann JK. New extension of the Kalman filter to nonlinear systems. In: Signal Processing, Sensor Fusion, and Target Recognition VI. International Society for Optics and Photonics. ; 1997; Orlando, FL, United States: 182–193.

7. Sipos BJ. Application of the manifold-constrained unscented Kalman filter. In: IEEE/ION Position, Location and Navigation Symposium. IEEE. ; 2008; Monterey, CA, USA: 30–43.

8. Hauberg S, Lauze F, Pedersen KS. Unscented Kalman filtering on Riemannian manifolds. *Journal of mathematical imaging and vision* 2013; 46(1): 103–120.

9. Brossard M, Barrau A, Bonnabel S. A code for unscented kalman filtering on manifolds (UKF-M). In: International Conference on Robotics and Automation. IEEE. ; 2020: 5701–5708.

10. Cantelobre T, Chabbazian C, Croux A, Bonnabel S. A real-time unscented Kalman filter on manifolds for challenging AUV navigation. In: International Conference on Intelligent Robots and Systems. IEEE. ; 2020: 2309–2316.

11. Sarkka S. On unscented Kalman filtering for state estimation of continuous-time nonlinear systems. *IEEE Transactions on Automatic Control* 2011; 57(4): 1000–1004.

12. Chong EK, Zak SH. *An introduction to optimization*. Canada: John Wiley & Sons . 2013. ISBN 978-1-118-27901-4.

13. Hall B. *Lie groups, Lie algebras, and representations: an elementary introduction*. Switzerland: Springer . 2015. ISBN 978-3-319-13466-6.

14. Do Carmo MP. *Riemannian geometry*. Boston: Birkhäuser . 1992. ISBN 978-0-8176-3490-2.
23. Hajiyev C, Soken HE. Robust adaptive unscented Kalman filter for attitude estimation of pico satellites. *International Journal of Adaptive Control and Signal Processing* 2014; 28(2): 107–120.

24. Ko W, Phogat KS, Petit N, Chang DE. Tracking controller design for satellite attitude under unknown constant disturbance using stable embedding. *Journal of Electrical Engineering & Technology* 2021; 16(2): 1089–1097.

25. Kraft E. A quaternion-based unscented Kalman filter for orientation tracking. In: Proceedings of the Sixth International Conference of Information Fusion. IEEE. ; 2003: 47–54.

26. Hauberg S. Directional statistics with the spherical normal distribution. In: 21st International Conference on Information Fusion (FUSION). IEEE. ; 2018: 704–711.

27. Bergmann R. MVIRT, a toolbox for manifold-valued image restoration. In: IEEE International Conference on Image Processing. IEEE. ; 2017; Beijing, China: 201–205.

28. Brent RP. Fast multiple-precision evaluation of elementary functions. *Journal of the ACM (JACM)* 1976; 23(2): 242–251.

29. Trawny N, Roumeliotis SI. Indirect Kalman filter for 3D attitude estimation. *University of Minnesota, Dept. of Comp. Sci. & Eng., Tech. Rep* 2005; 2: 2005.