Attitude Estimation Using Parallel Quaternion Particle Filter Based on New Quaternion Distribution∗

Zhaihe ZHOU,† Yulu ZHONG, Chuanwei ZENG, and Xiangrui TIAN

Department of Test Engineering, College of Automation Engineering, Nanjing University of Aeronautics and Astronautics, Nanjing 211106, China

An improved algorithm is proposed to manage with the huge computation burden of the quaternion particle filter in aircraft attitude estimation. Based on the particle filtering frame, the new filter provides robust performance for nonlinear and non-Gaussian stochastic systems. And the posterior distribution of the new estimator is approximated as a new quaternion distribution to realize parallel computation. In addition, similar to the extended Kalman filter, this new method implements time update by replacing particles update with linear transformation to reduce computational complexity. Numerical simulations are carried out to compare the new algorithm to the extended Kalman filter and to quaternion particle filter in simulation results. The simulation results indicate that this estimation technique has faster convergence rate than the extended Kalman filter and takes less computation times than quaternion particle filter under the same accuracy as quaternion particle filter.

Key Words: Attitude Estimation, Parallel Quaternion Particle Filter, Quaternion Distribution

Nomenclature

\[ q \]: quaternion \\
\[ \tilde{\omega} \]: gyroscope measurement \\
\[ \Phi \]: orthogonal transition matrix \\
\[ \eta \]: Gaussian white-noise \\
\[ \Re \]: Cartesian coordinate system \\
\[ f \]: specific force \\
\[ r \]: reference vector \\
\[ Z \]: measurement vector \\
\[ C \]: attitude matrix \\
\[ w \]: weight of particle \\
\[ g \]: gravity vector \\
\[ I \]: identity matrix \\
\[ \mathcal{N} \]: Gaussian distribution \\
\[ M \]: particles number \\
\[ \theta \]: pitch \\
\[ \gamma \]: roll \\
\[ \psi \]: yaw

Subscripts

\[ k \]: discrete time \\
\[ t \]: continuous time \\
\[ a \]: accelerometer \\
\[ m \]: magnetometer \\
\[ n \]: reference coordinate system \\
\[ b \]: body coordinate system \\
\[ ie \]: self-rotation angle velocity of the earth \\
\[ en \]: angular velocity of body’s motion

1. Introduction

Sequential attitude estimation appears a critical part in micro-aircraft navigation system. It is a mature scheme for micro-aircraft that the data measured by gyroscope, magnetometer and accelerometer was used to determine the attitude of aircraft. Several attitude representation methods have been developed to ameliorate the mathematical models of attitude estimation. With the characteristic of nonsingular, the quaternion is widely used in the kinematics equation.1) A great number of researches have been done for improving estimated accuracy with various novel algorithms over recent decades. However, the unit quaternion must keep the normalization constraint to represent spacecraft rotation. The standardized extended Kalman filter (EKF) algorithm was used to estimate the spacecraft attitude.2) The EKF implements attitude error representation within the filtering algorithm to avoid quaternion normalization. But the method renders the error covariance nearly singular. The approach of Ref. 3) use an accessional normalization stage that normalize the quaternion by brute force to prevent singular.3) The other approach named multiplicative EKF (MEKF) was explicitly proposed, which convert the error quaternion to the three-component attitude representation.4)

In order to cope with the strongly nonlinear problem, an unscented quaternion estimator (USQUE) algorithm was proposed.5) Both MEKF and USQUE project the error quaternion resided on three-dimensional sphere \( S^3 \) onto three-dimensional Euclidean space \( E^3 \). So the Gaussian distribution is not guaranteed in \( S^3 \). Rather, they are not really quaternion estimator. Moreover, the USQUE is sensitive to the statistical distribution of the stochastic processes.6) Another Kalman-based estimator is tri-axial attitude determination (TRIAD)13) which was developed against the magnetometer distort.
A particle filter (PF) that used the modified Rodrigues parameters (MRPs) was created to avoid the quaternion’s unit normalization constraint problem and the nonlinear, non-Gaussian problem. Nevertheless, the high-dimensional normalization constraint problem and the nonlinear, nononto the body frame problem. A modiﬁed Gaussian Particle Filter that serves as the ed Rodrigues particle filter model is used to implement parallel computation and linear transformation. The discrete form of attitude kinematics differential equation can be written as

\[ q_{k+1} = \Phi(\omega_k)q_k \]  

The discrete form of attitude kinematics differential equation can be written as

\[ q_{k+1} = \Phi(\omega_k)q_k \]  

The \( \omega_k \) is an angle velocity vector at time \( k \). Provided \( \omega_k \) is constant during the sampling interval \( \Delta t \), the orthogonal transition matrix \( \Phi(\omega_k) \) yields

\[ \Phi(\omega_k) = \exp \left( \frac{1}{2} \Theta(\omega_k) \Delta t \right) \]

\[ \Theta(\omega_k) = \begin{bmatrix} 0 & -\omega_k^T \\ \omega_k & -[\omega_k \times] \end{bmatrix} \]

where \([\omega_k \times]\) denotes the cross-product matrix of \( \omega_k \).

2.2. Gyroscope measurement model

A rate-integrating gyro is often used to measure the angular rate of the aircraft. For this sensor, a widely used model is given by

\[ \dot{\omega}(t) = \omega(t) + \beta(t) + \eta_n(t) \]

\[ \beta(t) = \eta_n(t) \]

where \( \dot{\omega}(t) \) and \( \omega(t) \) denote the continuous-time measured angle velocity vector from gyroscope and the true angular velocity vector, and \( \eta_n(t) \) and \( \eta_n(t) \) are independent zero-mean Gaussian white-noise processes with

\[ E[\eta_n(t)\eta_n^*(t)] = I_{3 \times 3}\delta(t - \tau) \]

\[ E[\eta_n(t)\eta_n^*(t)] = I_{3 \times 3}\delta(t - \tau) \]

where \( \delta(\cdot) \) is a Dirac-delta function, and \( I_{3 \times 3} \) is a third order identity matrix.

To discretize the Eq. (3), there are

\[ \hat{\omega}_k = \omega_k + \beta_k + \eta_{nk} \]

\[ \beta_k = \beta_{k-1} + \Delta t\eta_{nk} \]

2.3. Observation model

In general, an observed vector \( y_k \) consists of the measured values of accelerometer and magnetometer. The corresponding vector \( r_k \), called reference vector, is composed of the local gravity acceleration and the local Earth magnetic field value. Then, the observation model can be defined as following

\[ y_k = C^o(q_k) r_k + \eta_{nk} \]

where \( \eta_{nk} \) is the measurement Gaussian white-noise vector. The relationship between the quaternion and the attitude matrix \( C^o(q_k) \) that transform the body from the frame \( \Re^B \) to the frame \( \Re^B \) at time \( k \) is

\[ C^o(q_k) = \left[ (q_{0k})^2 - \rho_k^2, 2q_{0k}\rho_k, 2q_{0k} \right]^T \]

\[ C^o(q_k) = \left[ (q_{0k})^2 - \rho_k^2, 2q_{0k}\rho_k, 2q_{0k} \right]^T \]

2.4. Accelerometer measurement model

Let \( \omega_{re} \) denotes the rotational angular velocity of the earth. The gravity acceleration and the speciﬁc force in the reference coordinate system are represented by \( g^r = [0 \ 0 \ -g]^T \) and \( f_k^r = [f_{xk}^r \ f_{yk}^r \ f_{zk}^r]^T \). Then the relationship between \( g^r \) and \( f_k^r \) can be written as

\[ f_k^r = \nu_{en} + (2\omega_{re} + \omega_{en}) × \nu_{en} - g^r \]
where $\mathbf{v}_n$ and $\omega_n$ respectively are the velocity and angular velocity for navigation frame with respect to the ECEF frame. In this paper, the true value of $\mathbf{v}_n$ and $\omega_n$ are assumed to be known for simplify the algorithm architecture that focuses on the improvement of computation speed. So the output of accelerometer $y_{ak}$ is given by
\begin{equation}
    y_{ak} = A_k f_n^a + \eta_{ak}
\end{equation}
where $A_k$ is a real attitude matrix at time $k$, and $\eta_{ak}$ is the accelerometer measurement noise.

### 2.5. Magnetometer measurement model

The output model of magnetometer measurement is given by
\begin{equation}
    y_{mk} = A_k m_k^m + \eta_{mk}
\end{equation}
The symbol $m_k^m$ is defined by $m_k^m \triangleq [\cos \alpha \ 0 \ \sin \alpha]^T$ where $\alpha$ is the dip angle.\(^{10}\) $\eta_{mk}$ is the magnetometer measurement noise.

### 3. Parallel Quaternion Particle Filter

In this section, we present the novel PQPF. The algorithm approximates the filtering distribution by new quaternion distribution. This filter inherently obeys the quaternion norm constraint, so that the particular normalization stage is unnecessary. In this algorithm, the resampling and the regularization scheme is not required due to the method that use linear transformation to maintain quaternion particles.

#### 3.1. Quaternion particle initialization

The quaternion particles can be initialized by magnetometer measurement data. The method is almost similar with the QPF initialization scheme in Ref. 6). It also gets the initial set of particles from the QPF initialization scheme in Ref. 6). It also gets the initial set of particles from the QPF initialization scheme in Ref. 6). It also gets the initial set of particles from the QPF initialization scheme in Ref. 6). It also gets the initial set of particles from the QPF initialization scheme in Ref. 6).

As shown in the left half plane of the Fig. 1, the Euler rotation axis $\mathbf{e}$ and rotation angle $\theta$ with missing information are determined by
\begin{equation}
    \mathbf{e} = \frac{\mathbf{y}_m \times m^n}{\| \mathbf{y}_m \times m^n \|}, \quad \theta = \frac{1}{2} \arccos \left( \frac{\mathbf{y}_m^T m^n}{\| \mathbf{y}_m \| \| m^n \|} \right)
\end{equation}
The incomplete Euler axis and angle can derive the initial quaternion particles by adjust the unknown angle $\gamma$ that depict in the right half plane of Fig. 1.

> Fig. 1. Initial quaternion particles determining.

\[ q = [\cos \gamma \ y_m \sin \gamma / \| y_m \|]^T [\cos \theta \ e \sin \theta]^T \] (12)

So, the angle should conform uniform distribution (i.e. $\gamma \sim U(0, 2\pi)$) to ensure the completeness of the initial particles.

#### 3.2. Importance sampling

In this part, as mentioned in above, a new quaternion distribution is used to approximate the filtering distribution. Then a simple choice for the importance sampling function $h(\cdot)$ is same as the predictive distribution since samples from this density can be easily obtained. The new quaternion probability density function (PDF) is written as
\begin{equation}
    p_{\text{new}}(q, P_{qq}) = \frac{2}{\pi^2 \sqrt{\det(P_{qq})}} \left( q^T P_{qq}^{-1} q \right)^3
\end{equation}
where $P_{qq}$ is a quaternion second moment and $\det(\cdot)$ is a determinant of matrix. This would allow $P_{qq}$ to define this new distribution much as the mean and covariance serve to define a Gaussian distribution in $R^n$.\(^7\) The random quaternion that conform the new quaternion distribution can be derived from
\begin{equation}
    q = \frac{L \xi}{\sqrt{\xi^T L^T L \xi}}
\end{equation}
where $\xi$ is random vector that conform uniformly distributed over unit hypersphere.\(^1\) and $L$ is the lower triangular matrix obtained from the Choleshy factorization of $P_{qq}$. The specific process about the generation of random quaternion is detailed in Appendix A.

#### 3.3. Measurement update

We denote by $Z_k$ the measurement constructed from observation vector $y_k$ and reference vector $r_k$ at time $k$. Then, $Z_{1:k}$ denote the measurement up to time $k$, i.e., $Z_{1:k} \equiv \{Z_1, \ldots, Z_k\}$. After receiving the observation vector $y_k$ that is statistically independent of past observations, the filtering distribution at time $k$ can be written as
\begin{equation}
    p(q_k | Z_{1:k}) = \int p(q_k | Z_{1:k-1}) p(Z_{1:k-1}) d q_{k-1}
\end{equation}
\begin{equation}
    \propto p(q_k | Z_{1:k-1}) p(Z_{1:k-1})
\end{equation}
Let $\{w_k(i)\}_{i=1}^M$ represent the associated weights of $M$ independent unit quaternion particles $\{q_k(i)\}_{i=1}^M$ that are sampled from importance PDF. The respective weights are obtained by
\begin{equation}
    \hat{w}_k(i) \propto \frac{p(Z_k | q_k(i)) p(q_k(i) | Z_{1:k-1})}{h(q_k(i) | Z_{1:k})}
\end{equation}
\begin{equation}
    \propto \frac{p(Z_k | q_k(i)) p(q_k(i) | Z_{1:k-1})}{h(q_k(i) | Z_{1:k})}
\end{equation}
The posterior probability distribution is approximated by new quaternion distribution $p_{\text{new}}(q, P_{qq})$, so the Eq. (15) can be rewritten as $p(q_k | Z_k) \approx p_{\text{new}}(q, P_{qq})$, where $P_{qq}$ is the quaternion second moment at time $k$. $P_{qq}$ can be computed by
\begin{equation}
    P_{qq} = E[q_k q_k^T] = \sum_{i=1}^M w_k(i) q_k(i) q_k(i)^T
\end{equation}
where $w_k(i)$ is a normalized weight by
\[ w_k(i) = \tilde{w}_k(i) \sum_{j=1}^{M} \tilde{w}_k(i) \]  
(18)

And the optimal quaternion estimation can be obtained by two alternative methods, the minimum mean square error (MMSE) approach and the maximum a posteriori probability (MAP) approach. Among them, the result of MMSE approach is smoother than MAP approach. So the MMSE approach is selected to estimate the optimal quaternion in this algorithm.

**Remark—brief introduction about MMSE method:**

The optimal quaternion calculated by

\[ \hat{q} = \arg \max_{\hat{q} \in \mathbb{S}^3} Tr[C_n^0(\hat{q})B^T] \]  
(19)

where

\[ B \triangleq \sum_{i=1}^{M} w_i C_n^0(q(i)) \]  
(20)

Then the trace of \( C_n^0(q)B^T \) can be written as

\[ Tr[C_n^0(q)B^T] = q^T K q \]

where

\[ K = 4 \sum_{i=1}^{M} w(i)q(i)q^T(i) - w_{tot} I_{4 \times 4} \]  
(21)

and \( I_{4 \times 4} \) is a fourth order identity matrix.

Substituting Eq. (21) to Eq. (19), we have

\[ \hat{q} = \arg \max_{\hat{q} \in \mathbb{S}^3} q^T K q \]  
(23)

Apparently, the optimal quaternion is the eigenvector of \( K \) corresponding to the maximum eigenvalue. The detailed derivation process can be found in Ref. 11).

### 3.4 Quaternion particle propagation

Assume that the posterior probability has been obtained from measurement update at time \( k \). The PQPF utilize linear transformation to approximate the particles distribution. We have to rewrite Eq. (1) firstly for accomplishing linear transformation. Let \( \tilde{\omega}_k \) and \( \tilde{\delta}\omega_k \) denote the measurement angle velocity vector from gyroscope and the corresponding bias vector, respectively. Then, the Eq. (1) can be rewritten as

\[ q_{k+1} = \Phi(\tilde{\omega}_k - \tilde{\delta}\omega_k)q_k \]

\[ = \exp \left( \frac{1}{2} \Theta(\tilde{\omega}_k - \tilde{\delta}\omega_k) \Delta t \right) q_k \]

\[ = \exp \left( \frac{1}{2} \Theta(-\tilde{\delta}\omega_k) \Delta t \right) \exp \left( \frac{1}{2} \Theta(\tilde{\omega}_k) \Delta t \right) q_k \]

\[ = \Phi(-\tilde{\delta}\omega_k)q_{k+1} \]

Substituting Eq. (14) to \( \tilde{q}_{k+1} \) and noticing that \( \Phi(\tilde{\omega}_k) \) is a constant orthogonal transition matrix, we have

\[ \tilde{q}_{k+1} = \frac{\Phi(\tilde{\omega}_k)L_k \xi}{\sqrt{\xi^T (\Phi(\tilde{\omega}_k)L_k)^T (\Phi(\tilde{\omega}_k)L_k) \xi}} \]  
(25)

Obviously, \( \tilde{q}_{k+1} \) will conform the new quaternion distribution of \( p_{\text{new}}(q; \Phi(\tilde{\omega}_k)p_{\text{qpf}}\Phi^T(\tilde{\omega}_k)) \). So, we can directly update the quaternion second moment by

\[ \tilde{P}_{\text{qpf}} = \Phi(\tilde{\omega}_k)p_{\text{qpf}}\Phi^T(\tilde{\omega}_k) \]  
(26)

Then updating the quaternion particles from \( p_{\text{new}}(q; \Phi(\tilde{\omega}_k)p_{\text{qpf}}\Phi^T(\tilde{\omega}_k)) \) through Eq. (24), we have

\[ p(q_{k+1}|Z_{1:k}) = \int p(q_{k+1}|\tilde{q}_{k+1})p(\tilde{q}_{k+1}|Z_{1:k})dq_{k+1} \]

\[ = \int p(q_{k+1}|\tilde{q}_{k+1})p(\tilde{q}_{k+1}|q_k)p(q_k|Z_{1:k})dq_k \]

\[ = \int p(q_{k+1}|\tilde{q}_{k+1})p_{\text{new}}(\tilde{q}_{k+1}; \tilde{P}_{\text{qpf}})dq_{k+1} \]  
(27)

### 4. Estimating Gyro Biases

The gyro biases estimation process utilizes the modified Gaussian Particle Filter method so as to implement parallel computation within the PQPF proposed in Section 3. Likewise, the gyro biases propagation uses linear transformation to update the gyro biases particles.

#### 4.1 gyro biases particles initialization

Equation (4) point out that, for the noise of gyro biases that conforming zero-mean Gaussian distribution, the initial gyro biases particles can also be chosen as Gaussian distribution. So, the initial distribution is set as \( p_{\text{init}}(\omega^{\text{bias}}) \sim \mathcal{N}(0, \Delta^2 \sigma^2_{\omega^{\text{bias}}}) \). With regard to the situation that the filter initial moment is not the aircraft startup time, the gyro biases can be reset by sampling twice to gyro.

#### 4.2. Implementing the modified Gaussian particle filter

The linear transformation method implements in time updating to alleviate the original GPF algorithm workload. The estimate process divides into two steps—Measurement and Time Update.

**Measurement Update**

Essentially, the GPF approximates the posterior distributions by single Gaussians distribution. Note that the predictive distribution is derived from linear transformation, so the predictive distribution is also Gaussians distribution that the particles can weighted by

\[ w_k(i) \propto \frac{p(\omega^{\text{bias}}(i)|Z_{1:k})}{h(\omega^{\text{bias}}(i)|Z_{1:k})} \]

\[ \propto \frac{p(Z_k|\omega^{\text{bias}}(i))p(\omega^{\text{bias}}(i)|Z_{1:k-1})}{h(\omega^{\text{bias}}(i)|Z_{1:k})} \]

\[ \propto \frac{p(Z_k|\omega^{\text{bias}}(i))\mathcal{N}(\omega^{\text{bias}}(i); \tilde{\omega}^{\text{bias}}_{k-1}, \Sigma^{\text{bias}}_{k-1})}{h(\omega^{\text{bias}}(i)|Z_{1:k})} \]  
(28)

where \( \omega^{\text{bias}}_{k-1} \) and \( \Sigma^{\text{bias}}_{k-1} \) are the mean and covariance of predictive distribution. The corresponding particles weight also need normalized by Eq. (18) to generate normalized weight \( \{w_k(i)\}_{i=1}^{M} \). Then the posterior distribution estimated by Gaussians which the mean and covariance are
The feasibility of linear transformation that is implemented in time update thanks to the linearity of Eq. (5). According to the property of Gaussian distribution, the gyro biases predictive distribution can be directly determined by

\[ \mathbf{\omega}_{k+1}^{bias} = \mathbf{\omega}_k^{bias} + \mathbf{0}_{3 \times 1} \]

\[ \mathbf{\Sigma}_{k+1}^{bias} = \mathbf{\Sigma}_k^{bias} + \Delta T^2 \mathbf{\sigma}_\mathbf{w}^2 \mathbf{I}_{3 \times 3} \]

(Apparently, the updated distribution conforms the Gaussian distribution, i.e. \( p(\mathbf{\omega}_{k+1}^{bias}) \sim \mathcal{N}(\mathbf{\omega}_{k+1}^{bias}, \mathbf{\Sigma}_{k+1}^{bias}) \). Due to the particles directly sampling from the predictive distribution make the importance sampling function equal to the predictive probability density function. So, the entire algorithm scheme that incorporate the gyro biases estimator into the PQPF can be illustrated in Fig. 2. The textual algorithm flow is summarized in Appendix B.

5. Simulation Result

The performance of the PQPF algorithm is examined by simulation work. The whole simulation result consists of three parts: First part is a synthetic noise case that take from Ref. 5); Second part is a slightly different synthetic noise case that use the Allan analysis result of Inertial Measurement Unit (IMU); Third part use real sensor analyzed in second part to compare the result of PQPF and QPF.

Before the discussion of the simulation result, the new quaternion distribution is verified as Fig. 3 and Fig. 4, which is fitted to Gaussian distribution. As shown in Fig. 3 the four elements of quaternion manifest expected randomness in hypersphere that is analogous to Gaussian distribution in Euclidean space. The first row and first column of the matrix plot is the scalar part of quaternion. And the rest part is the vector part of quaternion.

Then the frequency of quaternion scalar part is counted as Fig. 4, which is fitted to Gaussian distribution.

5.1. Synthetic noise case using Ref. 5) parameters

The simulation parameters are set as follows: The aircraft is equipped with a TAM, gyroscope and an accelerometer. The TAM sensor noise and gyroscope are respectively modeled by a Gaussian white-noise process with a standard deviation of 50 nT and Eq. (5) with \( \mathbf{\sigma}_\mathbf{w} = 3.1623 \times 10^{-4} \mu \text{rad/s}^{3/2} \) and
\( \sigma_c = 3.1623 \mu \text{rad}/\sqrt{s} \). And each axis initial bias of gyroscopes is set as 0.1 deg/h. The accelerometer noise is also modeled by a Gaussian white-noise process with a standard deviation of \( 2.8935 \times 10^{-4} \text{m/s}^2 \). The angular velocity vector \( \omega_{in} \) and the velocity vector \( v_{in} \) all set as 0. The sample intervals of all sensors are set as 10 s.

For the first situation, the PQPF algorithm is executed with no initial attitude errors. The number of PQPF particles is set as 1500. The PQPF estimated errors and \( \sqrt{3} \) error bounds are shown in Fig. 5.

As shown in Fig. 5, the estimated error of PQPF is always within their respective \( \sqrt{3} \) error bounds, which indicates that the algorithm can effectively estimate the attitude of aircraft.

Then, the proposed algorithm is compared with EKF, UKF (USQUE) and QPF under a same simulation scenario. As can be seen in Fig. 6, these different approaches almost work out similar result. So, the advantage of PQPF cannot reflect in no initial attitude errors situation compare with the EKF and USQUE. Thus, the initial attitude errors are going to change slightly to further evaluate these estimators.

In addition, the attitude estimation error is computed as

\[
\delta \hat{q} = 2 \arccos(\hat{q}_1)
\]

(31)

where \( \hat{q}_1 \) is the scalar component of the error quaternion \( \delta \hat{q} \) that defined by \( ^{5} \)

where \( \hat{q} \) and \( \hat{q}^{-1} \) respectively are the real quaternion and the inverse of estimated quaternion, and \( \otimes \) denotes the quaternion product operator.

In this simulation, the initial attitude errors are set to \(-5^\circ, 5^\circ\) and \(15^\circ\) for each axis. The initial bias covariance is set to \((0.2/\text{h})^2 \) and the initial attitude covariance is set to \((50^\circ)^2 \).

The particles number of PQPF and QPF are all set to 1500.

Figure 7 presents the simulation result that the QPF and PQPF have similar performance contrast to that the EKF and UKF hardly converges to the same value as PQPF within the same time span.

Both Fig. 5 and Fig. 7 have provided an adequate explanation for that the PQPF estimation is almost the same as QPF.

5.2. Synthetic noise case using Allan Analysis result

In this scenario, we use Allan Variance to analysis the noise of inertial sensors as in Ref. 18). The result of Allan Analysis\(^ {19} \) is used to generate pseudo-sensor data. The objective of Allan Analysis is Xsens product named Mti-G-710 that has clearly product output specification as Ref. 20).

Our analysis result is similar to the product manual\(^ {20} \) and the Woodmans’ analysis result,\(^ {19} \) which list as Table 1.

The time interval is set as 0.01s. Then the corresponding estimation result with no initial attitude and bias is plotted as Fig. 8. The particles number of PQPF and QPF are set as 1500.
There is no doubt that the QPF is indeed significant advantage to other estimators including PQPF. Nevertheless, the PQPF performance is closest to QPF by contrast with EKF and UKF. A dramatic situation is shown in Fig. 9 when the initial attitude errors are set to $\frac{\pi}{40}$, $\frac{\pi}{40}$ and $\frac{3}{4}$ for each axis. Apparently, the EKF never converges while the USQUE, PQPF and QPF all have well estimated results.

Figure 9 shows that the UKF using sigma-point method take long convergence time to solve the divergence phenomenon that produced by EKF. Moreover, state estimators based on particle swarm method appear significant advantage. All of the above evidence provided by different parameters simulation proof that the QPF is best estimation method in terms of the effect of estimation error. However, the computation time must be considered as for the practicability of algorithm. Therefore, the next discussion mainly concentrates on the time performance between QPF and PQPF with various particles number.

The following figure reveals the estimated accuracy effect from various particle numbers.

There is no doubt that the QPF is indeed significant advantage to other estimators including PQPF. Nevertheless, the PQPF performance is closest to QPF by contrast with EKF and UKF. A dramatic situation is shown in Fig. 9 when the initial attitude errors are set to $-50^\circ$, $50^\circ$ and $160^\circ$ for each axis. Apparently, the EKF never converges while the USQUE, PQPF and QPF all have well estimated results.

Figure 9 shows that the UKF using sigma-point method take long convergence time to solve the divergence phenomenon that produced by EKF. Moreover, state estimators based on particle swarm method appear significant advantage. All of the above evidence provided by different parameters simulation proof that the QPF is best estimation method in terms of the effect of estimation error. However, the computation time must be considered as for the practicability of algorithm. Therefore, the next discussion mainly concentrates on the time performance between QPF and PQPF with various particles number.

The following figure reveals the estimated accuracy effect from various particle numbers.

There is no doubt that the QPF is indeed significant advantage to other estimators including PQPF. Nevertheless, the PQPF performance is closest to QPF by contrast with EKF and UKF. A dramatic situation is shown in Fig. 9 when the initial attitude errors are set to $-50^\circ$, $50^\circ$ and $160^\circ$ for each axis. Apparently, the EKF never converges while the USQUE, PQPF and QPF all have well estimated results.

Figure 9 shows that the UKF using sigma-point method take long convergence time to solve the divergence phenomenon that produced by EKF. Moreover, state estimators based on particle swarm method appear significant advantage. All of the above evidence provided by different parameters simulation proof that the QPF is best estimation method in terms of the effect of estimation error. However, the computation time must be considered as for the practicability of algorithm. Therefore, the next discussion mainly concentrates on the time performance between QPF and PQPF with various particles number.

The following figure reveals the estimated accuracy effect from various particle numbers.

| Items       | Values |
|-------------|--------|
| Gyro noise: | $\sigma_n = 0.009578$ /s/$\sqrt{Hz}$ |
|             | $\sigma_v = 15.34589$ /h          |
| Acc. noise: | $\sigma_{acc} = 6 \times 10^{-7}$ m/s$^2$/Hz |
| Mag. noise: | $\sigma_{mag} = 0.348726$ mGauss   |

Figure 10 displays the Root Mean Square Error (RMSE) value for 500 to 5000 particles used in PQPF and QPF. As can be seen from Fig. 10, the RMSE value gradually decline with the particles number increasing.

From Fig. 11, the overhead of parallel computation startup render that the computation time difference between PQPF and QPF seemingly reduce when the particles from 500
(not using parallel computation) to 1000 (using parallel computation). Beyond that, it is obvious that the computational time of PQPF is less than QPF especially when the number of particles is set to a larger number. This is due to the new algorithm not requiring the resample procedure that has computational complexity of $O(N)$. Although the Monte-Carlo method is used to generate random quaternion that seems complicated. However, the parallelizability of the PQPF can complement this defect.

5.3. Real measurement case using Xsens sensor

In the final scenario, the real IMU raw data that sample from Xsens fixed in two-axis turntable is used to compare the performance of the QPF and PQPF. The output specification of Xsens as following: the inertial data are combined by rate of turn and acceleration; the output rate of inertial device is 100 Hz. Because the output configuration set as gyro and acceleration combination, the reference vector is just gravity vector, which render the three-dimensional measurement equations. The results of static experimental are plot in Fig. 12.

As shown in Fig. 12, there are explicit convergence at time span 0–3 s before the similar estimated result of QPF and PQPF. Then the turntable is started to validate the dynamic performance of PQPF. The motion of turntable is shown in the top half of Fig. 13.

Figure 13 shows that the QPF estimated result present sawtooth wave in contrast to the smooth result of PQPF, which like a Gaussian filter. Finally, the elapsed time advantage of PQPF that reduce about 30 percent computation time compare to QPF is listed in Table 2.

6. Conclusion

The parallel quaternion particle filter provides an effective estimation for attitude determination. The new algorithm belongs to a Gaussian filter that takes a particle-based approach to update the filtering distribution. Additionally, the inherent normalization procedure within the random quaternion generation renders the normalization stage unnecessary. Furthermore, the PQPF is able to be implemented in parallel without resampling. The simulation study further demonstrates that the novel filter has similar estimation performance with less computation burden.
Acknowledgments

The authors express their gratitude to the financial support by the project of China Postdoctoral Science Foundation (NO. 171980) and the project of National Natural Science Foundation of China (NO. 51505221).

References

1) Titterton, D. H. and Weston, J. L.: Strapdown Inertial Navigation Technology, Peter Pregrinus Ltd., 1997.
2) Lefferts, E., Markley, F., and Shuster, M.: Kalman Filtering for Spacecraft Attitude Estimation, *J. Guid. Control Dyn.*, 5, 5 (1982), pp. 474–479.
3) Bar-Izchack, I. Y. and Oshman, Y.: Attitude Determination from Vector Observations: Quaternion Estimation, *IEEE Trans. Aerosp. Electron. Syst.*, AES-21, 1 (1985), pp. 128–136.
4) Markley, F.: Attitude Error Representations for Kalman Filtering, *J. Guid. Control Dyn.*, 1293, 1999, pp. 1–26.
5) Crassidis, J. L. and Markley, F. L.: Unscented Filtering for Spacecraft Attitude Estimation, *J. Guid. Control Dyn.*, 126, 4 (2003), pp. 536–542.
6) Oshman, Y. and Carmi, A.: Attitude Estimation from Vector Observations Using a Genetic-algorithm-embedded Quaternion Particle Filter, *J. Guid. Control Dyn.*, 29, 4 (2006), pp. 879–891.
7) Cheng, Y.: Particle Filtering for Sequential Spacecraft Attitude Estimation, AIAA Guidance, Navigation, and Control Conference and Exhibit, 2004, p. 04-5337.
8) Psiaki, M. L.: Estimation Using Quaternion Probability Densities on the Unit Hypersphere, *J. Astronaut. Sci.*, 54, 3 (2006), pp. 415–431.
9) Shuster, M. D.: A Survey of Attitude Representations, *J. Astronaut. Sci.*, 41, 4 (1993), pp. 439–517.
10) IEEE Standard Specification Format Guide and Test Procedure for Linear, Single-Axis, Non-Gyroscopic Accelerometers, IEEE Std 1293-1998 (R2008), IEEE, New York, 1999, pp. 1–249.
11) Suh, Y. S.: Orientation Estimation Using a Quaternion-Based Indirect Kalman Filter with Adaptive Estimation of External Acceleration, *IEEE Trans. Instrum. Meas.*, 59, 12 (2010), pp. 3296–3305.
12) Markley, F. L., Cheng, Y., Crassidis, J. L., and Oshman, Y.: Averaging Quaternions, *J. Guid. Control Dyn.*, 30, 4 (2007), pp. 1193–1197.
13) Psiaki, M. L., Martel, F., and Pal, P. K.: Three Axis Attitude Determination via Kalman Filtering of Magnetometer Data, *J. Guid. Control Dyn.*, 13, 3 (1990), pp. 506–514.
14) Kotcheva, J. H. and Djuric, P. M.: Gaussian Particle Filtering, *IEEE Trans. Signal Process.*, 51, 10 (2003), pp. 2592–2601.
15) Kiani, M. and Pourtakdoust, S. H.: Adaptive Square-Root Cubature–Quadrature Kalman Particle Filter for Satellite Attitude Determination Using Vector Observations, *Acta Astronautica*, 105 (2014), pp. 109–116.
16) Carmi, A. and Oshman, Y.: Fast Particle Filtering for Attitude and Angular-Rate Estimation from Vector Observations, *J. Guid. Control Dyn.*, 32, 1 (2009), pp. 70–78.
17) Shuster, M. D.: Uniform Attitude Probability Distributions, *J. Astronaut. Sci.*, 51, 4 (2003), pp. 451–475.
18) El-Shemy, N., Hou, H., and Niu, X.: Analysis and Modeling of Inertial Sensors Using Allan Variance, *IEEE Trans. Instrum. Meas.*, 57, 1 (2008), pp. 140–149.
19) Woodman, O. J. and Woodman, C. O.: An Introduction to Inertial Navigation, *J. Navigation*, 9, 3 (2007).
20) Xsens Technologies B.V.: Mit User Manual, Revision 2019, A, pp. 28–29.

Appendix

A. Random quaternion generation algorithm

1. Compute the lower triangular matrix $L$ using Cholesky factorization to $P_{qf}$. And compute the maximum eigenvalue $(\lambda_{L,L})_{\text{max}}$ of $L^T L$.
2. Sample a vector $\xi$ from four-dimensional Gaussian distribution $N(0, I_{4 \times 4})$.
3. Normalize the vector as $z = \xi / \|\xi\|_2$.
4. Compare the ratio $\xi^T L^T L \xi / (\lambda_{L,L})_{\text{max}}$ to $z$ that conform the uniform distribution $z \sim U(0, 1)$. If $\xi^T L^T L \xi / (\lambda_{L,L})_{\text{max}} > z$, then use Eq. (14) to obtain the random quaternion, else go to step 2.

B. PQPF algorithm

1. Initialize the quaternion particles using Section 3.1 and gyro bias particles using Section 4.1.
2. Use linear transformation to predict the quaternion particles distribution and gyro bias particles distribution with Eq. (26) and Eq. (30), respectively.
3. Draw particles from $p_{\text{new}}(q, \bar{P}_{qf+1})$ and incorporate the novel information to obtain the quaternion predictive distribution using Eq. (24).
4. Compute the particles weight from Eq. (16) and Eq. (28). Then normalize the weight by Eq. (18).
5. Estimate the optimal quaternion and the quaternion second moment $P_{qf+1}$ by Eq. (19) and Eq. (17), respectively.
6. Use Eq. (29) to get gyro bias particles covariance. Then go to Step 2 for next time step.

Saburo Matunaga
Associate Editor