

Robust Spacecraft Angular Rate Estimation from Vector Observations Using Interlaced Particle Filtering

Avishy Carmi* and Yaakov Oshman†

Technion—Israel Institute of Technology, 32000 Haifa, Israel

DOI: 10.2514/1.28932

A novel algorithm is presented for the estimation of the spacecraft angular rate from vector observations. Belonging to the class of Monte Carlo sequential methods, the new estimator is a particle filter that uses approximate numerical representation techniques for performing the otherwise exact time propagation and measurement update of potentially non-Gaussian probability density functions in inherently nonlinear systems. This paper develops the filter and its implementation in the case of a low Earth orbit spacecraft, acquiring noisy geomagnetic field measurements via a three-axis magnetometer. Because the effective measurement noise in this case is time correlated, a special procedure is developed to account for that correlation in the particle filter implementation. The new estimator copes with the absence of an exact inertia tensor by employing a secondary particle filter that computes a maximum-likelihood estimate of the tensor of inertia, thus avoiding the need to expand the primary filter's state. This renders the new estimator highly efficient and enables its implementation with a remarkably small number of particles. The results of a simulation study are presented, in which the new filter is compared to a recently presented conventional extended Kalman filter. The comparison demonstrates the viability and robustness of the new algorithm and its fast convergence rate.

I. Introduction

SPACECRAFT attitude control systems and attitude estimation algorithms must use angular rate information for tasks such as detumbling, nutation damping, momentum management, and attitude propagation. A widely used angular rate sensor onboard spacecraft is the rate gyroscope triad, whose purpose is to provide three-axis rate information.

Much experience has shown that rate gyros are failure prone. They tend to saturate during high angular rate scenarios such as tumbling and initial attitude acquisition. Moreover, gyros may not be suitable for low-cost satellites due to price, power consumption, and weight and volume considerations. This leads to the requirement of reliable, gyroless rate estimation schemes that can provide backup capabilities for spacecraft that use rate gyros, and affordable rate solutions for low-cost gyroless satellites.

The most common source of information for deriving attitude and attitude-rate estimates are so-called vector measurements. These can be obtained from a star tracker, sun sensor, Earth sensor, and three-axis magnetometer (TAM). Whereas high-accuracy star trackers are extremely expensive and sun sensors are useless during sun eclipse [for low Earth orbit satellites], the TAM is an integrated part of any spacecraft, and its readings are available at any time.

Several methods have been proposed in the past for gyroless angular rate estimation from vector measurements. Modeling the angular rate as a stochastic process and considering the nonlinearities involved, most of the methods use the framework of extended Kalman filtering. Psiaki et al. [1] present an extended Kalman filter

(EKF) for attitude and attitude-rate estimation from TAM readings. Based upon perfectly known spacecraft ephemeris, this algorithm converges in about an orbit and yields coarse estimates. Challa et al. [2] use a slow deterministic algorithm to obtain a coarse angular rate estimate that is later fed into an EKF-based orientation and angular rate estimator. This algorithm uses temporal derivatives of Earth's magnetic field and is limited to steady-state, operational conditions and when the angle between the satellite momentum and Earth's magnetic field is larger than 15 deg. Azor et al. [3] present an extended interlaced Kalman filter that uses vector measurements and their derivatives to estimate the angular rate. In related work, Harman and Bar-Itzhack [4] present a pseudolinear Kalman (PSELIKA) filter and a continuous-discrete algebraic Riccati equation (SDARE) filter that also use the derivatives of vector measurements to estimate the angular rate components. Azor et al. [5] use the PSELIKA and the SDARE filtering schemes to estimate the angular rate based on quaternion measurements or their temporal derivatives. The algorithms presented in [3–5] assume the knowledge of spacecraft attitude. In [6] a PSELIKA filter has been presented for estimating both attitude and angular rate, but this filter has been shown to be restricted with regard to its measurement requirements and update rate. Crassidis and Markley [7] propose a predictive filter for gyroless attitude estimation that is not based on the Kalman filtering methodology. Based on the minimum model error estimator, and assuming that the spacecraft tensor of inertia is perfectly known, their algorithm estimates the spacecraft total momentum (along with the quaternion of rotation), thereby providing an estimate of the angular rate.

A rather different class of algorithms has been recently introduced in [8–10]. In this approach, the angular rate is estimated independently of any attitude or orbital information. This class of estimators relies on the underlying assumption that the inertial vector measurement source (i.e., the sun direction vector or the magnetic field vector) is nearly constant in inertial frame between two successive measurements. Thus, the EKF algorithm presented in [8] uses a coarse batch estimator for initialization and executes numerical differentiation of the vector measurements, which limits its attainable accuracy. Improved versions of the estimators of [8] are presented in [9], where the EKF is formulated to account, in addition to the angular rate components, for corrections in five of the inertia tensor elements. However, the deterministic batch estimator of [9] shares the same disadvantage of the EKF of [8] because it uses differentiation of the magnetic field vector. The EKF proposed in [10] uses analytic propagation to achieve better accuracy and to

Presented as Paper 6399 at the AIAA Guidance, Navigation, and Control Conference and Exhibit, San Francisco, California, 15–18 August 2005; received 22 November 2006; revision received 19 February 2007; accepted for publication 19 February 2007. Copyright © 2007 by Avishy Carmi and Yaakov Oshman. Published by the American Institute of Aeronautics and Astronautics, Inc., with permission. Copies of this paper may be made for personal or internal use, on condition that the copier pay the \$10.00 per-copy fee to the Copyright Clearance Center, Inc., 222 Rosewood Drive, Danvers, MA 01923; include the code 0731-5090/07 \$10.00 in correspondence with the CCC.

*Doctoral Student, Department of Aerospace Engineering; avishy@aerodyne.technion.ac.il. Member AIAA.

†Professor, Department of Aerospace Engineering, Holder of the Louis and Helen Rogow Chair in Aeronautical Engineering, Member, Technion's Asher Space Research Institute; yaakov.oshman@technion.ac.il. Associate Fellow AIAA.

reduce the computational burden. However, whereas this estimator exhibits fast convergence when the spacecraft inertia tensor is perfectly known, its performance degrades when the inertia tensor is unknown and has to be estimated.

Based upon the underlying assumption of [8–10], and motivated by [11], this paper proposes to sequentially estimate the angular rate from vector observations only, using a particle filter (PF). PFs, also known as sequential Monte Carlo (SMC) methods, refer to a set of algorithms implementing a recursive Bayesian model using simulation-based methods [12]. Avoiding the underlying assumptions of the Kalman filter, namely, that the state space is linear and Gaussian, these rather general and flexible methods enable solving for the posterior probability distributions of the unknown variables (upon which all inference on these variables is based) within a Bayesian framework, exploiting the recent dramatic increase in computing power. It should be emphasized that PFs are not just smart implementations of the Kalman filter or its nonlinear variants/extensions; rather, they are entirely different algorithms that lead to entirely different solutions to the nonlinear, non-Gaussian filtering problem. Contrary to Kalman filter extensions, the solutions obtained using PF algorithms are approximations to the optimal (in the Bayesian sense) solutions, which can be made arbitrarily close to the exact solutions by increasing the number of particles involved in the computation, thereby also increasing the computational workload.

Estimating the angular rate using a PF enjoys two major advantages relative to existing methods. First, the PF is easy to implement, and (when implemented with a sufficiently large number of particles) is insensitive to the initial conditions and to the nonlinearities involved. Second, the PF is not constrained with regard to the noise distributions, that is, it can work with any (not necessarily Gaussian) noise distribution associated with the particular sensors involved. In contradistinction, the KF assumes Gaussian distributions of the driving noise processes, an assumption that not always holds. On the other hand, implementing the PF with a large number of particles (to achieve high accuracy) can lead to an impractical computational burden.

This paper presents a novel efficient PF algorithm that alleviates the workload problem via two special measures. First, a special initialization procedure is used, to compensate for the small number of particles. Based upon the first vector observation, an initial set of samples is generated that represents a small, but highly likely, region of state space. This ensures fast convergence with a relatively small number of particles and renders the PF suitable for real-time implementation. Second, greater robustness and higher computational efficiency are achieved by estimating the spacecraft entire inertia tensor using a second, static PF, which works with the main PF in an interlaced manner.

A major contribution of this paper is related to the fact that the effective measurement noise for the mathematical model under investigation turns out to be time correlated. To enable the PF to take into account that time correlation, an approximate likelihood computation procedure is developed herein. This procedure extends the standard procedure used when the observations are conditionally independent given the state process.

The remainder of this paper is organized as follows. The next section presents a brief introduction to SMC methods. Next, the mathematical model of the angular rate estimation problem is outlined. Section IV provides a detailed development of the angular rate PF for the case where the inertia tensor is exactly known. A special approximate likelihood computation procedure is developed to accommodate the effective measurement noise time correlation. The extension of the PF algorithm to account for inertia tensor uncertainties via the use of a secondary, static PF then follows in the next section. Section VI presents the results of an extensive simulation study that was carried out to assess the performance of the new algorithm and compare it with the EKF estimator of [10]. Concluding remarks are offered in the last section. As a matter of notational convention, lower case and upper case letters are used to denote random variables and their realizations, respectively.

II. Particle Filtering

Particle filters are numerical simulation-based methods aimed at sequentially approximating the optimal filtering distribution. Based upon the strong law of large numbers, these methods implement sampling techniques to obtain a finite number of samples (called particles). The samples are then manipulated via a Bayesian recursion yielding a two-staged filtering methodology, comprising a propagation stage and a measurement update stage.

PFs draw much attention due to their ability to maintain an approximation of the entire posterior filtering distribution. Along with their rather simple simulation-based mechanization, this makes them highly preferable for highly nonlinear and non-Gaussian filtering applications.

Many researchers regard PFs as computationally intensive methods that are mainly acceptable for postprocessing computations. However, smart implementation, along with proper exploitation of the problem's unique characteristics, can render these methods fast and amenable for real-time applications.

A. Bayesian Approach to Filtering

Let the unobserved process $\{x_k, k \in \mathbb{N}\}$ be an \mathbb{R}^n -valued Markov process with a given initial probability density function p_{x_0} that evolves according to a transition kernel $p_{x_k|x_{k-1}}$. The observation process $\{y_k, k \in \mathbb{N}\}$ is an \mathbb{R}^p -valued stochastic process. Given x_k , the observation process is a conditionally independent sequence, possessing the conditional probability density function (pdf) $p_{y_k|x_k}$. Let $\mathcal{X}^k \triangleq \{x_0, \dots, x_k\}$ and $\mathcal{Y}^k \triangleq \{y_1, \dots, y_k\}$ be the process and observation time histories up to time k , respectively, and let $X^k \triangleq \{X_0, \dots, X_k\}$ and $Y^k \triangleq \{Y_1, \dots, Y_k\}$ be the realizations of \mathcal{X}^k and \mathcal{Y}^k , respectively.

In filtering problems one is commonly interested in estimating the marginal pdf $p_{x_k|\mathcal{Y}^k}$ (filtering density) sequentially in time. Adopting the Bayesian approach to filtering, this density is obtained, using a two-step recursion, as

$$p_{x_k|\mathcal{Y}^{k-1}}(X_k | Y^{k-1}) = \int_{-\infty}^{+\infty} p_{x_k|x_{k-1}}(X_k | X_{k-1}) p_{x_{k-1}|\mathcal{Y}^{k-1}}(X_{k-1} | Y^{k-1}) dX_{k-1} \quad (1a)$$

$$p_{x_k|\mathcal{Y}^k}(X_k | Y^k) = \frac{p_{y_k|x_k}(Y_k | X_k)}{\int_{-\infty}^{+\infty} p_{y_k|x_k}(Y_k | X_k) p_{x_k|\mathcal{Y}^{k-1}}(X_k | Y^{k-1}) dX_k} p_{x_k|\mathcal{Y}^{k-1}}(X_k | Y^{k-1}) \quad (1b)$$

In most cases one cannot obtain the normalizing density $p_{y_k|\mathcal{Y}^{k-1}}$ and the marginals of the posterior density, $p_{\mathcal{X}^k|\mathcal{Y}^k}$. Thus, these expressions can rarely be used in a straightforward implementation. Instead, approximations should be used, using alternative methods.

B. Particle Approximation

The PF mechanization approximates Eqs. (1) using a finite number of samples. To understand the rationale behind this method, assume that N independent random samples (called "particles"), denoted by $\{X_k(i)\}_{i=1}^N$ are sampled from the posterior distribution. Then, it follows directly from the strong law of large numbers that, for any function f that is integrable with respect to $p_{\mathcal{X}^k|\mathcal{Y}^k}$, [13]

$$\frac{1}{N} \sum_{i=1}^N f(X_k(i)) \rightarrow E[f(\mathcal{X}^k) | Y^k] \quad (2)$$

where (here and in the sequel) the symbol \rightarrow stands for almost sure convergence in N . Equation (2) means that the continuous posterior pdf $p_{\mathcal{X}^k|\mathcal{Y}^k}$ can be effectively approximated by its particles, and that the level of accuracy of this approximation is determined by the number of particles used.

Unfortunately, sampling directly from the posterior distribution is typically infeasible. For this reason the concept of importance sampling is an integral part of any practical PF [14]. When using importance sampling, the samples are drawn from a so-called importance distribution. The importance distribution can be chosen arbitrarily; the only constraint it should satisfy is that its support must include the support of the posterior distribution. Nevertheless, because the choice of importance distribution greatly affects the behavior of the PF, it constitutes a major consideration during PF design.

A simple choice of importance density is the prior density $p_{\mathcal{X}^k}$. Adopting the prior as the importance density, an approximation of the expectation in Eq. (2) is obtained as

$$\frac{1}{N} \sum_{i=1}^N w_k(i) f(X^k(i))$$

where

$$w_k(i) \triangleq \frac{p_{\mathcal{X}^k | \mathcal{Y}^k}(X^k(i) | Y^k)}{p_{\mathcal{X}^k}(X^k(i))} \quad (3)$$

is the importance weight of the i th sample. Let

$$\tilde{w}_k(i) \triangleq \frac{w_k(i)}{\sum_{j=1}^N w_k(j)} \quad (4)$$

be the normalized importance weight of the i th particle, then it can be easily verified that

$$\sum_{i=1}^N \tilde{w}_k(i) f(X^k(i)) \rightarrow E[f(\mathcal{X}^k) | Y^k] \quad (5)$$

The PF algorithm exploits the recursive structure of Eqs. (1) to compute the importance weights sequentially in time. Indeed, Eq. (3) yields

$$\begin{aligned} w_k(i) &\propto \frac{p_{y_k | x_k}(Y_k | X_k(i)) p_{x_k | x_{k-1}}(X_k(i) | X_{k-1}(i)) p_{\mathcal{X}^{k-1} | \mathcal{Y}^{k-1}}(X^{k-1}(i) | Y^{k-1})}{p_{x_k | x_{k-1}}(X_k(i) | X_{k-1}(i)) p_{\mathcal{X}^{k-1}}(X^{k-1}(i))} \\ &= p_{y_k | x_k}(Y_k | X_k(i)) w_{k-1}(i) \end{aligned} \quad (6)$$

C. Particle Degeneracy and Resampling

Practical implementation of the sequential importance sampling method, consisting of iterating Eq. (6), inevitably results in zero weights for all but, usually, one particle, after just a few iterations. This phenomenon is known as particle degeneracy in the PF literature [14]. Particle degeneracy occurs due to the use of a finite number of particles, which consequently allows only a partial representation of the sample space. A solution to this problem was introduced a decade ago as an ad hoc procedure known as resampling.

Resampling consists of discarding state trajectories whose contributions to the final estimate are small and multiplying trajectories whose contributions are expected to be significant. This means regeneration of particles with large importance weights and eliminating those with small importance weights. The resampling procedure decreases the particle degeneracy algorithmically, but introduces some practical problems. During the resampling procedure, more likely particles are multiplied, so that the particle cloud is concentrated in regions of interest of the state space. This produces a new particle system in which several particles have the same location. Moreover, if the dynamic noise is small, the particle system ultimately concentrates in a single point in state space. This loss of diversity eventually prevents the filter from correctly representing the posterior. One way of maintaining the particles' diversity is by injecting artificial process noise into the system. This

technique is known as regularization, or roughening (see [12], p. 247).

III. Mathematical Model

This section presents the angular rate process and observation models. The process mathematical model is based on Euler's equation. The observation model relates the body-referenced vector measurements to the sought-for angular rate process. It turns out that the observation model's effective measurement noise is time correlated. This correlation is carefully handled in the next section.

A. Angular Rate Process Model

Let $\{\omega_k\}_{k=0}^{\infty}$, $\omega \in \mathbb{R}^3$ be the angular velocity (angular rate) process of some body-frame Cartesian coordinate system \mathcal{B} with respect to some inertial reference coordinate system \mathcal{R} , resolved in \mathcal{B} . Representing the angular rate of a spacecraft, this process is the discrete-time equivalent of a continuous-time stochastic process satisfying Euler's equation. Using the common white noise engineering notation, this equation is written as

$$\dot{\omega}(t) = J^{-1}[-\omega(t) \times J\omega(t)] + \xi(t), \quad t \in [t_k, t_{k+1}) \quad (7)$$

where J denotes the spacecraft tensor of inertia. Given some initial distribution $\omega_0 \sim p_{\omega_0}$, Eq. (7) is driven by the process noise $\xi(t)$, representing the combined effect of external disturbance torques such as the aerodynamic, gravity gradient, and residual magnetic dipole torques.

B. Observation Model

The discrete-time quaternion stochastic process satisfies the recurrence equation

$$\mathbf{q}_k = \Phi_k \mathbf{q}_{k-1}, \quad k = 1, 2, \dots \quad (8)$$

where the discrete-time process $\{\mathbf{q}_k\}_{k=0}^{\infty}$ denotes the quaternion of rotation from a given reference frame \mathcal{R} onto the body frame \mathcal{B} , with some initial pdf $\mathbf{q}_0 \sim p_{\mathbf{q}_0}$. The quaternion process takes its values on the unit 3-sphere \mathbb{S}^3 and is constructed from vector and scalar parts, respectively,

$$\mathbf{q}_k = [\mathbf{e}_k^T \quad q_{4k}]^T \quad (9)$$

The orthogonal transition matrix Φ_k is expressed using ω_k . Assuming that ω_k is constant during the sampling time interval Δt yields [15]

$$\Phi_k = \Phi(\omega_k) = \begin{bmatrix} \cos(\frac{1}{2}\|\omega_k\|\Delta t) I_{3 \times 3} - [\psi_k \times] & \psi_k \\ -\psi_k^T & \cos(\frac{1}{2}\|\omega_k\|\Delta t) \end{bmatrix} \quad (10)$$

where

$$\psi_k \triangleq \sin\left(\frac{1}{2}\|\omega_k\|\Delta t\right) \frac{\omega_k}{\|\omega_k\|} \quad (11)$$

Let

$$\delta \mathbf{q}_{k-1} \triangleq \mathbf{q}_k \otimes \mathbf{q}_{k-1}^{-1}, \quad k = 1, 2, \dots \quad (12)$$

be the rotation quaternion associated with the rotation of \mathbf{q}_{k-1} onto \mathbf{q}_k , where \otimes denotes quaternion multiplication [15]. Then, Eq. (8) yields

$$\delta \mathbf{q}_{k-1} = (\Phi_k \mathbf{q}_{k-1}) \otimes \mathbf{q}_{k-1}^{-1} = \Phi_k (\mathbf{q}_{k-1} \otimes \mathbf{q}_{k-1}^{-1}) = \Phi_k [\mathbf{0}_{1 \times 3} \quad 1]^T \quad (13)$$

and Eqs. (10) and (13) give

$$\delta \mathbf{e}_{k-1} = \psi_k, \quad \delta q_{4k-1} = \cos(\frac{1}{2}\|\omega_k\|\Delta t) \quad (14)$$

Equation (14) defines the transformation between ω_k and $\delta \mathbf{q}_{k-1}$. For

conciseness, the quaternion vector and scalar parts will not be designated separately in the ensuing, and thus the following notation will be used for the transformation in Eq. (14):

$$\delta \mathbf{q}_{k-1} = \mathcal{Q}(\boldsymbol{\omega}_k) \quad (15)$$

Now let \mathbf{r}_k and \mathbf{y}_k be a pair of corresponding vector measurements acquired at time k in the two Cartesian coordinate systems \mathcal{R} and \mathcal{B} , respectively. Let $A(\mathbf{q}_k)$ be the attitude matrix corresponding to the quaternion \mathbf{q}_k . In general, the reference vector \mathbf{r}_k can be assumed to be known exactly, whereas the body vector \mathbf{y}_k is measured. The relation between the body and reference frame vectors is given by

$$\mathbf{y}_k = A(\mathbf{q}_k)\mathbf{r}_k + \delta \mathbf{y}_k, \quad k = 0, 1, \dots \quad (16)$$

where $\{\delta \mathbf{y}_k\}_{k=0}^{\infty}$ is the zero-mean measurement noise process, whose pdf is known, $\delta \mathbf{y}_k \sim p_{\delta \mathbf{y}_k}$, with covariance R_k^y . Using the definition in Eq. (12), the vector observation at time k can be written as

$$\begin{aligned} \mathbf{y}_k &= A(\delta \mathbf{q}_{k-1} \otimes \mathbf{q}_{k-1})\mathbf{r}_k + \delta \mathbf{y}_k = A(\delta \mathbf{q}_{k-1})A(\mathbf{q}_{k-1})\mathbf{r}_k + \delta \mathbf{y}_k \\ &k = 1, 2, \dots \end{aligned} \quad (17)$$

Assuming, as in [8–10], that the reference frame vector \mathbf{r}_{k-1} does not change considerably during the time interval Δt , yielding $\mathbf{r}_{k-1} \approx \mathbf{r}_k$, Eqs. (15–17) give the following observation equation:

$$\mathbf{y}_k = A(\delta \mathbf{q}_{k-1})(\mathbf{y}_{k-1} - \delta \mathbf{y}_{k-1}) + \delta \mathbf{y}_k = A(\mathcal{Q}(\boldsymbol{\omega}_k))\mathbf{y}_{k-1} + \mathbf{v}_k \quad (18)$$

The process $\{\mathbf{v}_k\}_{k=1}^{\infty}$, where

$$\mathbf{v}_k \triangleq \delta \mathbf{y}_k - A(\delta \mathbf{q}_{k-1})\delta \mathbf{y}_{k-1} \quad (19)$$

is the effective observation noise process.

A slightly different formulation of the observation equation (18) is obtained by noticing, from Eq. (12), that

$$A(\mathbf{q}_k) = A(\mathcal{Q}(\boldsymbol{\omega}_k))A(\mathbf{q}_{k-1}) \quad (20)$$

thus identifying $A(\mathcal{Q}(\boldsymbol{\omega}_k))$ as the transition matrix corresponding to the solution of the attitude kinematics equation

$$\frac{d}{dt}A(t) = [-\boldsymbol{\omega} \times]A(t) \quad (21)$$

over the sampling interval Δt . Defining the matrix operator $\Gamma(\cdot)$ as

$$\Gamma(\boldsymbol{\omega}_k) \triangleq \exp[-[\boldsymbol{\omega}_k \times]\Delta t] \quad (22)$$

Equation (21) yields

$$A(\mathcal{Q}(\boldsymbol{\omega}_k)) = \Gamma(\boldsymbol{\omega}_k) \quad (23)$$

Using Eq. (23) in Eq. (18) gives

$$\mathbf{y}_k = \Gamma(\boldsymbol{\omega}_k)\mathbf{y}_{k-1} + \mathbf{v}_k(\boldsymbol{\omega}_k) \quad (24)$$

with

$$\mathbf{v}_k(\boldsymbol{\omega}_k) = \delta \mathbf{y}_k - \Gamma(\boldsymbol{\omega}_k)\delta \mathbf{y}_{k-1} \quad (25)$$

where it is noted that the effective observation noise depends on the angular rate random variable $\boldsymbol{\omega}_k$.

Clearly, the effective measurement noise process of the observation model in Eq. (24), $\{\mathbf{v}_k(\boldsymbol{\omega}_k)\}_{k=1}^{\infty}$, is time correlated (colored). The derivation of an angular rate PF for the specific case where the measurement noise is colored is detailed in the next section.

IV. Angular Rate Particle Filter

The angular rate particle filter is first derived in this section while assuming a perfectly known spacecraft inertia tensor. This

assumption will be relaxed in the next section. The new filter is termed in the sequel omega particle filter (OPF).

A. Likelihood Computation for Colored Measurement Noise

Because the effective measurement noise is colored, the assumption that the observations are conditionally independent given the state process, which underlies Eq. (1b), is violated. The classical solution to this problem consists of state augmentation [10]. However, implementation of a high-dimensional particle filter is computationally inefficient because of the large number of particles required for properly representing a high-dimensional pdf.

An efficient method for modeling colored noise measurements is proposed in this work. The method, which consists of approximating the likelihood of each particle, demands no major modifications in the PF mechanization.

Let \mathbf{Y}_k be a realization of a single measurement \mathbf{y}_k and let $\boldsymbol{\Omega}_k$ be a realization of $\boldsymbol{\omega}_k$. For convenience, redefine the observation time history up to time k and its realization as $\mathcal{Y}^k \triangleq \{\mathbf{y}_0, \dots, \mathbf{y}_k\}$ and $Y^k \triangleq \{\mathbf{Y}_0, \dots, \mathbf{Y}_k\}$, respectively. Because the observation process is colored, the Bayesian recursion [formulated in Eq. (1b) for the case of conditionally independent measurements] is now written as

$$\begin{aligned} p_{\boldsymbol{\omega}_k|\mathcal{Y}^k}(\boldsymbol{\Omega}_k | Y^k) &= \frac{p_{\mathbf{y}_k|\boldsymbol{\omega}_k, \mathcal{Y}^{k-1}}(\mathbf{Y}_k | \boldsymbol{\Omega}_k, Y^{k-1})}{\int_{-\infty}^{+\infty} p_{\mathbf{y}_k|\boldsymbol{\omega}_k, \mathcal{Y}^{k-1}}(\mathbf{Y}_k | \boldsymbol{\Omega}_k, Y^{k-1}) p_{\boldsymbol{\omega}_k|\mathcal{Y}^{k-1}}(\boldsymbol{\Omega}_k | Y^{k-1}) d\boldsymbol{\Omega}_k} \\ &\times p_{\boldsymbol{\omega}_k|\mathcal{Y}^{k-1}}(\boldsymbol{\Omega}_k | Y^{k-1}) \end{aligned} \quad (26)$$

Clearly, Eq. (26) differs from Eq. (1b) only in the likelihood term $p_{\mathbf{y}_k|\boldsymbol{\omega}_k, \mathcal{Y}^{k-1}}$, which is affected by the time correlation of the measurement process. An approximate computation of this likelihood is presented in the ensuing.

1. Markov Modeling of the Effective Measurement Noise

First, the effective measurement noise process $\{\mathbf{v}_k(\boldsymbol{\omega}_k)\}_{k=1}^{\infty}$ is approximated by a Gauss–Markov process with equivalent second-order statistics. The approximated noise is described via the following first-order Markovian model

$$\mathbf{v}_k(\boldsymbol{\omega}_k) = M_k \mathbf{v}_{k-1}(\boldsymbol{\omega}_{k-1}) + \mathbf{w}_k, \quad k = 2, 3, \dots \quad (27)$$

where M_k is a decorrelation matrix and $\{\mathbf{w}_k\}_{k=2}^{\infty}$ is a zero-mean, white Gaussian driving noise with covariance $E[\mathbf{w}_k \mathbf{w}_k^T] = R_k^w$. The initial statistics of Eq. (27) is set according to the definition of the observation noise in Eq. (25), thus $\mathbf{v}_1(\boldsymbol{\omega}_1)$ is Gaussian with mean and covariance given by

$$E[\mathbf{v}_1(\boldsymbol{\omega}_1) | \boldsymbol{\omega}_1 = \boldsymbol{\Omega}_1] = E[\mathbf{v}_1(\boldsymbol{\omega}_1) | \boldsymbol{\omega}_1 = \boldsymbol{\Omega}_1] = \mathbf{0} \quad (28a)$$

$$\begin{aligned} E[\mathbf{v}_1(\boldsymbol{\omega}_1)\mathbf{v}_1(\boldsymbol{\omega}_1)^T | \boldsymbol{\omega}_1 = \boldsymbol{\Omega}_1] &= E[\mathbf{v}_1(\boldsymbol{\omega}_1)\mathbf{v}_1(\boldsymbol{\omega}_1)^T | \boldsymbol{\omega}_1 = \boldsymbol{\Omega}_1] \\ &= R_1^y + \Gamma(\boldsymbol{\Omega}_1)R_0^y\Gamma(\boldsymbol{\Omega}_1)^T \end{aligned} \quad (28b)$$

The matrix M_k and the covariance R_k^w are next evaluated by comparing the second-order moments of $\{\mathbf{v}_k(\boldsymbol{\omega}_k)\}_{k=1}^{\infty}$ and $\{\mathbf{v}_k(\boldsymbol{\omega}_k)\}_{k=1}^{\infty}$ based on Eqs. (25) and (27), respectively.

Using Eq. (27), the conditional one-lag autocorrelation of the process $\{\mathbf{v}_k(\boldsymbol{\omega}_k)\}_{k=1}^{\infty}$ is

$$\begin{aligned} E[\mathbf{v}_k(\boldsymbol{\omega}_k)\mathbf{v}_{k-1}(\boldsymbol{\omega}_{k-1})^T | \boldsymbol{\omega}_k = \boldsymbol{\Omega}_k, \boldsymbol{\omega}_{k-1} = \boldsymbol{\Omega}_{k-1}] &= M_k E[\mathbf{v}_{k-1}(\boldsymbol{\omega}_{k-1})\mathbf{v}_{k-1}(\boldsymbol{\omega}_{k-1})^T | \boldsymbol{\omega}_{k-1} = \boldsymbol{\Omega}_{k-1}] \end{aligned} \quad (29)$$

A corresponding term is directly derived from Eq. (25) as

$$E[\mathbf{v}_k(\boldsymbol{\omega}_k)\mathbf{v}_{k-1}(\boldsymbol{\omega}_{k-1})^T \mid \boldsymbol{\omega}_k = \boldsymbol{\Omega}_k, \boldsymbol{\omega}_{k-1} = \boldsymbol{\Omega}_{k-1}] = -\Gamma(\boldsymbol{\Omega}_k)R_{k-1}^y \quad (30)$$

Now, using Eq. (25) to express the covariance of $\mathbf{v}_{k-1}(\boldsymbol{\omega}_{k-1}) \mid \boldsymbol{\omega}_{k-1}$ yields

$$E[\mathbf{v}_{k-1}(\boldsymbol{\omega}_{k-1})\mathbf{v}_{k-1}(\boldsymbol{\omega}_{k-1})^T \mid \boldsymbol{\omega}_{k-1} = \boldsymbol{\Omega}_{k-1}] = R_{k-1}^y + \Gamma(\boldsymbol{\Omega}_{k-1})R_{k-2}^y\Gamma(\boldsymbol{\Omega}_{k-1})^T \quad (31)$$

Setting

$$E[\mathbf{v}_{k-1}(\boldsymbol{\omega}_{k-1})\mathbf{v}_{k-1}(\boldsymbol{\omega}_{k-1})^T \mid \boldsymbol{\omega}_{k-1} = \boldsymbol{\Omega}_{k-1}] = E[\mathbf{v}_{k-1}(\boldsymbol{\omega}_{k-1})\mathbf{v}_{k-1}(\boldsymbol{\omega}_{k-1})^T \mid \boldsymbol{\omega}_{k-1} = \boldsymbol{\Omega}_{k-1}] \quad (32)$$

and using Eq. (31) in Eq. (29) yields

$$E[\mathbf{v}_k(\boldsymbol{\omega}_k)\mathbf{v}_{k-1}(\boldsymbol{\omega}_{k-1})^T \mid \boldsymbol{\omega}_k = \boldsymbol{\Omega}_k, \boldsymbol{\omega}_{k-1} = \boldsymbol{\Omega}_{k-1}] = M_k \left[R_{k-1}^y + \Gamma(\boldsymbol{\Omega}_{k-1})R_{k-2}^y\Gamma(\boldsymbol{\Omega}_{k-1})^T \right] \quad (33)$$

Similarly, setting

$$E[\mathbf{v}_k(\boldsymbol{\omega}_k)\mathbf{v}_{k-1}(\boldsymbol{\omega}_{k-1})^T \mid \boldsymbol{\omega}_k = \boldsymbol{\Omega}_k, \boldsymbol{\omega}_{k-1} = \boldsymbol{\Omega}_{k-1}] = E[\mathbf{v}_k(\boldsymbol{\omega}_k)\mathbf{v}_{k-1}(\boldsymbol{\omega}_{k-1})^T \mid \boldsymbol{\omega}_k = \boldsymbol{\Omega}_k, \boldsymbol{\omega}_{k-1} = \boldsymbol{\Omega}_{k-1}] \quad (34)$$

the decorrelation matrix M_k is derived from Eqs. (30) and (33) as

$$M_k = -\Gamma(\boldsymbol{\Omega}_k)R_{k-1}^y \left[R_{k-1}^y + \Gamma(\boldsymbol{\Omega}_{k-1})R_{k-2}^y\Gamma(\boldsymbol{\Omega}_{k-1})^T \right]^{-1} \quad (35)$$

$k = 2, 3, \dots$

The covariance R_k^w is obtained by noticing from Eq. (27) that

$$E[\mathbf{v}_k(\boldsymbol{\omega}_k)\mathbf{v}_k(\boldsymbol{\omega}_k)^T \mid \boldsymbol{\omega}_k = \boldsymbol{\Omega}_k, \boldsymbol{\omega}_{k-1} = \boldsymbol{\Omega}_{k-1}] = M_k E[\mathbf{v}_{k-1}(\boldsymbol{\omega}_{k-1})\mathbf{v}_{k-1}(\boldsymbol{\omega}_{k-1})^T \mid \boldsymbol{\omega}_k = \boldsymbol{\Omega}_k, \boldsymbol{\omega}_{k-1} = \boldsymbol{\Omega}_{k-1}] M_k^T + R_k^w = M_k E[\mathbf{v}_{k-1}(\boldsymbol{\omega}_{k-1})\mathbf{v}_{k-1}(\boldsymbol{\omega}_{k-1})^T \mid \boldsymbol{\omega}_{k-1} = \boldsymbol{\Omega}_{k-1}] M_k^T + R_k^w \quad (36)$$

Finally, replacing the covariances of $\mathbf{v}_k(\boldsymbol{\omega}_k) \mid \boldsymbol{\omega}_k$ and $\mathbf{v}_{k-1}(\boldsymbol{\omega}_{k-1}) \mid \boldsymbol{\omega}_{k-1}$ in Eq. (36) with their $\mathbf{v}_k(\boldsymbol{\omega}_k) \mid \boldsymbol{\omega}_k$ and $\mathbf{v}_{k-1}(\boldsymbol{\omega}_{k-1}) \mid \boldsymbol{\omega}_{k-1}$ equivalents yields

$$R_k^w = R_k^y + \Gamma(\boldsymbol{\Omega}_k)R_{k-1}^y\Gamma(\boldsymbol{\Omega}_k)^T - M_k \left[R_{k-1}^y + \Gamma(\boldsymbol{\Omega}_{k-1})R_{k-2}^y\Gamma(\boldsymbol{\Omega}_{k-1})^T \right] M_k^T \quad (37)$$

$k = 2, 3, \dots$

2. Likelihood Approximation

Using the Markov model for the effective measurement noise, the following theorem presents an approximation of the likelihood $p_{y_k|\boldsymbol{\omega}_k, \mathcal{Y}^{k-1}}$. For notational convenience, the following definition is used:

$$\boldsymbol{\mu}_k(\boldsymbol{\alpha}, \boldsymbol{\beta}) \triangleq \Gamma(\boldsymbol{\alpha})\mathbf{Y}_{k-1} + M_k\mathbf{Y}_{k-1} - M_k\Gamma(\boldsymbol{\beta})\mathbf{Y}_{k-2}, \quad k = 2, 3, \dots \quad (38)$$

Theorem 1. Assume that the angular rate process in Eq. (7) is driven by a low-intensity process noise, and that the effective measurement noise process $\{\mathbf{v}_k(\boldsymbol{\omega}_k)\}_{k=1}^\infty$ is adequately represented by the Gauss-Markov model in Eq. (27). Then, the likelihood $p_{y_k|\boldsymbol{\omega}_k, \mathcal{Y}^{k-1}}$ can be approximated as

$$p_{y_k|\boldsymbol{\omega}_k, \mathcal{Y}^{k-1}}(\mathbf{Y}_k \mid \boldsymbol{\Omega}_k, \mathbf{Y}^{k-1}) \approx \frac{1}{(2\pi)^{3/2} |R_k^w|^{1/2}} \times \exp \left\{ -\frac{1}{2} [\mathbf{Y}_k - \boldsymbol{\mu}_k(\boldsymbol{\Omega}_k, f^{-1}(\boldsymbol{\Omega}_k))]^T (R_k^w)^{-1} \times [\mathbf{Y}_k - \boldsymbol{\mu}_k(\boldsymbol{\Omega}_k, f^{-1}(\boldsymbol{\Omega}_k))] \right\} \triangleq \hat{p}_{y_k|\boldsymbol{\omega}_k, \mathcal{Y}^{k-1}}(\mathbf{Y}_k \mid \boldsymbol{\Omega}_k, \mathbf{Y}^{k-1}) \quad (39)$$

where $f(\boldsymbol{\Omega}_{k-1})$ is the solution at time t_k of the certainty-equivalent first-order differential equation corresponding to Eq. (7)

$$\dot{\boldsymbol{\Omega}}(t) = J^{-1}(-\boldsymbol{\Omega}(t) \times J\boldsymbol{\Omega}(t)) \quad (40)$$

with initial condition $\boldsymbol{\Omega}(t_{k-1}) = \boldsymbol{\Omega}_{k-1}$, f^{-1} is the inverse operator of f , and M_k is computed using Eq. (35) with $f^{-1}(\boldsymbol{\Omega}_k)$ replacing $\boldsymbol{\Omega}_{k-1}$.

Proof. First, observe that the likelihood $p_{y_k|\boldsymbol{\omega}_k, \mathcal{Y}^{k-1}}$ can be expressed as

$$p_{y_k|\boldsymbol{\omega}_k, \mathcal{Y}^{k-1}}(\mathbf{Y}_k \mid \boldsymbol{\Omega}_k, \mathbf{Y}^{k-1}) = \int_{-\infty}^{+\infty} p_{y_k|\boldsymbol{\omega}_k, \boldsymbol{\omega}_{k-1}, \mathcal{Y}^{k-1}}(\mathbf{Y}_k \mid \boldsymbol{\Omega}_k, \boldsymbol{\Omega}_{k-1}, \mathbf{Y}^{k-1}) \times p_{\boldsymbol{\omega}_{k-1}|\boldsymbol{\omega}_k, \mathcal{Y}^{k-1}}(\boldsymbol{\Omega}_{k-1} \mid \boldsymbol{\Omega}_k, \mathbf{Y}^{k-1}) d\boldsymbol{\Omega}_{k-1} \quad (41)$$

Thus, the problem of computing the likelihood translates to that of computing the two pdfs under the integral sign.

To approximate $p_{y_k|\boldsymbol{\omega}_k, \boldsymbol{\omega}_{k-1}, \mathcal{Y}^{k-1}}$, Eqs. (24) and (27) are used. Replacing $\mathbf{v}_k(\boldsymbol{\omega}_k)$ with its corresponding approximation $\mathbf{v}_k(\boldsymbol{\omega}_k)$ in Eq. (24) yields

$$\mathbf{v}_k(\boldsymbol{\omega}_k) \approx \mathbf{y}_k - \Gamma(\boldsymbol{\omega}_k)\mathbf{y}_{k-1} \quad (42)$$

Using Eq. (42) in Eq. (27) yields

$$\mathbf{y}_k \approx \Gamma(\boldsymbol{\omega}_k)\mathbf{y}_{k-1} + M_k(\mathbf{y}_{k-1} - \Gamma(\boldsymbol{\omega}_{k-1})\mathbf{y}_{k-2}) + \mathbf{w}_k \quad (43)$$

From Eq. (43) it immediately follows that

$$p_{y_k|\boldsymbol{\omega}_k, \boldsymbol{\omega}_{k-1}, \mathcal{Y}^{k-1}}(\mathbf{Y}_k \mid \boldsymbol{\Omega}_k, \boldsymbol{\Omega}_{k-1}, \mathbf{Y}^{k-1}) \approx p_{\mathbf{w}_k}(\mathbf{Y}_k - \Gamma(\boldsymbol{\Omega}_k)\mathbf{Y}_{k-1} - M_k(\mathbf{Y}_{k-1} - \Gamma(\boldsymbol{\Omega}_{k-1})\mathbf{Y}_{k-2})) = \frac{1}{(2\pi)^{3/2} |R_k^w|^{1/2}} \times \exp \left\{ -\frac{1}{2} [\mathbf{Y}_k - \boldsymbol{\mu}_k(\boldsymbol{\Omega}_k, \boldsymbol{\Omega}_{k-1})]^T (R_k^w)^{-1} [\mathbf{Y}_k - \boldsymbol{\mu}_k(\boldsymbol{\Omega}_k, \boldsymbol{\Omega}_{k-1})] \right\} \quad (44)$$

To compute the second pdf appearing under the integral sign in Eq. (41), $p_{\boldsymbol{\omega}_{k-1}|\boldsymbol{\omega}_k, \mathcal{Y}^{k-1}}$, it is assumed that the process noise in Eq. (7) is of low intensity (spacecraft motion is typically influenced by extremely small external disturbances that form the driving process noise, e.g., aerodynamic forces, gravity gradient, and magnetic dipole torques). Based on this assumption, the transition kernel can be approximated as

$$p_{\boldsymbol{\omega}_k|\boldsymbol{\omega}_{k-1}}(\boldsymbol{\Omega}_k \mid \boldsymbol{\Omega}_{k-1}) \approx \delta(\boldsymbol{\Omega}_k - f(\boldsymbol{\Omega}_{k-1})) \quad (45)$$

where $\delta(\cdot)$ denotes Dirac's delta distribution. Also,

$$p_{\boldsymbol{\omega}_k|\boldsymbol{\omega}_{k-1}, \mathcal{Y}^{k-1}} = p_{\boldsymbol{\omega}_k|\boldsymbol{\omega}_{k-1}} \quad (46)$$

because, given $\boldsymbol{\omega}_{k-1}$, the distribution of the state $\boldsymbol{\omega}_k$ is solely determined by the driving process noise, rendering it independent of the measurement process [see Eq. (7)]. Hence, using Eq. (45), it follows that

$$\begin{aligned}
& p_{\omega_k|\mathcal{Y}^{k-1}}(\boldsymbol{\Omega}_k | Y^{k-1}) \\
&= \int_{-\infty}^{+\infty} p_{\omega_k|\omega_{k-1}}(\boldsymbol{\Omega}_k | \boldsymbol{\Omega}_{k-1}) p_{\omega_{k-1}|\mathcal{Y}^{k-1}}(\boldsymbol{\Omega}_{k-1} | Y^{k-1}) d\boldsymbol{\Omega}_{k-1} \\
&\approx \int_{-\infty}^{+\infty} \delta(\boldsymbol{\Omega}_k - f(\boldsymbol{\Omega}_{k-1})) p_{\omega_{k-1}|\mathcal{Y}^{k-1}}(\boldsymbol{\Omega}_{k-1} | Y^{k-1}) d\boldsymbol{\Omega}_{k-1} \\
&= p_{\omega_{k-1}|\mathcal{Y}^{k-1}}(f^{-1}(\boldsymbol{\Omega}_k) | Y^{k-1})
\end{aligned} \tag{47}$$

Using Eqs. (45–47), $p_{\omega_{k-1}|\omega_k, \mathcal{Y}^{k-1}}$ can be expressed as

$$\begin{aligned}
& p_{\omega_{k-1}|\omega_k, \mathcal{Y}^{k-1}}(\boldsymbol{\Omega}_{k-1} | \boldsymbol{\Omega}_k, Y^{k-1}) \\
&= \frac{p_{\omega_k|\omega_{k-1}, \mathcal{Y}^{k-1}}(\boldsymbol{\Omega}_k | \boldsymbol{\Omega}_{k-1}, Y^{k-1}) p_{\omega_{k-1}|\mathcal{Y}^{k-1}}(\boldsymbol{\Omega}_{k-1} | Y^{k-1})}{p_{\omega_k|\mathcal{Y}^{k-1}}(\boldsymbol{\Omega}_k | Y^{k-1})} \\
&= \frac{p_{\omega_k|\omega_{k-1}}(\boldsymbol{\Omega}_k | \boldsymbol{\Omega}_{k-1}) p_{\omega_{k-1}|\mathcal{Y}^{k-1}}(\boldsymbol{\Omega}_{k-1} | Y^{k-1})}{p_{\omega_k|\mathcal{Y}^{k-1}}(\boldsymbol{\Omega}_k | Y^{k-1})} \\
&\approx \frac{\delta(\boldsymbol{\Omega}_k - f(\boldsymbol{\Omega}_{k-1})) p_{\omega_{k-1}|\mathcal{Y}^{k-1}}(\boldsymbol{\Omega}_{k-1} | Y^{k-1})}{p_{\omega_{k-1}|\mathcal{Y}^{k-1}}(f^{-1}(\boldsymbol{\Omega}_k) | Y^{k-1})} \\
&\approx \delta(\boldsymbol{\Omega}_k - f(\boldsymbol{\Omega}_{k-1}))
\end{aligned} \tag{48}$$

Finally, using Eqs. (44) and (48) in Eq. (41) yields

$$\begin{aligned}
& p_{y_k|\omega_k, \mathcal{Y}^{k-1}}(\mathbf{Y}_k | \boldsymbol{\Omega}_k, Y^{k-1}) \\
&\approx \int_{-\infty}^{+\infty} p_{y_k|\omega_k, \omega_{k-1}, \mathcal{Y}^{k-1}}(\mathbf{Y}_k | \boldsymbol{\Omega}_k, \boldsymbol{\Omega}_{k-1}, Y^{k-1}) \\
&\quad \times \delta(\boldsymbol{\Omega}_k - f(\boldsymbol{\Omega}_{k-1})) d\boldsymbol{\Omega}_{k-1} \\
&= p_{y_k|\omega_k, \omega_{k-1}, \mathcal{Y}^{k-1}}(\mathbf{Y}_k | \boldsymbol{\Omega}_k, f^{-1}(\boldsymbol{\Omega}_k), Y^{k-1}) \\
&\approx \frac{1}{(2\pi)^{3/2} |R_k^w|^{1/2}} \exp\left\{-\frac{1}{2}[\mathbf{Y}_k - \boldsymbol{\mu}_k(\boldsymbol{\Omega}_k, f^{-1}(\boldsymbol{\Omega}_k))]^T\right. \\
&\quad \left. \times (R_k^w)^{-1}[\mathbf{Y}_k - \boldsymbol{\mu}_k(\boldsymbol{\Omega}_k, f^{-1}(\boldsymbol{\Omega}_k))]\right\}
\end{aligned} \tag{49}$$

which completes the proof. \square

3. Computational Effort vs Performance

The colored noise modeling technique, described above, requires the evaluation of the matrices in Eqs. (35) and (37), consequently increasing the computation load. An alternative, simplified approach is to consider each measurement's correlation with its immediate past (its preceding measurement) only, while neglecting the correlation with all other measurements. This approach results in reducing the required computational load, at the price of sacrificing optimality and performance. In this case, the likelihood of each realization $\boldsymbol{\Omega}_k$ is approximated as

$$\begin{aligned}
& \hat{p}_{y_k|\omega_k, \mathcal{Y}^{k-1}}(\mathbf{Y}_k | \boldsymbol{\Omega}_k, Y^{k-1}) = p_{y_k|\omega_k, y_{k-1}}(\mathbf{Y}_k | \boldsymbol{\Omega}_k, Y_{k-1}) \\
&= p_{v_k(\omega_k)}(\mathbf{Y}_k - \Gamma(\boldsymbol{\Omega}_k) Y_{k-1})
\end{aligned} \tag{50}$$

where the pdf of $v_k(\omega_k)$ is computed using the convolution operator, based on Eq. (25)

$$\begin{aligned}
& p_{v_k(\omega_k)}(\mathbf{V}_k(\boldsymbol{\Omega}_k)) \\
&= \int_{-\infty}^{+\infty} p_{\delta y_k}(\mathbf{V}_k(\boldsymbol{\Omega}_k) + \Gamma(\boldsymbol{\Omega}_k) \delta \mathbf{Y}_{k-1}) \\
&\quad \times p_{\delta y_{k-1}}(\delta \mathbf{Y}_{k-1}) d\delta \mathbf{Y}_{k-1}
\end{aligned} \tag{51}$$

The choice whether to implement the proposed noise modeling technique or to disregard the measurements correlation is case dependent, and trial runs should be performed to compare the cost-to-benefit ratios obtained with the two alternatives.

B. Measurement Update

Given the measurement \mathbf{Y}_k , the updated filtering pdf at time k , based on the modified Bayesian recursion in Eq. (26), satisfies

$$p_{\omega_k|\mathcal{Y}^k}(\boldsymbol{\Omega}_k | Y^k) \propto p_{y_k|\omega_k, \mathcal{Y}^{k-1}}(\mathbf{Y}_k | \boldsymbol{\Omega}_k, Y^{k-1}) p_{\omega_k|\mathcal{Y}^{k-1}}(\boldsymbol{\Omega}_k | Y^{k-1}) \tag{52}$$

Let $\{\boldsymbol{\Omega}_k(i)\}_{i=1}^N$ and $\{\tilde{w}_k(i)\}_{i=1}^N$ denote N independent samples from the filtering pdf at time k , and their associated normalized weights, respectively. Setting the importance distribution to be the prior pdf, and using the approximated likelihood in Eq. (39), Eq. (52) yields the updated importance weights as

$$w_k(i) = \hat{p}_{y_k|\omega_k, \mathcal{Y}^{k-1}}(\mathbf{Y}_k | \boldsymbol{\Omega}_k(i), Y^{k-1}) \tilde{w}_{k-1} \tag{53}$$

These weights are then normalized according to Eq. (4).

Equation (53) is referred to as the update stage. Still, in accordance with Eq. (52), an evolution stage needs to be incorporated, as the samples need to represent the propagated pdf.

C. Particle Evolution

Passing the angular rate samples at time $k-1$ through the certainty-equivalent process [Eq. (40)] results in a new set of samples. This is almost equivalent to applying Eq. (1a) to the samples, that is

$$\begin{aligned}
& p_{\omega_k|\mathcal{Y}^{k-1}}(\boldsymbol{\Omega}_k | Y^{k-1}) = \int_{-\infty}^{+\infty} p_{\omega_k|\omega_{k-1}}(\boldsymbol{\Omega}_k | \boldsymbol{\Omega}_{k-1}) \\
&\quad \times p_{\omega_{k-1}|\mathcal{Y}^{k-1}}(\boldsymbol{\Omega}_{k-1} | Y^{k-1}) d\boldsymbol{\Omega}_{k-1}
\end{aligned} \tag{54}$$

The minor difference is due to the process noise distribution that forms the transition kernel $p_{\omega_k|\omega_{k-1}}$. When the process noise is of low intensity, the new angular rate samples thus obtained adequately represent $p_{\omega_k|\mathcal{Y}^{k-1}}$. In other cases, the injection of an additional, artificial noise improves the set.

D. Filtered Angular Rate

At time k , N weighted samples are available. Obtaining the optimal angular rate estimate can be carried out in several ways, depending on the objective. Two methods for doing so are proposed in this work, using the minimum mean-squared error (MMSE) and the maximum a posteriori (MAP) approach.

1. MMSE Approach

The MMSE angular rate estimate is obtained by computing the weighted average of the samples $\{\boldsymbol{\Omega}_k(i)\}_{i=1}^N$, that is

$$\hat{\boldsymbol{\Omega}}_k = \sum_{i=1}^N \tilde{w}_k(i) \boldsymbol{\Omega}_k(i) \tag{55}$$

2. MAP Approach

The MAP angular rate estimate is defined as

$$\hat{\boldsymbol{\Omega}}_k \triangleq \arg \max_{\boldsymbol{\Omega}_k \in \{\boldsymbol{\Omega}_k(i)\}_{i=1}^N} p_{\omega_k|\mathcal{Y}^k}(\boldsymbol{\Omega}_k | Y^k) \tag{56}$$

Using the filter samples $\{\boldsymbol{\Omega}_k(i), \tilde{w}_k(i)\}_{i=1}^N$, the MAP estimate is obtained as

$$\hat{\boldsymbol{\Omega}}_k = \boldsymbol{\Omega}_k \left(\arg \max_{1 \leq i \leq N} \tilde{w}_k(i) \right) \tag{57}$$

Experience shows that the MAP estimate is usually noisier than the MMSE estimate due to the fact that a resampling procedure is carried out every once in a while. Several smoothing methods to overcome this problem are suggested in [12]. (Smoother MAP estimates are obtained by reducing the effective sample size threshold, which results in requiring fewer resampling procedures to take place.)

E. Resampling

To avoid particle degeneracy, a resampling procedure is implemented. The measure of degeneracy adopted here is the effective sample size. Introduced by Kong et al. [16], this criterion is defined using the variances of the importance weights. The effective sample size N_{eff} is defined as

$$N_{\text{eff}} \triangleq \frac{N}{1 + \text{var}w_k} = \frac{N}{E_\pi[w_k^2]} < N \quad (58)$$

An empirical estimate of N_{eff} is given by [12]

$$\hat{N}_{\text{eff}} = \frac{1}{\sum_{i=1}^N \tilde{w}_k(i)^2} \quad (59)$$

The resampling procedure is used whenever $\hat{N}_{\text{eff}} \leq N_{\text{th}}$, where N_{th} is a predetermined threshold. The new set of samples is generated by resampling each particle $\boldsymbol{\Omega}_k(i)$ with probability $\tilde{w}_k(i)$. This consists of multiplying each sample according to its associated normalized weight. The number of offspring for each sample is evaluated by

$$N_k(i) = \text{int}(N\tilde{w}_k(i)), \quad i = 1, \dots, N \quad (60)$$

where $\text{int}(x)$ denotes the integer nearest to x for any $x \in \mathbb{R}$. To compensate for the loss of particle diversity, an artificial perturbation scheme based upon regularization is introduced into the algorithm.

F. Regularization

Let $\{\boldsymbol{\Omega}_k(i)\}_{i=1}^N$ be the stock of N angular rate samples at time k . The number of offspring for each particle in the stock is determined using Eq. (60). The angular rate offspring are then computed in the following manner: a set of vectors are sampled from a 3-dimensional zero-mean, unit covariance Gaussian (or some other) kernel denoted by \mathcal{K} , to obtain

$$\delta\boldsymbol{\Omega}_o(j) \sim \mathcal{K}(0, I_{3 \times 3}), \quad j = 1, \dots, \text{int}^-(N_k(i)/2) \quad (61)$$

where $\text{int}^-(x)$ denotes the nearest integer to x such that $x \geq \text{int}^-(x)$ (floor function). The next step consists of rescaling and rotating these vectors according to some regularization intensity measure. A natural choice is the sample covariance of the set $\{\boldsymbol{\Omega}_k(i)\}_{i=1}^N$,

$$\hat{P}_\Omega = \sum_{i=1}^N \tilde{w}_k(i) \left[\boldsymbol{\Omega}_k(i) - \hat{\boldsymbol{\Omega}}_k \right] \left[\boldsymbol{\Omega}_k(i) - \hat{\boldsymbol{\Omega}}_k \right]^T \quad (62)$$

Thus, the vectors are transformed according to

$$\begin{aligned} \delta\boldsymbol{\Omega}(j) &= h\hat{P}_\Omega^{1/2}\delta\boldsymbol{\Omega}_o(j) \\ \delta\boldsymbol{\Omega}(j + \text{int}^-(N_k(i)/2)) &= -h\hat{P}_\Omega^{1/2}\delta\boldsymbol{\Omega}_o(j) \\ j &= 1, \dots, \text{int}^-(N_k(i)/2) \end{aligned} \quad (63)$$

where h denotes the regularization bandwidth, and $\hat{P}_\Omega^{1/2}$ is the matrix square root of \hat{P}_Ω . If $N_k(i)$ is odd, the last vector $\delta\boldsymbol{\Omega}(N_k(i))$ is set to

$[0, 0, 0]^T$, thus maintaining a symmetric set of $N_k(i)$ vectors $\{\delta\boldsymbol{\Omega}(l)\}_{l=1}^{N_k(i)}$. In this work, the bandwidth h is set as suggested in [12] (p. 253), that is

$$h = [4/(N(n+2))]^{\frac{1}{n+4}} \quad (64)$$

with $n = 3$, corresponding to the angular rate vector dimension. The i th particle's offspring are then obtained by

$$\bar{\boldsymbol{\Omega}}_k(l) = \boldsymbol{\Omega}_k(i) + \delta\boldsymbol{\Omega}(l), \quad l = 1, \dots, N_k(i) \quad (65)$$

After obtaining the angular rate offspring, each one should be weighted properly. This second stage weighting is crucial for the overall performance of the filter. The resampling procedure is, in fact, an external interference that injects random particles that have no past trajectories. Proper weighting of the offspring reduces the effect of this contamination, whereas improper weighting of the offspring degrades the quality of the filtering representation and, in some extreme cases, can cause divergence. Reweighting can be carried out based upon the regularization kernel \mathcal{K} , so that the new particles are treated as if they were sampled from a continuous pdf. Another idea, which tends to give better results, is to reweight the offspring proportionally to their likelihood. Thus, the second stage importance weights are computed as

$$\bar{w}_k(j) = \frac{1}{c} \hat{p}_{y_k | \omega_k, \mathcal{Y}^{k-1}}(\mathbf{Y}_k | \bar{\boldsymbol{\Omega}}_k(j), Y^{k-1}) \quad (66)$$

where the normalizing constant c is selected based on numerical considerations. A particle stock with skewed importance weights can be improved in the next time step by properly choosing c . In this work the value $c = \hat{p}_{y_k | \omega_k, \mathcal{Y}^{k-1}}(\mathbf{Y}_k | \hat{\boldsymbol{\Omega}}_k, Y^{k-1})$ is used.

G. OPF Algorithm Summary

The angular rate particle filtering algorithm is summarized (using informal pseudocode) in Algorithm 1.

V. Robust Interlaced PF Estimator

In this section the OPF is modified to account for uncertainties in the spacecraft inertia tensor. The technique described herein is motivated by the approach of [11], and is based upon interlacing the PF with a maximum-likelihood (ML) estimator for maintaining computational efficiency. In this work the ML estimator is implemented as a secondary, static PF.

Let \mathbf{J} be the columnwise vectorized spacecraft tensor of inertia

$$\mathbf{J} = \text{vec}(\mathbf{J}) \quad (67)$$

This parameter vector will be termed the inertia vector in the sequel. The parameterization of any pdf using the real-valued vector of deterministic but unknown parameters \mathbf{J} is implied by the notational convention $p_{\cdot|\mathbf{J}}(\cdot | \mathbf{J})$. Using the interlacing approach, the PF algorithm is used for the representation of $p_{\omega_k | \mathbf{J}, \mathcal{Y}^k}$, thus keeping the

Algorithm 1 The OPF algorithm for perfectly known inertia tensor

-
- 1 Particle evolution: integrate Eq. (7) over the time interval Δt for all particles $\{\boldsymbol{\Omega}_{k-1}(i)\}_{i=1}^N$, to obtain a new set $\{\boldsymbol{\Omega}_k(i)\}_{i=1}^N$.
 - 2 **if** a new measurement, \mathbf{Y}_k , is available, **then**
 - 3 Update the importance weights using Eq. (53) and normalize for all $i = 1, \dots, N$
 - 4 Compute the filtered angular rate as described in Sec. IV.D.
 - 5 **if** $N_{\text{eff}} < N_{\text{th}}$ **then**
 - 6 Compute regularization intensity using Eq. (62)
 - 7 **for** all $i = 1, \dots, N$ **do**
 - 8 Compute the number of offspring $N_k(i)$ for particle $\boldsymbol{\Omega}_k(i)$
 - 9 Produce regularized weighted offspring as described in Sec. IV.F.
 - 10 **end for**
 - 11 **end if**
 - 12 **else** {time propagation}
 - 13 Obtain the filtered angular rate as described in Sec. IV.D.
 - 14 **end if**
-

dimension of the state low, whereas \mathbf{J} is estimated via a separate estimator assuming the knowledge of $\hat{\omega}_k$. Because the observation model in Eq. (24) is independent of \mathbf{J} ,

$$p_{y_k|\omega_k, \mathcal{Y}^{k-1}, \mathbf{J}} = p_{y_k|\omega_k, \mathcal{Y}^{k-1}} \quad (68)$$

Hence, implementing the PF algorithm for $p_{\omega_k|\mathbf{J}, \mathcal{Y}^k}$ does not require any modification in the computation of the importance weights.

Defined as the ML estimate of \mathbf{J} , the inertia vector estimate is obtained by maximizing the likelihood function

$$\mathcal{L}(\mathbf{J} | Y^k) \triangleq p_{\mathcal{Y}^k|\mathbf{J}}(Y^k | \mathbf{J}) \quad (69)$$

The computation of this estimate is performed via a secondary, static particle filter. The update stage of this filter uses the following approximation of the likelihood function.

A. Likelihood Approximation

The likelihood function $\mathcal{L}(\mathbf{J} | \mathcal{Y}^k)$ can be expressed as

$$\mathcal{L}(\mathbf{J} | Y^k) = \prod_{j=1}^k p_{y_j|\omega_j, \mathcal{Y}^{j-1}, \mathbf{J}}(Y_j | Y^{j-1}, \mathbf{J}) \quad (70)$$

whence

$$\begin{aligned} \mathcal{L}(\mathbf{J} | Y^k) &= \prod_{j=1}^k \int_{-\infty}^{+\infty} p_{y_j|\omega_j, \mathcal{Y}^{j-1}, \mathbf{J}}(Y_j | \boldsymbol{\Omega}_j, Y^{j-1}, \mathbf{J}) \\ &\quad \times p_{\omega_j|\mathcal{Y}^{j-1}, \mathbf{J}}(\boldsymbol{\Omega}_j | Y^{j-1}, \mathbf{J}) d\boldsymbol{\Omega}_j \\ &= \prod_{j=1}^k \int_{-\infty}^{+\infty} p_{y_j|\omega_j, \mathcal{Y}^{j-1}}(Y_j | \boldsymbol{\Omega}_j, Y^{j-1}) \\ &\quad \times p_{\omega_j|\mathcal{Y}^{j-1}, \mathbf{J}}(\boldsymbol{\Omega}_j | Y^{j-1}, \mathbf{J}) d\boldsymbol{\Omega}_j \\ &= \prod_{j=1}^k E[p_{y_j|\omega_j, \mathcal{Y}^{j-1}}(Y_j | \omega_j, \mathcal{Y}^{j-1}) | \mathcal{Y}^{j-1} = Y^{j-1}, \mathbf{J}] \quad (71) \end{aligned}$$

The expectation in Eq. (71) is performed with respect to the propagated filtering density, $p_{\omega_j|\mathcal{Y}^{j-1}, \mathbf{J}}$, thus an empirical approximation of the likelihood is obtained as

$$\hat{\mathcal{L}}(\mathbf{J} | Y^k) = \prod_{j=1}^k \sum_{i=1}^N \tilde{w}_j(i) p_{y_j|\omega_j, \mathcal{Y}^{j-1}}(Y_j | \boldsymbol{\Omega}_{j-1}^+(i), Y^{j-1}) \quad (72)$$

where $\boldsymbol{\Omega}_{j-1}^+(i)$ is the i th angular rate particle at time $j-1$, propagated via Eq. (40) with inertia tensor \mathbf{J} . Equation (72) is computationally intensive because it has to be evaluated over the entire particle set for a given \mathbf{J} . This computational effort can be significantly reduced by further assuming that the angular rate particles are concentrated near the filtered angular rate. In this case the likelihood can be approximated by

$$\begin{aligned} \hat{\mathcal{L}}(\mathbf{J} | Y^k) &\approx \prod_{j=1}^k p_{y_j|\omega_j, \mathcal{Y}^{j-1}}(Y_j | \hat{\boldsymbol{\Omega}}_{j-1}^+, Y^{j-1}) \\ &\approx \prod_{j=1}^k \hat{p}_{y_j|\omega_j, \mathcal{Y}^{j-1}}(Y_j | \hat{\boldsymbol{\Omega}}_{j-1}^+, Y^{j-1}) \quad (73) \end{aligned}$$

where $\hat{\boldsymbol{\Omega}}_{j-1}^+$ is the filtered angular rate at time $j-1$, propagated via Eq. (40) with inertia tensor \mathbf{J} . Equation (73) requires the evaluation of the likelihood only once per time interval for a given realization of \mathbf{J} , thus facilitating real-time computation.

B. Inertia Estimation Using a Secondary Particle Filter

The inertia PF maintains a population of particles representing the inertia tensor elements. The filter's measurement update stage consists of sequentially computing the particles' weights whenever measurements are acquired, as is done in an ordinary PF. However, in

contrary with a conventional PF, which involves an evolution stage, in this case the parameters are kept static over time. The robustness of the resulting interlaced algorithm is gained by leaving the parameter population unchanged over a certain time interval before the first resampling is performed. Thus, the likelihood $\mathcal{L}(\mathbf{J} | Y^k)$ is represented over the entire parameter region, instead of concentrating over just a small portion of it. This requirement is also crucial, because during the initial stage of the estimation process the likelihood function in Eq. (73) is highly inaccurate, because it depends on the angular rate estimates, which can be highly erroneous. The determination of the first time instance k_R , from which resampling is allowed, can be performed using trial and error or by checking the whiteness of the filter innovations over some period of time.

Let $\{\mathbf{J}(i)\}_{i=1}^{N_j}$ denote a population of N_j inertia vectors. Sequential computation of the likelihood for each member of this population requires the evaluation of Eq. (73) whenever a measurement is available. Naturally, the weight associated with every inertia vector in this population is proportional to the likelihood. Denoting the weight of the i th inertia vector at time k by $\varphi_k(i)$, Eqs. (39) and (73) yield

$$\begin{aligned} \varphi_k(i) &= \frac{1}{(2\pi)^{3/2} |R_k^w|^{1/2}} \exp\left\{-\frac{1}{2} \left[Y_k - \boldsymbol{\mu}_k(\hat{\boldsymbol{\Omega}}_{k-1}^+(i), \hat{\boldsymbol{\Omega}}_{k-1}) \right]^T \right. \\ &\quad \left. \times (R_k^w)^{-1} \left[Y_k - \boldsymbol{\mu}_k(\hat{\boldsymbol{\Omega}}_{k-1}^+(i), \hat{\boldsymbol{\Omega}}_{k-1}) \right] \right\} \varphi_{k-1}(i) \\ i &= 1, \dots, N_j \quad (74) \end{aligned}$$

The notation $\hat{\boldsymbol{\Omega}}_{k-1}^+(i)$ implies that the filtered angular rate at time $k-1$ is propagated via Eq. (40) with inertia tensor $\mathbf{J}(i)$. After obtaining the associated weights they are normalized to yield

$$\tilde{\varphi}_k(i) = \frac{\varphi_k(i)}{\sum_{j=1}^{N_j} \tilde{\varphi}_k(j)} \quad (75)$$

Finally, the ML estimate of \mathbf{J} is obtained as the inertia vector having the greatest importance weight in the population $\{\mathbf{J}(i)\}_{i=1}^{N_j}$, that is

$$\hat{\mathbf{J}}_k = \mathbf{J}(\arg \max_{1 \leq i \leq N_j} \tilde{\varphi}_k(i)) \quad (76)$$

The ML estimate thus obtained may be quite rough. Smoother estimates can be derived by taking the weighted average of the inertia vectors, that is

$$\hat{\mathbf{J}}_k = \sum_{i=1}^{N_j} \tilde{\varphi}_k(i) \mathbf{J}(i) \quad (77)$$

1. Computationally Efficient Estimation of the Complete Inertia Tensor

Applying the inertia PF to the complete inertia tensor results in a 6-dimensional PF, which may be computationally intensive. In such a case, large inertia uncertainties may require the use of too many particles to reach an acceptable accuracy.

To alleviate this problem, two separate inertia PFs can be interlaced for approximating

$$\mathcal{L}(\mathbf{J}^d | Y^k) = p_{\mathcal{Y}^k|\mathbf{J}^d, \mathbf{J}^o}(Y^k | \mathbf{J}^d, \hat{\mathbf{J}}_{k-1}^o)$$

and

$$\mathcal{L}(\mathbf{J}^o | Y^k) = p_{\mathcal{Y}^k|\mathbf{J}^d, \mathbf{J}^o}(Y^k | \hat{\mathbf{J}}_{k-1}^d, \mathbf{J}^o)$$

where

$$\mathbf{J}^d \triangleq [J_{11}, J_{22}, J_{33}]^T, \quad \mathbf{J}^o \triangleq [J_{21}, J_{23}, J_{31}]^T \quad (78)$$

Following this idea, the update stage of the PF for $\mathcal{L}(\mathbf{J}^d | Y^k)$ is based

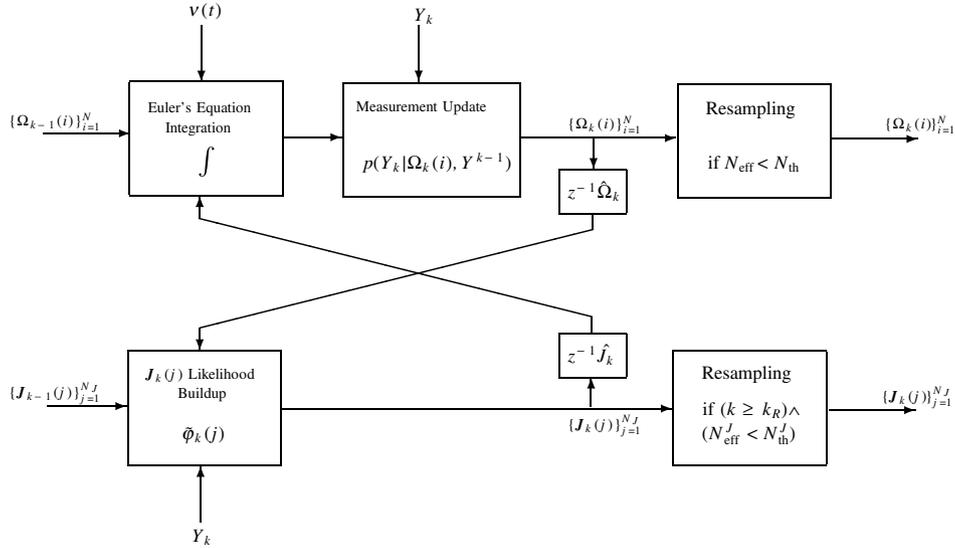


Fig. 1 Robust OPF scheme.

on using Eq. (74) with $\hat{\mathbf{J}}_{k-1}^o$ as a fixed parameter. Analogously, the second PF's update stage uses the latest estimate $\hat{\mathbf{J}}_{k-1}^d$.

Applying these two PFs consecutively whenever a measurement is available might be time consuming. Another approach, which tends to be slightly less accurate but is much faster, is to use the PFs alternately. Thus, only the first PF is applied at the first measurement update, whereas only the second PF is applied at the next one, and so on.

C. Robust OPF Algorithm

A single robust OPF cycle is schematically illustrated in Fig. 1. The algorithm is summarized (using informal pseudocode) in Algorithm 2.

D. OPF Initialization

Large initial attitude errors require a large number of particles, at least until the high-likelihood zones are populated. A simple initialization procedure that requires a significantly smaller number of particles is used in this work. The idea is based on the fact that the first two vector observations define a quaternion of rotation up to one degree of freedom. This quaternion is associated with the rotation of \mathbf{q}_0 onto \mathbf{q}_1 , that is

$$\delta \mathbf{q}_0 = \mathbf{q}_1 \otimes \mathbf{q}_0^{-1} \quad (79)$$

This degree of freedom is used to generate the initial set of $\{Q(\boldsymbol{\Omega}_1(i))\}_{i=1}^N$ [see Eq. (15)] from the first two observations \mathbf{Y}_0 and \mathbf{Y}_1 . The angular rate particles are then obtained using the inverse transformation of Eq. (15).

Algorithm 2 The robust OPF

-
- 1 Start with previous step's estimates, $\hat{\boldsymbol{\Omega}}_{k-1}$ and $\hat{\mathbf{J}}_{k-1}$
 - 2 Use previous step's inertia vector estimate in the evolution stage of the OPF.
 - 3 **if** new measurement, \mathbf{Y}_k , is available, **then**
 - 4 OPF: perform measurement update and get current estimate, $\hat{\boldsymbol{\Omega}}_k$
 - 5 Compute the weights for every member of the set $\{\mathbf{J}(i)\}_{i=1}^{N_J}$ using Eq. (74), and normalize
 - 6 Compute the inertia vector estimate using Eq. (77)
 - 7 If needed, resample the members in $\{\mathbf{J}(i)\}_{i=1}^{N_J}$
 - 8 **if** $N_{\text{eff}} < N_{\text{th}}$ **then**
 - 9 OPF: resample
 - 10 **end if**
 - 11 **else** {time propagation}
 - 12 Obtain the filtered angular rate as described in Sec. IV.D.
 - 13 **end if**
-

VI. Simulation Study

An extensive simulation study has been performed, to assess the performance of the new OPF algorithm and compare it with the recently proposed angular rate EKF algorithm of [10], using a realistic spacecraft model. The study also demonstrates the robustness of the OPF estimator in the presence of uncertain spacecraft inertia.

The spacecraft initial attitude and orbital parameters (i.e., altitude and inclination) are randomly sampled (the altitude not exceeding 1000 km), and each component of the initial angular velocity is sampled from a uniform distribution over the interval $[-30, 30]$ deg/s. In all simulations the spacecraft angular rates are numerically integrated using the Dormand–Prince explicit Runge–Kutta (4,5) formula, implemented in MATLAB's ODE45 routine [17,18]. The integration takes into account the equivalent external disturbance torque, which represents the effects of the aerodynamic, gravity gradient, and residual magnetic dipole torques. Adopting the parameters used in the simulations of [10], this torque is modeled as a stationary white Gaussian noise with intensity $1 (\mu\text{rad})^2/\text{s}^3$. The spacecraft is equipped with a TAM that provides the vector measurements. The TAM noise is modeled as a zero-mean, Gaussian white process with a standard deviation of 50 nT. The Earth magnetic field is modeled using the eighth-order international geomagnetic reference field. Two cases are examined, as described in the sequel.

A. Perfect Model Knowledge

In the first case, the filters are implemented assuming perfect knowledge of the spacecraft inertia tensor. The inertia tensor and the EKF initialization are taken as in [10], thus $\mathbf{J} = \text{diag}\{500, 550, 600\} \text{ kg} \cdot \text{m}^2$. The EKF is running in information form and is initialized with $(P_0)^{-1} = 10^{-8} \text{ I rad}^2/\text{s}^2$ and initial estimate $\hat{\boldsymbol{\omega}}_0 = [0, 0, 0]^T$. The OPF is initialized with $N = 900$ particles, using the initialization scheme described previously. After two measurement updates, the filter continues using only the $N = 150$ particles having the largest importance weights. At first, the OPF is applied using the colored noise modeling technique described in Sec. IV.A. In this case each particle is weighted using the likelihood approximation in Theorem 1. In another set of runs the OPF is examined while taking into account each measurement's correlation with its immediate past only (according to the argumentation in Sec. IV.A). In this case, each particle is weighted using the approximated effective noise pdf in Eqs. (50) and (51). Because the measurement noise is Gaussian, from Eq. (25) it follows that, given $\boldsymbol{\omega}_k$, the effective measurement noise is Gaussian also, with the following statistical moments:

$$E[\mathbf{v}_k(\boldsymbol{\omega}_k) | \boldsymbol{\omega}_k = \boldsymbol{\Omega}_k] = 0, \quad k = 1, 2, \dots \quad (80a)$$

$$R_k \triangleq E[v_k(\omega_k)v_k(\omega_k)^T | \omega_k = \Omega_k] = R_k^y + \Gamma(\Omega_k)R_{k-1}^y\Gamma(\Omega_k)^T$$

$$k = 1, 2, \dots$$

(80b)

[Notice that, as could be expected, Eq. (80b) also results upon setting $M_k = 0$ in Eq. (37), signifying the lack of correlation in the model (27).] Recalling that, in the case under investigation, the measurement noise is stationary, and assuming further that $\Gamma(\Omega_k) \approx I_{3 \times 3}$ (for small enough Δt), Eq. (80b) yields

$$R_k \approx 2R_k^y$$

(81)

Using the simplified method of considering each measurement's correlation with its immediate past only, the likelihood approximation for each particle is given by

$$\hat{p}_{y_k|\omega_k, y^{k-1}}(Y_k | \Omega_k(i), Y^{k-1}) = \frac{1}{(2\pi)^{3/2} |2R_k^y|^{1/2}}$$

$$\times \exp\left\{-\frac{1}{4}[Y_k - \Gamma(\Omega_k(i))Y_{k-1}]^T (R_k^y)^{-1} [Y_k - \Gamma(\Omega_k(i))Y_{k-1}]\right\}$$

(82)

The resampling threshold is set to $N_{th} = \frac{2}{3}N$, based on tuning runs. Decreasing N_{th} may be beneficial, because resampling procedures will be executed less frequently, consequently introducing less Monte Carlo variations into the estimates. However, this also increases the algorithm's sensitivity to heavy-tailed measurement noise pdfs. The numerical integration of the Euler equation in the evolution stage of the PF is performed using the Runge-Kutta-4 formula, with constant time interval of $\Delta t/5$. The TAM sampling rate is 2 Hz. The normed angular rate estimation error (in deg/s) is defined as

$$\|\delta\omega_k\| \triangleq \|\omega_k - \hat{\omega}_k\|_2$$

(83)

Figure 2 shows the spacecraft angular rates in a typical single run. These rates are similar to those tested in [10]. The estimation errors of the OPF and EKF filters in that run are shown in Fig. 3, clearly demonstrating the performance advantage of the OPF over the EKF.

Figure 4 presents the statistical distribution of the angular rate estimation error of the OPF algorithm based on a 5000-run Monte Carlo study. Figure 4a presents the performance of the PF when using the colored noise modeling technique, whereas Fig. 4b shows the estimation error of the simplified PF that uses the likelihood approximation of Eq. (82) (notice the ordinate axis logarithmic scale in both figures). The curves in Figs. 4a and 4b, top to bottom, correspond to the 95, 85, 50, 15, and 5 percentiles of the Monte Carlo runs, respectively. The performance benefit of the colored noise modeling technique is noticeable upon comparing both figures: whereas the 5 and 15 percentile lines in Fig. 4a reach values of approximately 0.002 deg/s and 0.005 deg/s, respectively, the

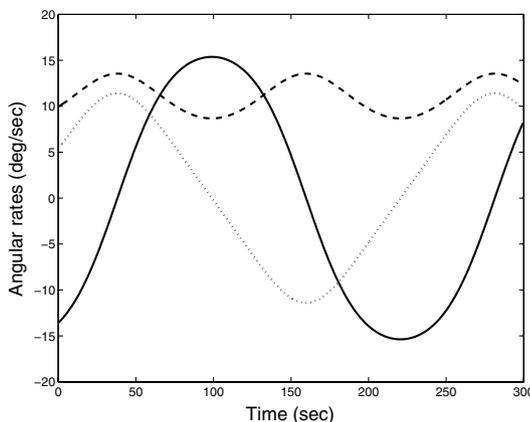


Fig. 2 Spacecraft angular rates in a single typical run: dotted line, ω_x ; solid line, ω_y ; and dashed line, ω_z .

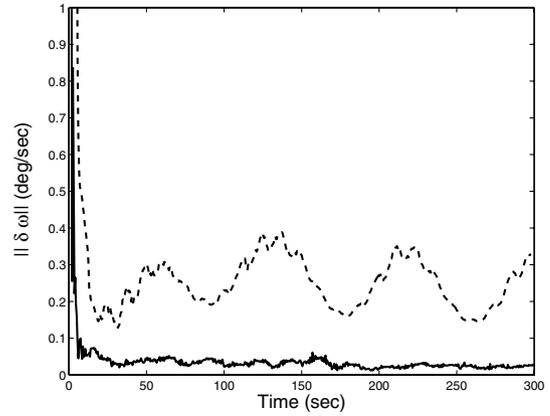
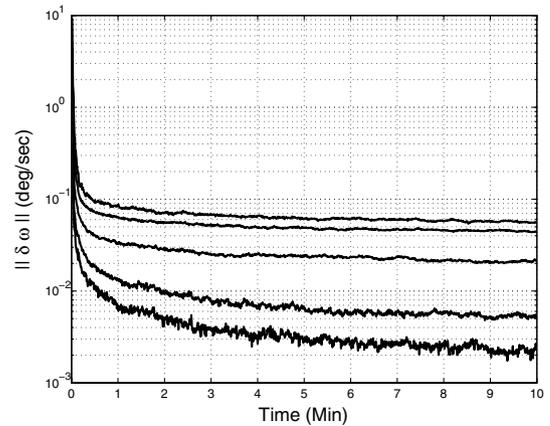
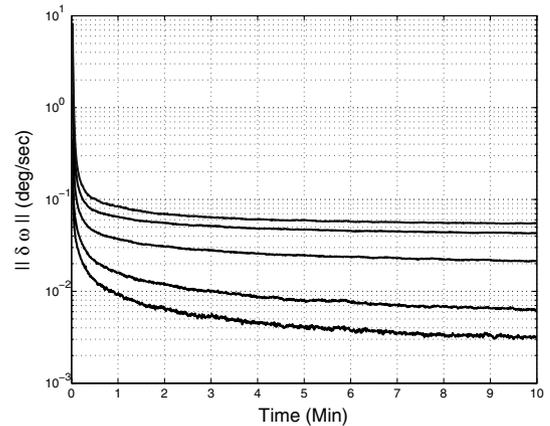


Fig. 3 Angular rate estimation error in a single typical run: solid line, OPF; and dashed line, EKF.



a) Correlated noise OPF



b) Simplified OPF

Fig. 4 Statistical distribution of angular rate estimation error of the OPF algorithm, based on 5000 Monte Carlo runs. Lines, top to bottom: 95, 85, 50, 15, and 5 percentiles.

corresponding percentile lines in Fig. 4b reach values of approximately 0.003 deg/s and 0.006 deg/s, respectively. Thus, although the colored noise modeling technique is more computationally intensive, it does result in smaller minimum estimation errors. Notice that the rougher plots of the 5 and 15 percentiles in Fig. 4a (compare with Fig. 4b) indicate that, as could be expected, the decorrelation procedure increases the filter's sensitivity to the information contained in the measurements (this increased sensitivity is better pronounced at small estimation errors).

For comparison, Fig. 5 presents the corresponding results for the EKF algorithm. As can be seen from Figs. 4 and 5, in 95% of the runs, the OPF's steady-state angular rate estimation error is lower than 0.05 deg/s, whereas the EKF's steady-state estimation error reaches similar values in just 5% of the runs.

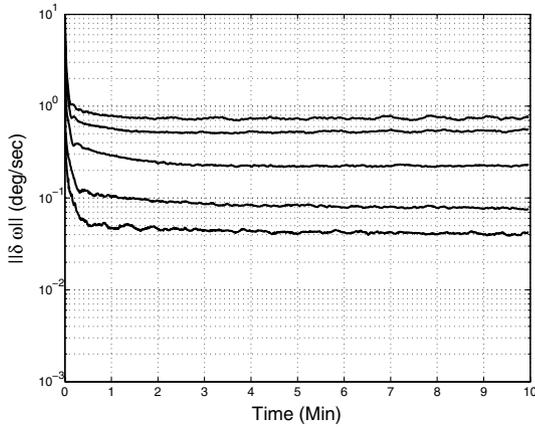


Fig. 5 Statistical distribution of angular rate estimation error of the EKF algorithm, based on 5000 Monte Carlo runs. Lines, top to bottom: 95, 85, 50, 15, and 5 percentiles.

B. Computational Complexity Evaluation

To assess the OPF algorithm’s computational complexity, the mean cycle computation time, denoted by Δt , is adopted as a complexity index. This index is computed by averaging 1000 filter cycle computation times for both the OPF and EKF algorithms, where each filter cycle includes one measurement update and one evolution stage. Both algorithms are implemented using MATLAB (running in interpreter mode) on a Pentium 4/2.8 GHz machine. Table 1, showing the mean cycle computation times for both algorithms, demonstrates that the OPF algorithm with 150 particles is approximately 1.6 times slower than the EKF of [10]. Still, the OPF is sufficiently fast for real-time implementation, because (even when implemented in MATLAB’s interpreter mode) it requires about 200 s for processing 1000 filter cycles, that last 500 s at the filter’s 2 Hz update rate.

C. Model Uncertainty

1. Initial Inertia Uncertainty

This section demonstrates the performance of the simplified OPF filter in the presence of large errors in the inertia tensor. A Monte Carlo study is performed, where the true initial angular rate is uniformly sampled with norm not exceeding 30 deg/s. The spacecraft orbit altitude and inclination are set to 350 km and 35 deg, respectively (corresponding to the orbital parameters of the tropical rainfall measurement mission spacecraft [15]). The true inertia tensor is

$$J = \begin{bmatrix} 500 & 25 & 15 \\ 25 & 550 & 10 \\ 15 & 10 & 600 \end{bmatrix} \text{ kg} \cdot \text{m}^2 \quad (84)$$

The diagonal elements of the inertia tensor are estimated using an inertia PF. Denoting the initial particle population set of this filter by $\{J^d(i)\}_{i=1}^{N_j^d}$, where

$$J^d(i) \triangleq [J_{11}(i) \quad J_{22}(i) \quad J_{33}(i)]^T, \quad i = 1, \dots, N_j^d \quad (85)$$

the particles are sampled from a uniform distribution as follows:

$$\begin{aligned} J_{11}(i) &\sim U[425, 600], & J_{22}(i) &\sim U[467, 660] \\ J_{33}(i) &\sim U[588, 612] \end{aligned} \quad (86)$$

Table 1 Mean filter cycle computation time

Algorithm	Δt
Correlated noise OPF (150 particles)	0.20 s
EKF [10]	0.12 s

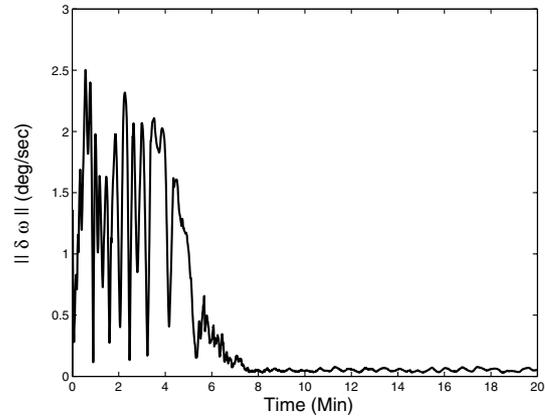


Fig. 6 OPF angular rate estimation error in the presence of inertia tensor uncertainty in a typical single run.

with $N_j^d = 700$ particles. The off-diagonal elements of the inertia tensor are estimated using a secondary inertia PF. Denoting its initial population set by $\{J^o(i)\}_{i=1}^{N_j^o}$, where

$$J^o(i) \triangleq [J_{21}(i) \quad J_{23}(i) \quad J_{13}(i)]^T, \quad i = 1, \dots, N_j^o \quad (87)$$

the particles are sampled from a uniform distribution as follows:

$$J_{21}(i) \sim U[0, 30], \quad J_{23}(i) \sim U[0, 30], \quad J_{13}(i) \sim U[0, 30] \quad (88)$$

with $N_j^o = 700$ particles. The OPF is initialized as before with the same number of particles and the inertia PFs are applied alternately as described in the previous section. In this case $k_R = \infty$ (no resampling is used) for both inertia PFs.

Figure 6 shows the normed angular rate error in a single typical run of the OPF (the angular rates in this run are the same as in Fig. 2). The corresponding inertia component estimation errors are shown in Fig. 7. The effect of the inertia tensor uncertainty is clearly pronounced in the initial time interval of about 8 min, which is the time required for the inertia estimates to converge to within errors of about 7 kg · m².

Figure 8 shows the statistical distribution of the OPF’s angular rate estimation error in this case, based on 1000 Monte Carlo runs. As can be observed from this figure, the steady-state performance of the filter is not significantly affected by the inertia uncertainty. This can be attributed to the fact that the inertia tensor elements are accurately estimated. Indeed, defining the following estimation error measures for the diagonal and off-diagonal elements of the inertia tensor as

$$\|\delta J_k^d\| \triangleq \|\hat{J}_k^d - J^d\|_2 \quad \text{and} \quad \|\delta J_k^o\| \triangleq \|\hat{J}_k^o - J^o\|_2 \quad (89)$$

respectively, the statistical distributions of the inertia tensor

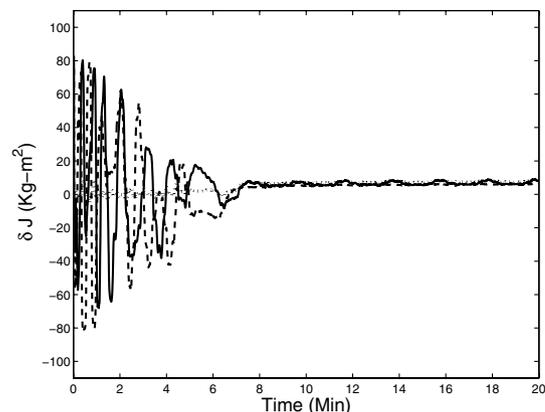


Fig. 7 OPF inertia estimation errors in a typical single run: solid line, δJ_{11} ; dashed line, δJ_{22} ; and dotted line, δJ_{33} .

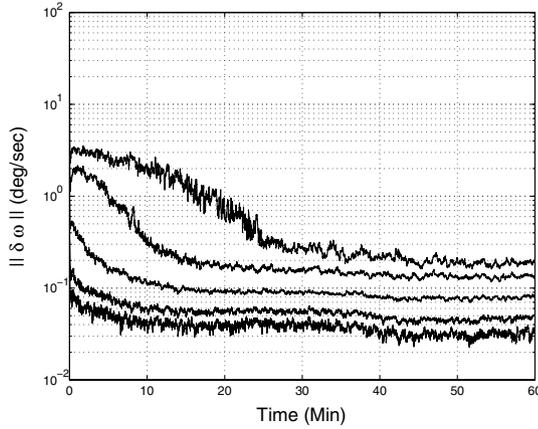
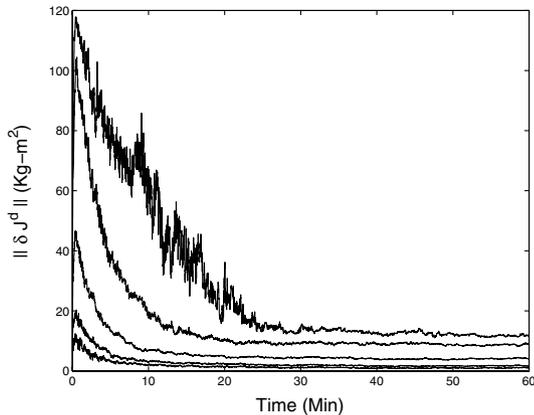


Fig. 8 Statistical distribution of angular rate estimation error of the OPF algorithm in the presence of inertia tensor uncertainty, based on 1000 Monte Carlo runs. Lines, top to bottom: 95, 85, 50, 15, and 5 percentiles.

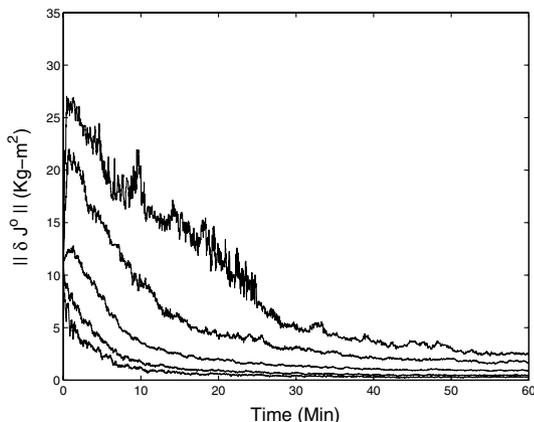
estimation errors are shown in Fig. 9. As can be clearly seen from this figure, it takes the filter about 25 min to estimate the inertia tensor to within an acceptable accuracy.

2. Abrupt Inertia Change

Figures 10 and 11 demonstrate the robustness of the algorithm in the case of a simulated abrupt physical failure that results in a sudden change of the inertia tensor. In this case the true initial spacecraft inertia matrix is $J = \text{diag}\{J_{11}, J_{22}, J_{33}\}$, where the diagonal entries are identical to those of Eq. (84). At $t = 25$ min the true inertia matrix is changed to $J = \text{diag}\{1.2J_{11}, 1.2J_{22}, J_{33}\}$; the OPF estimator is not aware of the sudden change. Notice that, as has



a) Diagonal elements



b) Off-diagonal elements

Fig. 9 Statistical distribution of inertia tensor estimation errors based on 1000 Monte Carlo runs. Lines, top to bottom: 95, 85, 50, 15, and 5 percentiles.

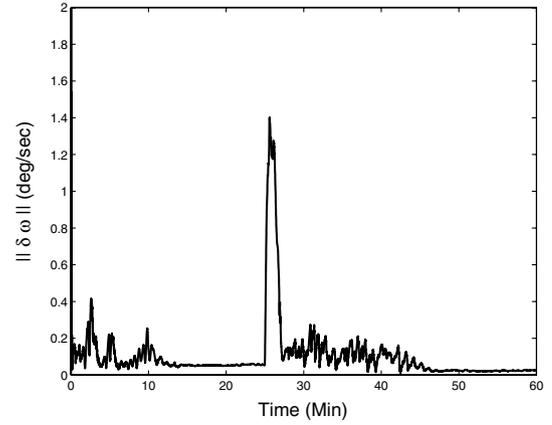


Fig. 10 Angular rate estimation error for the OPF in the presence of an abrupt change in the inertia tensor.

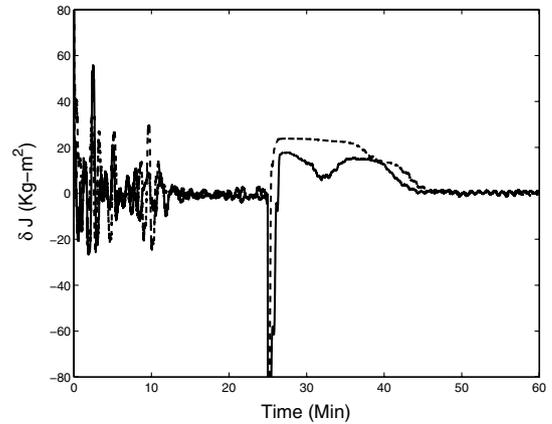


Fig. 11 Inertia components estimation errors in the presence of an abrupt change in the inertia tensor: solid line, δJ_{11} ; dashed line, δJ_{22} .

been observed in [10], when the inertia matrix is diagonal, Euler's equation is invariant relative to scaling the entire inertia matrix, yielding that only two diagonal elements are independent. Hence, in this case, just two inertia terms are modified.

As can be seen from Fig. 10, before the failure occurs the OPF reaches its steady-state estimation error after about 15 min. The change in the inertia tensor at $t = 25$ min does give rise to an estimation error of more than 1 deg/s; however, as Fig. 11 shows, the modified inertia terms are estimated correctly within about 20 min, after which the filter returns to nominal operation.

VII. Conclusions

A novel algorithm is presented for attitude-free estimation of angular rates from vector observations. The new filter belongs to a recently introduced class of angular rate estimators. However, whereas the three algorithms currently belonging to the class are extended Kalman filters, the new angular rate estimator is a particle filter that copes naturally with non-Gaussian driving processes and with the inherent nonlinearity of the angular rate estimation problem. Because the effective measurement noise in the problem under consideration is time correlated, a special procedure has been developed to enable the particle filter (classically designed for white noise processes) to cope with this correlation. To increase the estimator's robustness with respect to spacecraft inertia uncertainty, a secondary static particle filter is used that computes the maximum-likelihood estimate of the entire spacecraft inertia tensor. The secondary inertia filter is run in an interlaced manner with the primary rate particle filter. Thus, the need to augment the filter's state is avoided, along with the potential computational burden associated with the increased number of required particles. The performance of the new algorithm is shown via simulations to be superior to that of a recently introduced extended Kalman filter-based rate estimator. A

Monte Carlo simulation study is used to demonstrate the algorithm's robustness with respect to spacecraft inertia variations of up to 20%.

Acknowledgments

This research was supported by the Israel Science Foundation (Grant No. 1032/04), the Technion's Asher Space Research Fund, and the Robert and Mildred Rosenthal Aerospace Engineering Research Fund. The authors wish to express their gratitude to Paolo Tortora of the university of Bologna, for providing his EKF computer code.

References

- [1] Psiaki, M. L., Martel, F., and Pal, P. K., "Three Axis Attitude Determination via Kalman Filtering of Magnetometer Data," *Journal of Guidance, Control, and Dynamics*, Vol. 13, No. 3, 1999, pp. 506–514.
- [2] Challa, M., Kotaru, S., and Natanson, G., "Magnetometer-Only Attitude and Rate Estimates During the Earth Radiation Budget Satellite 1987 Control Anomaly," *Proceedings of the AIAA Guidance, Navigation, and Control Conference*, Vol. 2, AIAA, Reston, VA, 1997, pp. 830–840.
- [3] Azor, R., Bar-Itzhack, I. Y., and Harman, R. R., "Satellite Angular Rate Estimation from Vector Measurements," *Journal of Guidance, Control, and Dynamics*, Vol. 21, No. 3, May–June 1998, pp. 450–457.
- [4] Harman, R. R., and Bar-Itzhack, I. Y., "Pseudolinear and State-Dependent Riccati Equation Filters for Angular Rate Estimation," *Journal of Guidance, Control, and Dynamics*, Vol. 22, No. 5, 1999, pp. 723–725.
- [5] Azor, R., Bar-Itzhack, I. Y., Deutschmann, J. K., and Harman, R. R., "Angular-Rate Estimation Using Delayed Quaternion Measurements," *Journal of Guidance, Control, and Dynamics*, Vol. 24, No. 3, May–June 2001, pp. 436–443.
- [6] Bar-Itzhack, I. Y., Harman, R. R., and Choukroun, D., "State-Dependent Pseudo Linear Filters for Spacecraft Attitude and Rate Estimation," *Proceedings of the AIAA Guidance, Navigation, and Control Conference*, AIAA, Reston, VA, Aug. 2002.
- [7] Crassidis, J. L., and Markley, F. L., "Predictive Filtering for Attitude Estimation Without Rate Sensors," *Journal of Guidance, Control, and Dynamics*, Vol. 20, No. 3, May–June 1997, pp. 522–527.
- [8] Oshman, Y., and Dellus, F., "Spacecraft Angular Velocity Estimation Using Sequential Observations of a Single Directional Vector," *Journal of Spacecraft and Rockets*, Vol. 40, No. 2, March–April 2003, pp. 237–247.
- [9] Psiaki, M. L., and Oshman, Y., "Spacecraft Attitude Rate Estimation From Geomagnetic Field Measurements," *Journal of Guidance, Control, and Dynamics*, Vol. 26, No. 2, March–April 2003, pp. 244–252.
- [10] Tortora, P., Oshman, Y., and Santoni, F., "Spacecraft Angular Rate Estimation from Magnetometer Data Only Using an Analytic Solution of Euler's Equations," *Journal of Guidance, Control, and Dynamics*, Vol. 27, No. 3, 2004, pp. 365–373.
- [11] 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.
- [12] Doucet, A., de Freitas, N., and Gordon, N. (eds.), *Sequential Monte Carlo Methods in Practice*, Statistics for Engineering and Information Science, Springer, New York, 2001.
- [13] Geweke, J., "Bayesian Inference in Econometric Models Using Monte Carlo Integration," *Econometrica: Journal of the Econometric Society*, Vol. 57, No. 6, Nov. 1989, pp. 1317–1339.
- [14] Doucet, A., Godsill, S., and Andrieu, C., "On Sequential Monte Carlo Sampling Methods for Bayesian Filtering," *Statistics and Computing*, Vol. 10, No. 3, 2000, pp. 197–208.
- [15] Crassidis, J. L., and Markley, F. L., "Unscented Filtering for Spacecraft Attitude Estimation," *Journal of Guidance, Control, and Dynamics*, Vol. 26, No. 4, Aug. 2003, pp. 536–542.
- [16] Kong, A., Liu, J., and Wong, W. H., "Sequential Imputations and Bayesian Missing Data Problems," *Journal of the American Statistical Association*, Vol. 89, No. 425, 1994, pp. 278–288.
- [17] Dormand, J. R., and Prince, P. J., "A Family of Embedded Runge-Kutta Formulae," *Journal of Computational and Applied Mathematics*, Vol. 6, No. 1, 1980, pp. 19–26.
- [18] Higham, D. J., and Higham, N. J., *MATLAB Guide*, Society for Industrial and Applied Mathematics, Philadelphia, PA, 2000.