Particle Filtering for Attitude Estimation Using a Minimal Local-Error Representation

Yang Cheng* and John L. Crassidis†

University at Buffalo, State University of New York, Amherst, NY 14260-4400

A computationally efficient spacecraft attitude estimation particle filter is derived. The quaternion is used to represent the global attitude and the modified Rodrigues parameters are used to represent the local errors. The mean attitude is computed through the weighted sum of the modified Rodrigues parameter particles. The idea of progressive correction is employed to ensure the proper operation of the particle filter. Simulation results indicate that the performance of the particle filter exceeds that of the extended Kalman filter or the unscented Kalman filter for very large initialization errors.

I. Introduction

Spacecraft attitude estimation is the process of determining the orientation of the spacecraft with respect to a reference frame. Many nonlinear filtering methods have been applied to spacecraft attitude estimation. The extended Kalman filter (EKF) has been the most widely used nonlinear filtering method for real-time attitude estimation so far.[1] It works well in the linear regime where the linear approximation of the nonlinear dynamical system and observation model is valid. An unscented Kalman filter (UKF) has been proposed for attitude estimation, which uses a set of deterministically chosen *sigma points* to more accurately map a probability distribution and which is more robust than the EKF under large initial attitude-error conditions.[2] Other improved nonlinear filtering algorithms for attitude estimation are surveyed in Ref. 3. Most of these algorithms estimate the mean and covariance of the state vector based on second- or higher-order approximations of nonlinear functions. Although the mean and covariance are the sufficient statistics of a Gaussian distribution, they are not sufficient to represent a general probability distribution. When these methods are applied to strongly nonlinear and non-Gaussian estimation problems, where the probability distribution of the state vector may be multi-modal, heavy-tailed, or skewed, desired performance characteristics may not be obtained.

Particle filters[4] (PFs) have gained much attention in recent years. Like other approximate nonlinear filtering methods, the ultimate objective of the PF is to reconstruct the posterior probability density function (pdf) of the state vector, or the probability distribution of the state vector conditional on all the available measurements. However, the approximation of the PF is vastly different from that of conventional nonlinear filters. Approximating a continuous distribution of interest by a finite (but large) number of weighted random samples or particles in the state space, the PF assumes no functional form for the posterior probability distribution. In the simplest form of the PF, the particles are propagated through the dynamic model and then weighted according to the likelihood function, which determines how closely the particles match the measurements. Those that best match the measurements are multiplied and those that do not are discarded. In principle, the PF (with an infinity of particles) can approximate the posterior probability distribution of any form and solve any nonlinear and/or non-Gaussian estimation problems. In practice, however, it is nontrivial to design a PF with a relatively small number of particles. The performance of the PF heavily depends on whether the particles are located in the *right* regions of the state space. When the measurements are accurate, which is typical of spacecraft attitude estimation and many other estimation problems, the likelihood function concentrates in a very small region of the state space and the particles propagated through the dynamic model are more often than not located outside the significant regions of

*Research Assistant Professor, Department of Mechanical and Aerospace Engineering, cheng3@eng.buffalo.edu. Senior Member AIAA.

†Professor, Department of Mechanical and Aerospace Engineering, johnc@buffalo.edu, Associate Fellow AIAA.

the likelihood function. State estimates such as the mean and covariance approximated with these particles are imprecise. This problem becomes even worse when the initial estimation errors are large, for example, a few orders of magnitude larger than the sensor accuracy. In this paper a PF based on the idea of progressive correction[5] is given, which maintain diversity of the particle and guides the particles to the right regions of the state space gradually.

A quaternion-based PF for spacecraft attitude estimation is introduced in Ref. 6. The quaternion representation is preferred over other representations because of the bilinear nature of the kinematics, with a closed-form discrete-time solution, and because it contains no singularities.[7] However the unity norm of the quaternion must be maintained. Because the quaternion estimate in a PF is given by a weighted sum of quaternion particles, it is not guaranteed to preserve the norm constraint. An extra step based on either a minimum mean-square error[8] or maximum *a posteriori*[6] approach must be employed, which can add significant computations to the PF algorithm. In this paper this step is circumvented by using an unconstrained representation for the local error-space. The approach is similar to the concept in the UKF of Ref. 2 but differences exist that will be seen here.

The organization of this paper proceeds as follows. First, the PF is reviewed. Then, a brief review of the quaternion attitude kinematics and gyro model is provided. Next, an attitude estimation PF using a local-error representation is provided. Finally, the PF is compared with the EKF and UKF using simulated three-axis magnetometer and gyro measurements of the Tropical Rainfall Measurement Mission (TRMM).

II. Particle Filtering

The particular PF proposed for attitude estimation is based on the Bootstrap Filter (BF)[9] and is reviewed in this section. A general discrete-time state-space model consists of the system dynamical model and the measurement model. The system model relates the current state vector, $\mathbf{x}_k$, to the one-stage ahead state vector, $\mathbf{x}_{k+1}$, and the measurement model relates the state vector $\mathbf{x}_k$ to the measurement vector $\tilde{\mathbf{y}}_k$:

$$\mathbf{x}_{k+1} = \mathbf{f}_k(\mathbf{x}_k, \mathbf{u}_k, \mathbf{w}_k) \tag{1}$$

$$\tilde{\mathbf{y}}_k = \mathbf{h}_k(\mathbf{x}_k, \mathbf{v}_k) \tag{2}$$

In the above equations the system and measurement functions are denoted by $\mathbf{f}_k$ and $\mathbf{h}_k$, respectively. The subscript k in $\mathbf{f}_k$ and $\mathbf{h}_k$ indicates that the functions themselves can be time-varying. The measurement sampling period is denoted by $T_k = t_{k+1} - t_k$. The vector $\mathbf{u}_k$ is the deterministic input. The process noise $\mathbf{w}_k$ and the measurement noise $\mathbf{v}_k$ are assumed to be zero-mean white noise sequences. The distributions of the mutually independent $\mathbf{x}_0$, $\mathbf{w}_k$, and $\mathbf{v}_k$, denoted by $p_{\mathbf{x}_0}(\mathbf{x}_0)$, $p_{\mathbf{w}_k}(\mathbf{w}_k)$ and $p_{\mathbf{v}_k}(\mathbf{v}_k)$, respectively, are assumed to be known. No Gaussian assumptions for these distributions are needed.

Now the procedure of a BF with N weighted particles is reviewed. The sets of particles and their associated weights at t_k and t_{k+1} are denoted by $\{\mathbf{x}_k^{(i)}, w_k^{(i)}\}$ and $\{\mathbf{x}_{k+1}^{(i)}, w_{k+1}^{(i)}\}$, respectively, where $i = 1, \cdots, N$. Four steps, namely, prediction, update, resampling and regularization (roughening), constitute a filter cycle.

At the prediction step, the particles are propagated through the following equation while their weights remain unchanged:

$$\mathbf{x}_{k+1}^{(i)} = \mathbf{f}_k(\mathbf{x}_k^{(i)}, \mathbf{u}_k, \mathbf{w}_k^{(i)}) \tag{3}$$

where N samples $\mathbf{w}_k^{(i)}$ of the process noise are drawn according to the pdf $p_{\mathbf{w}_k}(\mathbf{w}_k)$, denoted by $\mathbf{w}_k^{(i)} \sim p_{\mathbf{w}_k}(\mathbf{w}_k), i = 1, \cdots, N$.

At the update step that follows, the weight associated with each particle is updated based on the likelihood function $p_{k+1}(\tilde{\mathbf{y}}_{k+1}|\mathbf{x}_{k+1}^{(i)})$:

$$w_{k+1}^{(i)} = w_k^{(i)} \cdot p(\tilde{\mathbf{y}}_{k+1}|\mathbf{x}_{k+1}^{(i)}) \tag{4a}$$

$$w_{k+1}^{(i)} \leftarrow \frac{w_{k+1}^{(i)}}{\sum_{i=1}^{N} w_{k+1}^{(i)}} \tag{4b}$$

where $\leftarrow$ denotes *assigned to*. Note that $\sum_{i=1}^{N} w_{k+1}^{(i)} = 1$ after the normalization done by Eq. (4b). When the measurement noise is additive, $\tilde{\mathbf{y}}_{k+1} = \mathbf{h}_{k+1}(\mathbf{x}_{k+1}) + \mathbf{v}_{k+1}$, the likelihood function has a simple form: $p_{k+1}(\tilde{\mathbf{y}}_{k+1}|\mathbf{x}_{k+1}^{(i)}) = p_{\mathbf{v}_{k+1}}(\tilde{\mathbf{y}}_{k+1} - \mathbf{h}_k(\mathbf{x}_{k+1}^{(i)}))$.

The above prediction and update steps implement a cycle of the sequential importance sampling algorithm. The variance associated with the weights in sequential importance sampling can only increase over time and eventually all but one particle will have negligible weight.[10] A common practice to overcome this degeneracy problem is resampling. The resampling scheme discards particles with small weights and multiplies particles with large weights. It is implemented by drawing samples (with replacement) N times from $\{\mathbf{x}_{k+1}^{(i)}, w_{k+1}^{(i)}\}$ to obtain N equally weighted particles, $\{\mathbf{x}_{k+1}^{(i)}, 1/N\}$. The total number of particles remains unchanged and no new particles are created after resampling. The normalized weight $w_{k+1}^{(i)}$ may be interpreted as the probability of occurrence for each particle. The probability of the particle $\mathbf{x}_{k+1}^{(i)}$ being chosen once is approximately $w_{k+1}^{(i)}$ and after resampling $\mathbf{x}_{k+1}^{(i)}$ will be multiplied approximately $(N \cdot w_{k+1}^{(i)})$ times. The resampling algorithm is a black-box algorithm that takes as input the normalized weights and particle indices, and outputs new indices. It has nothing to do with the particles' dimension, values and so on.

Since resampling may generate many identical particles and therefore greatly decrease the number of distinct particles, it is usually followed by a regularization step, which adds small noise to the resampled particles, to increase particle diversity:[5]

$$\mathbf{x}_{k+1}^{(i)} \leftarrow \mathbf{x}_{k+1}^{(i)} + \boldsymbol{\nu}_{k+1}^{(i)} \tag{5}$$

with $\boldsymbol{\nu}_{k+1}^{(i)}$ an independent jitter drawn from a Gaussian distribution $\mathcal{N}(\mathbf{0}, h^2\Sigma_{k+1})$, where h^2 is a tuning parameter and

$$\Sigma_{k+1} = \frac{1}{N-1}\sum_{i=1}^{N} \tilde{\mathbf{x}}_{k+1}^{S(i)} \tilde{\mathbf{x}}_{k+1}^{S(i)T} \tag{6}$$

$$\tilde{\mathbf{x}}_{k+1}^{S(i)} = \mathbf{x}_{k+1}^{S(i)} - \hat{\mathbf{x}}_{k+1}^{S} \tag{7}$$

$$\hat{\mathbf{x}}_{k+1}^{S} = \frac{1}{N}\sum_{i=1}^{N} \mathbf{x}_{k+1}^{S(i)} \tag{8}$$

Note that Σ is not a function of the particle weights. In tuning h, tradeoffs have to be made between spawning more distinct particles (large noise) and not altering the original distribution too much (small noise).

The resampling and regularization steps are used to guarantee the proper performance of the BF, but are not required for processing the filter. They may be applied at every cycle, as in the original BF, or when the effective sample size, N_{eff}, is small. If resampling is done at every cycle, then Eq. (4a) reduces to $w_{k+1}^{(i)} = p(\tilde{\mathbf{y}}_{k+1}|\mathbf{x}_{k+1}^{(i)})$.

The effective sample size is approximated by[4] $N_{\text{eff}} \approx 1/\sum_{i=1}^{N}(w_{k+1}^{(i)})^2$, which is a measure of the variation of the (normalized) weights. If only very few particles have significant weights while others are negligible (the sum is always 1), then $N_{\text{eff}} \approx 1$; if all the particles are nearly equally weighted, then $N_{\text{eff}} \approx N$. Usually, large N_{eff} is desired, but large N_{eff} alone does not necessarily mean vast diversity among particles, because N_{eff} is not a function of the particles $\mathbf{x}_{k+1}^{(i)}$ and large N_{eff} may correspond to the unfavorable case in which most of the particles are identical (due to resampling).

The side effect of resampling and regularization is that they introduce additional Monte Carlo variations. The resampling step has the "cut-tail" effect and results in many identical particles. Applying resampling too frequently may eliminate potentially good particles as well. The regularization step increases the sample covariance of the particles.

In the BF as well as other PFs, what is the most essential in the evolution of the filter is the particles and their associated weights. The estimates such as the mean and covariance are derived from the particles and computed using the following equations:

$$\hat{\mathbf{x}}_{k+1}^{+} = E\{\mathbf{x}_{k+1}|\tilde{\mathbf{Y}}_{k+1}\} \approx \sum_{i=1}^{N} w_{k+1}^{(i)} \mathbf{x}_{k+1}^{(i)} \tag{9a}$$

$$P_{k+1} = E\{(\mathbf{x}_{k+1} - \hat{\mathbf{x}}_{k+1}^{+})(\mathbf{x}_{k+1} - \hat{\mathbf{x}}_{k+1}^{+})^T|\tilde{\mathbf{Y}}_{k+1}\} \approx \sum_{i=1}^{N} w_{k+1}^{(i)} \tilde{\mathbf{x}}_{k+1}^{(i)} \tilde{\mathbf{x}}_{k+1}^{(i)T} \tag{9b}$$

where $\tilde{\mathbf{x}}_{k+1}^{(i)} = \mathbf{x}_{k+1}^{(i)} - \hat{\mathbf{x}}_{k+1}^{+}$. The superscript + is used to denote updated value because a distinction between propagated and updated values is needed in the attitude estimation PF. It is advised that when the mean and covariance are computed, they should be computed after the update but before resampling and regularization.[4]

The BF makes few assumptions about the dynamical system and measurement models, and involves only straightforward function evaluations and random sampling schemes, thus being very easy to implement. The function evaluations include the system function, measurement function, and the likelihood function (possibly up to a constant). The following sampling steps are needed: drawing samples from $p_{\mathbf{x}_0}(\mathbf{x}_0)$ at the initial time, drawing samples from $p_{\mathbf{w}_k}(\mathbf{w}_k)$ at the prediction step, generating uniform random numbers at the resampling step, and drawing samples from $\mathcal{N}(\mathbf{0}, h^2\Sigma_{k+1})$ at the regularization step. Note that in the BF it is not required to draw samples from the likelihood or evaluate the prior.

III. Attitude Kinematics and Gyro Models

The attitude matrix A is a proper orthogonal matrix, i.e. its inverse is given by its transpose and its determinant is +1. For spacecraft applications the attitude mapping is usually applied from the reference frame to the spacecraft body frame. Mathematically, the mapping from the reference frame to the body frame is given by $\mathbf{b} = A\mathbf{r}$, where $\mathbf{b}$ is the body-frame representation of a vector and $\mathbf{r}$ is the reference-frame representation of the vector. The quaternion is a four-dimensional vector, defined as $\mathbf{q} \equiv [\boldsymbol{\varrho}^T \;\; q_4]^T$, with $\boldsymbol{\varrho} \equiv [q_1 \;\; q_2 \;\; q_3]^T = \mathbf{e}\sin(\vartheta/2)$ and $q_4 = \cos(\vartheta/2)$, where $\mathbf{e}$ is the unit Euler axis and ϑ is the rotation angle. Since a four-dimensional vector is used to describe three dimensions, the quaternion components cannot be independent of each other. The quaternion satisfies a single constraint given by $\mathbf{q}^T\mathbf{q} = 1$. The attitude matrix is related to the quaternion by

$$A(\mathbf{q}) = \left(q_4^2 - ||\boldsymbol{\varrho}||^2\right) I_{3\times 3} + 2\boldsymbol{\varrho}\,\boldsymbol{\varrho}^T - 2q_4[\boldsymbol{\varrho}\times] \tag{10}$$

where $I_{3\times 3}$ is a 3×3 identity matrix and $[\boldsymbol{\varrho}\times]$ is the standard cross product matrix.[3]

The quaternion kinematics equation is given by

$$\dot{\mathbf{q}} = \begin{bmatrix} q_4 I_{3\times 3} + [\boldsymbol{\varrho}\times] \\ -\boldsymbol{\varrho}^T \end{bmatrix} \boldsymbol{\omega} \equiv \frac{1}{2}\Xi(\mathbf{q})\boldsymbol{\omega} \tag{11}$$

where $\boldsymbol{\omega}$ is the three-component angular rate vector. Successive rotations can be accomplished using quaternion multiplication. Here we adopt the convention of Refs. 1 and 7 which multiply the quaternions in the same order as the attitude matrix multiplication: $A(\mathbf{q}')A(\mathbf{q}) = A(\mathbf{q}' \otimes \mathbf{q})$. with

$$\mathbf{q}' \otimes \mathbf{q} = \Big[\Xi(\mathbf{q}) \;\vdots\; \mathbf{q}\Big]\,\mathbf{q}' \tag{12}$$

Also, the inverse quaternion is given by $\mathbf{q}^{-1} = \begin{bmatrix} -\boldsymbol{\varrho}^T & q_4 \end{bmatrix}^T$.

A common sensor that measures the angular rate is a rate-integrating gyro. For this sensor, a widely used model is given by[11]

$$\tilde{\boldsymbol{\omega}}(t) = \boldsymbol{\omega}(t) + \boldsymbol{\beta}(t) + \boldsymbol{\eta}_v(t) \tag{13a}$$

$$\dot{\boldsymbol{\beta}}(t) = \boldsymbol{\eta}_u(t) \tag{13b}$$

where $\tilde{\boldsymbol{\omega}}(t)$ is the continuous-time measured angular rate, and $\boldsymbol{\eta}_v(t)$ and $\boldsymbol{\eta}_u(t)$ are independent zero-mean Gaussian white-noise processes with

$$E\left\{\boldsymbol{\eta}_v(t)\boldsymbol{\eta}_v^T(\tau)\right\} = I_{3\times 3}\sigma_v^2\delta(t-\tau) \tag{14a}$$

$$E\left\{\boldsymbol{\eta}_u(t)\boldsymbol{\eta}_u^T(\tau)\right\} = I_{3\times 3}\sigma_u^2\delta(t-\tau) \tag{14b}$$

where $\delta(t-\tau)$ is the Dirac delta function.

IV. Attitude Estimation Particle Filter

Quaternions are desirable for attitude estimation because of their singular-free property. However the norm constraint must be maintained. Straightforward implementation of the particle filter estimate, given

by Eq. (9a), clearly shows that a weighted sum average of quaternions does not produce a normalized vector in general. To overcome this issue a local minimal-error representation, defined by the modified Rodrigues parameters (MRPs),[7] is used while maintaining the quaternion as the global representation. The standard positive form of the MRPs are defined by $\mathbf{p} \equiv \boldsymbol{\varrho}/(1+q_4)$, with $q_4 \geq 0$. Note that the standard positive form has a vector that is maintained within a unit sphere.

The state vector for the particle filter is given by $\mathbf{x} \equiv [\boldsymbol{\delta}\mathbf{p}^T \;\; \mathbf{b}^T]^T$, where $\boldsymbol{\delta}\mathbf{p}$ is the local error-MRP vector which is related to the local error-quaternion, $\boldsymbol{\delta}\mathbf{q} \equiv [\boldsymbol{\delta\varrho}^T \;\; \delta q_4]^T$, through $\boldsymbol{\delta}\mathbf{p} = f[\boldsymbol{\delta\varrho}/(1+\delta q_4)]$, where f is chosen to be 4 so that $\boldsymbol{\delta\varrho}$ is equal to physically intuitive roll, pitch and yaw angles for small errors. The steps involved for using the local error-MRPs in a particle filter are now shown in detail.

First an initial set of particles, $\mathbf{x}_0^{(i)} \equiv [\boldsymbol{\delta}\mathbf{p}_0^{(i)T} \;\; \mathbf{b}_0^{(i)T}]^T$, is drawn from any desired distribution. The error-MRP particles are checked to make sure that they have the minimum spread (or dispersion). In many cases, this means that the error-MRP particles are within the sphere of radius f; the ones that do not satisfy this requirement are found using the inequality $||\boldsymbol{\delta}\mathbf{p}_0^{(i)}|| > f$. The MRPs that satisfy this inequality are replaced with $-f^2\,\boldsymbol{\delta}\mathbf{p}_0^{(i)}/||\boldsymbol{\delta}\mathbf{p}_0^{(i)}||^2$, which represents the same rotation but is now within the sphere of radius f. States estimates at the initial time for the quaternion and gyro biases are given, denoted by $\hat{\mathbf{q}}_0^+$ and $\hat{\mathbf{b}}_0^+$, respectively. The set of initial quaternions $\mathbf{q}_0^{(i)}$ are computed by "adding" $\boldsymbol{\delta}\mathbf{p}_0^{(i)}$ to $\hat{\mathbf{q}}_0^+$, which is implemented as

$$\delta q_{4_0}^{(i)} = \frac{f^2 - ||\boldsymbol{\delta}\mathbf{p}_0^{(i)}||^2}{f^2 + ||\boldsymbol{\delta}\mathbf{p}_0^{(i)}||^2} \tag{15a}$$

$$\boldsymbol{\delta}\mathbf{q}_0^{(i)} = \begin{bmatrix} (1+\delta q_{4_0}^{(i)})\boldsymbol{\delta}\mathbf{p}_0^{(i)}/f \\ \delta q_{4_0}^{(i)} \end{bmatrix} \tag{15b}$$

$$\mathbf{q}_0^{(i)} = \boldsymbol{\delta}\mathbf{q}_0^{(i)} \otimes \hat{\mathbf{q}}_0 \tag{15c}$$

The initialization of the attitude estimation PF is summarized as follows:

- Generate N initial error-MRP particles $\boldsymbol{\delta}\mathbf{p}_0^{(i)}$ and N initial gyro bias error particles $\mathbf{b}_0^{(i)}$ according to the initial state probability distribution.
- Generate the initial quaternion particles $\mathbf{q}_0^{(i)}$ by "adding" $\delta\mathbf{p}_0^{(i)}$ to the initial quaternion estimate $\hat{\mathbf{q}}_0^+$.
- The weights are set to $w_0^{(i)} = 1/N$.

The next set of steps is repeated for each time propagation cycle in the PF. A discrete-time propagation is used with time interval Δt. If the inequality $||\Delta t\,\boldsymbol{\omega}|| << 1$ is valid, which is true for most spacecraft applications, then the covariance of the discrete-time process noise derived from the models in Eqs. (11) and (13) is given by[11]

$$Q = \begin{bmatrix} \left(\sigma_v^2\Delta t + \frac{1}{3}\sigma_u^2\Delta t^3\right) I_{3\times 3} & -\left(\frac{1}{2}\sigma_u^2\Delta t^2\right) I_{3\times 3} \\ -\left(\frac{1}{2}\sigma_u^2\Delta t^2\right) I_{3\times 3} & \left(\sigma_u^2\Delta t\right) I_{3\times 3} \end{bmatrix} \tag{16}$$

which is time invariant. Zero-mean Gaussian samples of the discrete-time process noise are drawn according to Q. This can be accomplished using the eigenvalue/eigenvector decomposition of Q. The samples are drawn using the variances defined by the eigenvalues and then these samples are rotated using the eigenvector matrix to produce correlated samples with covariance Q. Denote this vector of random samples as $\mathbf{w}_k = [\mathbf{w}_{1_k}^T \;\; \mathbf{w}_{2_k}^T]^T$, where $\mathbf{w}_{1_k}^T$ and $\mathbf{w}_{2_k}^T$ are 3×1 vectors. The angular velocity particles are given by

$$\boldsymbol{\omega}_k^{(i)} = \tilde{\boldsymbol{\omega}}_k^{(i)} - \mathbf{b}_k^{(i)} + \mathbf{w}_{1_k} \tag{17}$$

and the bias particles are given by

$$\mathbf{b}_{k+1}^{(i)} = \mathbf{b}_k^{(i)} + \mathbf{w}_{2_k} \tag{18}$$

Similarly, the zero-mean Gaussian samples can be drawn based on a Cholesky decomposition of Q:

$$Q = S^T S = \begin{bmatrix} \sqrt{\frac{\sigma_v^2}{\Delta t} + \frac{1}{12}\sigma_u^2\Delta t}\cdot I_{3\times 3} & -\frac{1}{2}\sigma_u\sqrt{\Delta t}\cdot I_{3\times 3} \\ 0_{3\times 3} & \sigma_u\sqrt{\Delta t}\cdot I_{3\times 3} \end{bmatrix} \begin{bmatrix} \sqrt{\frac{\sigma_v^2}{\Delta t} + \frac{1}{12}\sigma_u^2\Delta t}\cdot I_{3\times 3} & 0_{3\times 3} \\ -\frac{1}{2}\sigma_u\sqrt{\Delta t}\cdot I_{3\times 3} & \sigma_u\sqrt{\Delta t}\cdot I_{3\times 3} \end{bmatrix}$$

The matrix S^T is used to generate $\mathbf{w}_{1_k}$ and $\mathbf{w}_{2_k}$ by weighting vectors of random samples of unit variance. The error-quaternion for the i^{th} particle, denoted by $\boldsymbol{\delta}\mathbf{q}^{(i)} \equiv [\boldsymbol{\delta\varrho}^{(i)T} \;\; \delta q_4^{(i)}]^T$ is computed using

$$\delta q_{4_k}^{(i)} = \frac{f^2 - ||\boldsymbol{\delta}\mathbf{p}_k^{(i)}||^2}{f^2 + ||\boldsymbol{\delta}\mathbf{p}_k^{(i)}||^2} \tag{19a}$$

$$\boldsymbol{\delta}\mathbf{q}_k^{(i)} = \begin{bmatrix} (1 + \delta q_{4_k}^{(i)})\boldsymbol{\delta}\mathbf{p}_k^{(i)}/f \\ \delta q_{4_k}^{(i)} \end{bmatrix} \tag{19b}$$

The quaternion particles at time t_k are then given by

$$\mathbf{q}_k^{(i)} = \boldsymbol{\delta}\mathbf{q}_k^{(i)} \otimes \hat{\mathbf{q}}_k^+ \tag{20}$$

where $\hat{\mathbf{q}}_k^+$ is the updated estimated quaternion at time t_k. These quaternions are propagated to the next time step using

$$\mathbf{q}_{k+1}^{(i)} = \Omega(\boldsymbol{\omega}_k^{(i)})\mathbf{q}_k^{(i)} \tag{21}$$

with

$$\Omega(\boldsymbol{\omega}_k^{(i)}) \equiv \begin{bmatrix} \cos\left(0.5||\boldsymbol{\omega}_k^{(i)}||\Delta t\right) I_{3\times 3} - \left[\boldsymbol{\psi}_k^{(i)}\times\right] & \boldsymbol{\psi}_k^{(i)} \\ -\boldsymbol{\psi}_k^{(i)T} & \cos\left(0.5||\boldsymbol{\omega}_k^{(i)}||\Delta t\right) \end{bmatrix} \tag{22}$$

where $\boldsymbol{\psi}_k^{(i)} \equiv \sin\left(0.5||\boldsymbol{\omega}_k^{(i)}||\Delta t\right) \boldsymbol{\omega}_k^{(i)}/||\boldsymbol{\omega}_k^{(i)}||$, and Δt is the sampling interval in the gyro. Note that the estimated quaternion is also propagated using $\hat{\mathbf{q}}_{k+1}^- = \Omega(\hat{\boldsymbol{\omega}}_k^+)\hat{\mathbf{q}}_k^+$, where the superscript $-$ denotes a propagated quantity and $\hat{\boldsymbol{\omega}}_k^+ = \tilde{\boldsymbol{\omega}}_k - \hat{\mathbf{b}}_k^+$.

The error-quaternions at time t_{k+1} relative to $\hat{\mathbf{q}}_{k+1}^-$ are then computed using

$$\boldsymbol{\delta}\mathbf{q}_{k+1}^{(i)} \equiv \begin{bmatrix} \boldsymbol{\delta\varrho}_{k+1}^{(i)} \\ \delta q_{4_{k+1}}^{(i)} \end{bmatrix} = \mathbf{q}_{k+1}^{(i)} \otimes (\hat{\mathbf{q}}_{k+1}^-)^{-1} \tag{23}$$

The propagated error-MRPs are given by

$$\boldsymbol{\delta}\mathbf{p}_{k+1}^{(i)} = f\frac{\text{sign}(\delta q_{4_{k+1}}^{(i)})\boldsymbol{\delta\varrho}_{k+1}^{(i)}}{1 + |\delta q_{4_{k+1}}^{(i)}|} \tag{24a}$$

where sign is used to maintain consistency of the error-MRPs from the previous time point.

The next step involves computing the weights in the PF. This is done by using the measurement observations. At each time step it is assumed that m vector observations are available. The measurement model is given by

$$\tilde{\mathbf{y}}_{k+1} = \left.\begin{bmatrix} \mathbf{b}_1 \\ \vdots \\ \mathbf{b}_m \end{bmatrix}\right|_{k+1} + \mathbf{v}_{k+1} \tag{25}$$

where $\mathbf{v}_{k+1}$ is a zero-mean Gaussian noise process with covariance R_{k+1}. The likelihood function at time t_{k+1} used to update the weights is given by

$$L_{k+1}\left(\mathbf{x}_{k+1}^{(i)}\right) = p_{k+1}(\tilde{\mathbf{y}}_{k+1}|\mathbf{x}_{k+1}^{(i)}) \propto \exp\left[-\frac{1}{2}(\tilde{\mathbf{y}}_{k+1} - \mathbf{y}_{k+1}^{(i)})^T R_{k+1}^{-1}(\tilde{\mathbf{y}}_{k+1} - \mathbf{y}_{k+1}^{(i)})\right] \tag{26}$$

where $\propto$ stands for "proportional to" and

$$\mathbf{y}_{k+1}^{(i)} = \left.\begin{bmatrix} A(\mathbf{q}^{(i)})\mathbf{r}_1 \\ \vdots \\ A(\mathbf{q}^{(i)})\mathbf{r}_m \end{bmatrix}\right|_{k+1} \tag{27}$$

Equation (4) is used to update the weights. Note that the inverse of R_{k+1} is required, which is singular for unit vector observations. See Refs. 12 and 13 for a discussion on this issue in attitude filtering problems and an approach to overcome it.

The next step involves computing the updated quaternion and gyro biases. Equation (9a) is used to update the state estimate $\hat{\mathbf{x}}^+_{k+1} \equiv [\boldsymbol{\delta}\hat{\mathbf{p}}^{+T}_{k+1} \;\; \hat{\mathbf{b}}^{+T}_{k+1}]^T$. The updated gyro bias is simply given by the last three elements of $\hat{\mathbf{x}}^+_{k+1}$. The updated error-quaternion is computed using

$$\delta\hat{q}^+_{4_{k+1}} = \frac{f^2 - ||\boldsymbol{\delta}\hat{\mathbf{p}}^+_{k+1}||^2}{f^2 + ||\boldsymbol{\delta}\hat{\mathbf{p}}^+_{k+1}||^2} \tag{28a}$$

$$\boldsymbol{\delta}\hat{\mathbf{q}}^+_{k+1} = \begin{bmatrix} (1 + \delta\hat{q}^+_{4_{k+1}})\boldsymbol{\delta}\hat{\mathbf{p}}^+_{k+1}/f \\ \delta\hat{q}^+_{4_{k+1}} \end{bmatrix} \tag{28b}$$

and the updated quaternion is given by

$$\hat{\mathbf{q}}^+_{k+1} = \boldsymbol{\delta}\hat{\mathbf{q}}^+_{k+1} \otimes \hat{\mathbf{q}}^-_{k+1} \tag{29}$$

It is important to note that quaternion normalization is maintained throughout the entire process without any further steps to do so.

A complete cycle of the attitude estimation PF is summarized as follows:

- Prediction
  1. Propagate $\mathbf{b}^{(i)}_k \rightarrow \mathbf{b}^{(i)}_{k+1}$.
  2. Compute $\boldsymbol{\omega}^{(i)}_k$ and $\hat{\boldsymbol{\omega}}_k$.
  3. Propagate $\hat{\mathbf{q}}^+_k \rightarrow \hat{\mathbf{q}}^-_{k+1}$ through quaternion kinematics.
  4. Propagate $\mathbf{q}^{(i)}_k \rightarrow \mathbf{q}^{(i)}_{k+1}$ through quaternion kinematics.
- Update
  1. Compute the likelihood functions $L_{k+1}(\mathbf{x}^{(i)}_{k+1})$.
  2. Update the particle weights $w^{(i)}_{k+1}$ with the likelihood functions.
  3. Compute $\boldsymbol{\delta}\mathbf{q}^{(i)}_{k+1}$ as the difference between $\mathbf{q}^{(i)}_{k+1}$ and $\hat{\mathbf{q}}^-_{k+1}$.
  4. Convert $\boldsymbol{\delta}\mathbf{q}^{(i)}_{k+1} \rightarrow \boldsymbol{\delta}\mathbf{p}^{(i)}_{k+1}$.
  5. Compute the mean MRPs $\boldsymbol{\delta}\hat{\mathbf{p}}^+_{k+1}$ and the mean gyro bias $\hat{\mathbf{b}}^+_{k+1}$.
  6. Compute $\hat{\mathbf{q}}^+_{k+1}$ by "adding" $\boldsymbol{\delta}\hat{\mathbf{p}}^+_{k+1}$ to $\hat{\mathbf{q}}^-_{k+1}$.
  7. Compute the matrix Σ_{k+1}.
  8. Resample to obtain N copies of $\boldsymbol{\delta}\mathbf{p}^{(i)}_{k+1}$ and $\mathbf{b}^{(i)}_{k+1}$ and set $w^{(i)}_{k+1}$ to $1/N$.
  9. Perturb $\boldsymbol{\delta}\mathbf{p}^{(i)}_{k+1}$ and $\mathbf{b}^{(i)}_{k+1}$ by adding to them zero-mean Gaussian noise of covariance $h^2\Sigma_{k+1}$.
  10. Compute $\mathbf{q}^{(i)}_{k+1}$ by "adding" the perturbed $\boldsymbol{\delta}\mathbf{p}^{(i)}_{k+1}$ to $\hat{\mathbf{q}}^-_{k+1}$.

The above PF does not work well when the number N of particles is not large enough. With large initial errors, small process noise (gyro measurement noise), and a narrow likelihood function (the effective measurement noise of attitude sensors onboard most spacecraft is below one degree with that of the star tracker in the order of arcseconds), no initial attitude particles are likely to be in the significant regions of the likelihood functions. In most cases, only one or very few particles have large weights initially. The resampling step that follows discards most particles, the PF immediately loses particle diversity, and the particle cloud collapses into those few particles with large weights.

To overcome this problem, the principle of progressive correction splits the original likelihood function into n likelihood functions:

$$L_{k+1}(\mathbf{x}_{k+1}) = \prod_{j=1}^{n} L^{1/\lambda^{(j)}_{k+1}}_{k+1}(\mathbf{x}_{k+1}) \tag{30}$$

where $\lambda_{k+1}^{(j)} > 1$ and $\sum_{j=1}^{n} 1/\lambda_{k+1}^{(j)} = 1$. Given the measurement vector $\tilde{\mathbf{y}}_{k+1}$, the particles are updated with n identical fictitious measurements $\tilde{\mathbf{y}}_{k+1}$ at t_{k+1}, for which the likelihood functions are $L_{k+1}^{\lambda_{k+1}^{(j)}}(\mathbf{x}_{k+1})$. The updates are implemented by the repeated use of the aforementioned ten steps. New quaternion reference $\hat{\mathbf{q}}_{k+1}^{-}$ is needed for each update, and therefore $\hat{\mathbf{q}}_{k+1}^{-}$ is set to $\hat{\mathbf{q}}_{k+1}^{+}$ at the end of an update, following Step 10. Note that Step 9 is the only step at update that provides new particles, some of which are likely to agree with the measurement better than the particles directly propagated through the dynamical model. This amounts to a local random search of the state space for better particles.

For the likelihood function given by Eq. (26), $L_{k+1}^{1/\lambda_{k+1}^{(j)}}(\mathbf{x}_{k+1})$ is equal to $L_{k+1}(\mathbf{x}_{k+1})$ where R_{k+1} is replaced by $\lambda_{k+1}^{(j)} \cdot R_{k+1}$. Greater-than-one $\lambda_{k+1}^{(j)}$ increases the spread of the likelihood function and reduces the variance of the corresponding particle weights, which reduces the number of particles eliminated by the resampling step. A decreasing sequence in general, they are computed from the particles as

$$\lambda_{k+1}^{(j)} = \max\left(1, \frac{\max_i L_{k+1}(\mathbf{x}_{k+1}^{(i)})}{\log(\delta_{\max})}\right) \tag{31}$$

where $\delta_{\max}$ is a tuning parameter. Small $\delta_{\max}$ results in large $\lambda_{k+1}^{(j)}$.

Modification of the progressive correction method is made when applied to spacecraft attitude estimation with extremely poor initial guesses. It is observed that using large $\lambda_{k+1}^{(j)}$ in the initialization phase is essential to the convergence of the PF from extremely large initial errors. Measures are taken to ensure that $\lambda_{k+1}^{(i)}$ does not go too small while the attitude errors are large. The number of progressive corrections at each measurement time is limited to two because a large number of progressive corrections cause significant bias errors[5] and more importantly may eliminate the too many particles too quickly. After $\lambda_{k+1}^{(1)}$ is computed using Eq. (31), it is increased to the nearest member of a predetermined set. For example, if $\lambda_{k+1}^{(1)}$ is between 10 and 100, it is replaced by 100; if $\lambda_{k+1}^{(1)}$ is between 100 and 1000, it is replaced by 1000; and so on. In addition, $\lambda_{k+1}^{(2)}$ is chosen as the larger of $0.5\lambda_{k+1}^{(1)}$ and the value computed using Eq. (31). Note that $\lambda_{k+1}^{(j)}$ computed this way does not necessarily satisfy $\sum_{j=1}^{n} 1/\lambda_{k+1}^{(j)} = 1$. When that is the case, i.e., $\sum_{j=1}^{n} 1/\lambda_{k+1}^{(j)} < 1$, the information contained in the measurements is not made full use of in the Bayesian sense.

V. Simulation Example

In this section a simulation is shown that compares the performance of the PF to the UKF and EKF. The values of the main parameters of the PF are: the number of particles $N = 2000$, the perturbation level $h = 0.1$, the number of progressive corrections $n = 2$, and the progressive correction parameter $\delta_{\max} = e^6 \approx 403$. The PF is implemented in MATLAB.

The simulation is taken directly from Ref. 2, which involves an Earth-pointing spacecraft in a near-circular 90 min (350 km) orbit with an inclination of 35°. The nominal Earth-pointing mission mode requires a rotation once per orbit about the spacecraft's y-axis. The simulation shown here assumes only three-axis magnetometer (TAM) and gyro measurements. The magnetic field reference is modeled using a 10^{th}-order International Geomagnetic Reference Field model.[14] TAM sensor noise is modeled by a zero-mean Gaussian white-noise process with a standard deviation of 30 nT. The gyro measurements are simulated using $\sigma_u = 3.1623 \times 10^{-4}$ μrad/sec$^{3/2}$ and $\sigma_v = 0.31623$ μrad/sec$^{1/2}$, and an initial bias of 0.1 deg/hr on each axis.

Attitude errors of $-50°$, $50°$ and $160°$ for each axis, respectively, are added into the initial-condition attitude estimate. The initial attitude covariance is set to $(50 \text{ deg})^2$ for each attitude component. The initial x- and z-axes estimates for the gyro biases are set to zero, however the initial y-axis bias estimate is 20 deg/hr. The initial bias covariance is set to $(20 \text{ deg/hr})^2$ for each axis. The initial particles are drawn using a Gaussian distribution with the aforemention covariance matrices, which ensures that each filter is initialized in a consistent manner.

A plot of the norm of the attitude errors for this simulation case is shown in Figure 1. The EKF does not converge for this case since the first-order approximation cannot adequately capture the large initial condition errors. The UKF does have better convergence properties than the EKF, however the PF provides the best convergence performance. Both the EKF and the UKF are more sensitive to the initial conditions than the PF. Because the PF makes extensive use of pseudo-random numbers, its estimates vary from run

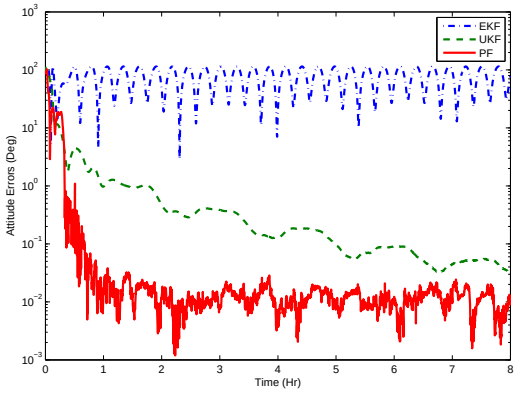

Figure 1. Norm of Attitude Errors

to run even if the same measurements are processed. The performance of the PF is checked by running the filter fifty times. It always converges and the statistics of the estimation results in the fifty runs are almost identical.

When little information about the initial attitude is available to the filter, as is the case with this example, it is preferable to use the uniform attitude distribution[15] to generate the initial attitude particles. The error result of the PF with the initial attitude particles draw from the uniform attitude distribution is almost identical to what is shown in the figure and is therefore not plotted.

The MATLAB implementation of the PF is a combination of built-in functions, user-written MEX files, and M-files. By default, the M-files are accelerated by the MATLAB JIT (Just-In-Time) accelerator. The MEX files implement four computationally intensive for-loops, such as the conversion between the quaternion particles and the MRP particles. No MEX files are used for the EKF or the UKF. The particular MATLAB implementation of the PF is several tens times computationally expensive than that of the UKF or the EKF. On a Thinkpad T61 computer (T9300/2.5 GHz; 2 GB RAM; Windows Vista Basic, 32-bit), it takes the PF about 20 seconds to process the 2881 TAM measurements, one per filter cycle. Hence, the execution time for one PF cycle is about 7.0 milliseconds on this computer. It is expected that a C implementation of the PF is faster than this MATLAB implementation.

VI. Conclusions

In this paper a new formation for attitude estimation using a particle filter was shown. Straightforward application of a particle filter with the quaternion requires a normalization procedure which can be computationally expensive. The approach developed in this paper used an unconstrained three-dimensional vector for the local-error representation while using quaternion for the global representation. In this approach the unconstrained local-error particles can be used directly in the particle filter. Also, the conversion from the local-error representation to the global representation requires less computations than implementing a strategy that explicitly maintains quaternion normalization. Although the particle filter shown in this paper is based on the bootstrap filter, the local/global representation can be still applied to any particle filter formulation. Simulation results indicated that the performance of the new filter based on the idea of progressive correction far exceeds the standard extended Kalman and unscented Kalman filter for large initialization errors.

References

[1]Lefferts, E. J., Markley, F. L., and Shuster, M. D., "Kalman Filtering for Spacecraft Attitude Estimation," *Journal of Guidance, Control, and Dynamics*, Vol. 5, No. 5, Sept.-Oct. 1982, pp. 417–429.

[2]Crassidis, J. L. and Markley, F. L., "Unscented Filtering for Spacecraft Attitude Estimation," *Journal of Guidance, Control and Dynamics*, Vol. 26, No. 4, July-Aug. 2003, pp. 536–542.

[3]Crassidis, J. L., Markley, F. L., and Cheng, Y., "A Survey of Nonlinear Attitude Estimation Methods," *Journal of*

Guidance, Control and Dynamics, Vol. 30, No. 1, Jan.-Feb. 2007, pp. 12–28.

[4]Arulampalam, M. S., Maskell, S., Gordon, N., and Clapp, T., "A Tutorial on Particle Filters for Online Nonlinear/Non-Gaussian Bayesian Tracking," *IEEE Transactions on Signal Processing*, Vol. 50, No. 2, Feb. 2002, pp. 174–185.

[5]Musso, C., Oudjane, N., and Gland, F. L., "Improving Regularized Particle Filters," *Sequential Monte Carlo Methods in Practice*, edited by A. Doucet, N. de Freitas, and N. Gordon, chap. 12, Springer-Verlag, New York, NY, 2001.

[6]Oshman, Y. and Carmi, A., "Attitude Estimation from Vector Observations Using a Genetic-Algorithm-Embedded Quaternion Particle Filter," *Journal of Guidance, Control and Dynamics*, Vol. 29, No. 4, July-Aug. 2006, pp. 879–891.

[7]Shuster, M. D., "A Survey of Attitude Representations," *Journal of the Astronautical Sciences*, Vol. 41, No. 4, Oct.-Dec. 1993, pp. 439–517.

[8]Markley, F. L., Cheng, Y., Crassidis, J. L., and Oshman, Y., "Averaging Quaternions," *Journal of Guidance, Control and Dynamics*, Vol. 30, No. 4, July-Aug. 2007, pp. 1193–1196.

[9]Gordon, N. J., Salmond, D. J., and Smith, A. F. M., "Novel Approach to Nonlinear/Non-Gaussian Bayesian State Estimation," *Radar and Signal Processing, IEE Proceedings F*, Vol. 140, No. 2, April 1993, pp. 107–113.

[10]Doucet, A., de Freitas, N., and Gordon, N., "An Introduction to Sequential Monte Carlo Methods," *Sequential Monte Carlo Methods in Practice*, edited by A. Doucet, N. de Freitas, and N. Gordon, chap. 1, Springer-Verlag, New York, NY, 2001.

[11]Farrenkopf, R. L., "Analytic Steady-State Accuracy Solutions for Two Common Spacecraft Attitude Estimators," *Journal of Guidance and Control*, Vol. 1, No. 4, July-Aug. 1978, pp. 282–284.

[12]Shuster, M. D., "Kalman Filtering of Spacecraft Attitude and the QUEST Model," *Journal of the Astronautical Sciences*, Vol. 38, No. 3, July-Sept. 1990, pp. 377–393.

[13]Cheng, Y., Crassidis, J. L., and Markley, F. L., "Attitude Estimation for Large Field-of-View Sensors," *The Journal of the Astronautical Sciences*, Vol. 54, No. 3/4, July-Dec. 2006, pp. 433–448.

[14]Langel, R. A., "International Geomagnetic Reference Field: The Sixth Generation," *Journal of Geomagnetism and Geoelectricity*, Vol. 44, No. 9, 1992, pp. 679–707.

[15]Shuster, M. D., "Uniform Attitude Probability Distributions," *Journal of the Astronautical Sciences*, Vol. 51, No. 4, Oct.-Dec. 2003, pp. 451–475.