NONLINEAR FILTERING IN A SIMULATED THREE-AXIS SATELLITE ATTITUDE ESTIMATION AND CONTROL TESTED

Ronan Arraes Jardim Chagas

Divisão de Engenharia Eletrônica, Departamento de Sistemas e Controle
Instituto Tecnológico de Aeronáutica, São José dos Campos, São Paulo, Brasil
E-mail: ronan.jardim@gmail.com

Jacques Waldmann

Divisão de Engenharia Eletrônica, Departamento de Sistemas e Controle
Instituto Tecnológico de Aeronáutica, São José dos Campos, São Paulo, Brasil
E-mail: jacques@ita.br

Abstract: *This article investigates the performance of three distinct approaches to nonlinear filtering applied to a simulated three-axis satellite testbed used for evaluating attitude estimation and control algorithms: extended and unscented Kalman filters and a regularized particle filter. Each approach is numerically evaluated with respect to attitude and angular rate estimation accuracy, computational workload, convergence rate under uncertain initial conditions, and sensitivity to disturbances.*

Keywords: *Nonlinear Filtering, Extended Kalman Filter, Unscented Kalman Filter, Regularized Particle Filter, Nonlinear Dynamics.*

1 Introduction

Nonlinear estimation based on extended Kalman, unscented, and particle filtering is investigated to gauge the performance tradeoff among attitude and angular rate estimation accuracy, robustness to uncertain initial conditions, and computational workload. This investigation has been motivated by an experimental setup in the LabSim at INPE, where a 3-axis, air-suspended table has been instrumented as a testbed for designing and testing satellite attitude control algorithms and systems integration. The apparatus was used as an educational tool in a demonstration to graduate students and stirred up such an interest that inspired the effort to simulate a model of a similar testbed for the evaluation of nonlinear estimation algorithms and their feasibility for actual space applications, specifically the attitude control system of the low-cost ITASAT university satellite.

The simulated table neglects the actual mass unbalance and corresponding pendulous effect due to gravity torque. This inconvenience can be circumvented in the application of the results by careful table mass distribution. Sun sensors on board an orbiting satellite provide a reference direction for attitude estimation and control. This function can be simulated on the ground by use of accelerometers measuring the local vertical, given by the reaction to the gravity vector, and assuming the horizontal acceleration is negligible. Hence, one requirement for the control system is that it should align the table with the local horizontal plane. Hence, the table presents a sensor suite composed of two accelerometers, and in addition to that one 3-axis magnetometer. The latter is a means to measure on board the required additional reference direction, namely the local geomagnetic field, which is also provided by an external, horizontally aligned, ground-fixed 3-axis magnetometer. The actuator suite is composed of a momentum wheel for azimuth control Carrara and Milani within 1° relative to a desired direction, and compressed air nozzles for on-off torquing the table towards alignment within 0.5º relative to the local horizontal plane. Notice that the focus is on nonlinear attitude and angular rate estimation. Therefore, the control strategy has been devised making use of the usual linearized dynamics approach with feedback of the state estimate.

In addition to recent published work on extended Kalman and unscented filtering Chagas and Waldmann (2010), a third nonlinear estimator has been designed and tuned to determine table angular rates, Euler angles, and momentum wheel speed with respect to the table: a regularized particle filter. Robustness to model mismatch has been performed by neglecting momentum wheel friction and back-emf coefficients, mass unbalance, i.e. inertia products, and accelerometers biases in the estimator design. The norm of the angle estimation error about the error Euler axis and of the angular rate vector estimation error have been employed for performance evaluation of the nonlinear estimators.

Presented at the VI Simpósio Brasileiro de Engenharia Inercial, Rio de Janeiro, Brazil, October, 2010

2 System model

The mathematical system model is identical as the one built at Chagas and Waldmann (2010), but it is presented here for the sake of completeness.

A. Coordinate Frames

Three coordinate frames have been used to derive an adequate model. The first one is the body-fixed coordinate frame, or $\{X_b, Y_b, Z_b\}$, which is attached to the table with the Z axis perpendicular to the table plane and points upward. The second coordinate frame is the desired reference frame, or $\{X_d, Y_d, Z_d\}$, which is in alignment with the external magnetometer axes. Both *b* and *d* frames are shown in Fig. 1. The rotation sequence has been parameterized by Euler angles ψ, θ and ϕ, respectively about body axes Z_b, Y_b, and X_b, thus rotating a vector representation from the reference frame basis to the body frame one. Note that here the inertial coordinate frame neglects Earth's rotation rate.

The reference frame has been useful for comparing the built-in magnetometer measurements with respect to the external magnetometer data. Additionally, a horizontal coordinate frame, or $\{X_h, Y_h, Z_h\}$, has resulted from rotating the body-fixed, table coordinate frame with the above ϕ and θ Euler angles about X_b, and Y_b axes, respectively. The resulting horizontal frame is rotated by angle ψ about the local upward vertical with respect to the desired reference frame. This is shown in Fig. 2.

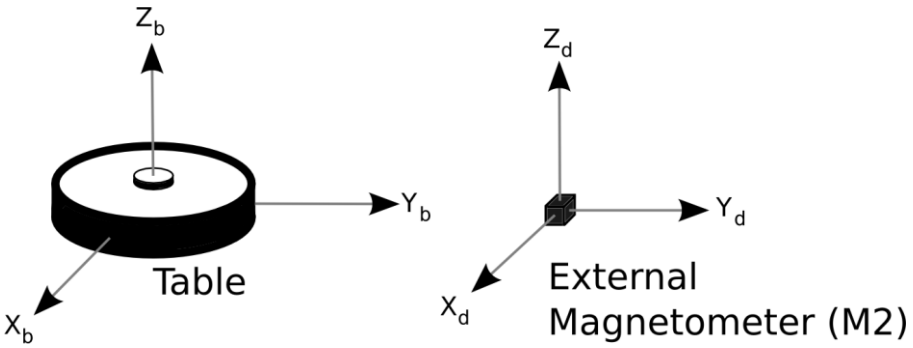

Figure 1. Table reference (left), and desired frame (right)

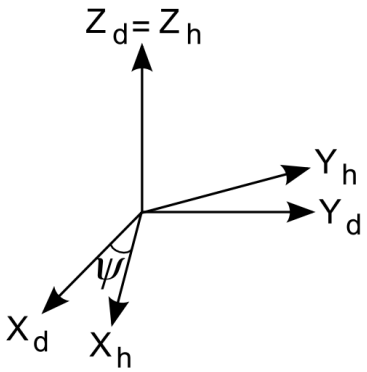

Figure 2. Horizontal frame

B. Sensors

The air-suspended table relies on three sensors for attitude estimation: two accelerometers and one magnetometer. The accelerometers are used to estimate the local vertical, and thus align the table with the horizontal coordinate frame. Data from the built-in magnetometer, called M1, has been compared with the output of the external magnetometer, called M2, to determine the error with respect to the reference azimuth direction about the local vertical.

The two accelerometers measure X_b ($Asp_{b,1}$), and Y_b ($Asp_{b,2}$) components of specific force due to gravity's reaction in the body-fixed, table coordinate frame, as in Eq. 1:

$$\mathbf{Asp}_b = \begin{bmatrix} Asp_{b,1} \\ Asp_{b,2} \\ Asp_{b,3} \end{bmatrix} = \mathbf{D}_d^b . \begin{bmatrix} 0 \\ 0 \\ 9{,}81 \end{bmatrix} = \begin{bmatrix} -9{,}81.\sin(\theta) \\ 9{,}81.\cos(\theta)\sin(\phi) \\ 9{,}81.\cos(\theta)\cos(\phi) \end{bmatrix} \quad (1)$$

where $\mathbf{D}_d^b$ is the direction cosine matrix (DCM) that transforms a vector representation from the reference frame to the table coordinate frame. Accelerometer bias and measurement noise have not been considered in Eq. 1, but they have been added to validate the closed-loop control and the distinct estimators.

Both magnetometers have been assumed to be located in such way that the local magnetic field vector is the same at both locations. Otherwise, comparing their respective measurements would not be feasible for tracking the reference azimuth direction thus compromising accuracy when estimating Euler angle ψ.

The magnetometer on board the air-suspended table outputs a vector measurement, $\mathbf{M2}_b$, which called for representation in the horizontal coordinate frame. That has been done with the estimated Euler angles $\hat{\phi}$ and $\hat{\theta}$ to approximate the DCM $\mathbf{D}_b^h$, as in Eq. 2.

$$\mathbf{M2}_h \approx \mathbf{D}_d^b . \mathbf{M2}_b \quad (2)$$

One can compare $\mathbf{M1}_h$ and $\mathbf{M2}_b$ to approximate the desired Euler angle ψ, which is the angle about the local vertical that the horizontal frame must be rotated to be in alignment with the reference frame, thus yielding Eq. 3:

$$\sin(\psi) = \mathrm{M1}_{\mathrm{d},2}.\mathrm{M2}_{\mathrm{h},1} - \mathrm{M1}_{\mathrm{d},1}.\mathrm{M2}_{\mathrm{h},2} \quad (3)$$

where $Mx_{d,y}$ is the *y*-th component of the unit-norm measurement vector produced by the *x*-th magnetometer. Therefore, the sensor suite described here allows for the measurement of the three Euler angles that rotate the desired reference coordinate frame into alignment with the table frame.

C. Actuators

A set of three actuators is used to control the air-suspended table about its three axes: two pneumatic actuators for the X_b and Y_b axes, and one reaction wheel for Z_b. The pneumatic actuators are controlled by a pulse width modulation (PWM) signal that determine the on-off duty cycle. Additive white noise has been included in the actuator model to account for the small turbulence at the nozzles when torquing the table. Three parameters are called for in such a model: the torque magnitude that is applied on the table by the nozzles when the actuator is on, the frequency of the PWM carrier, and the actuator noise variance.

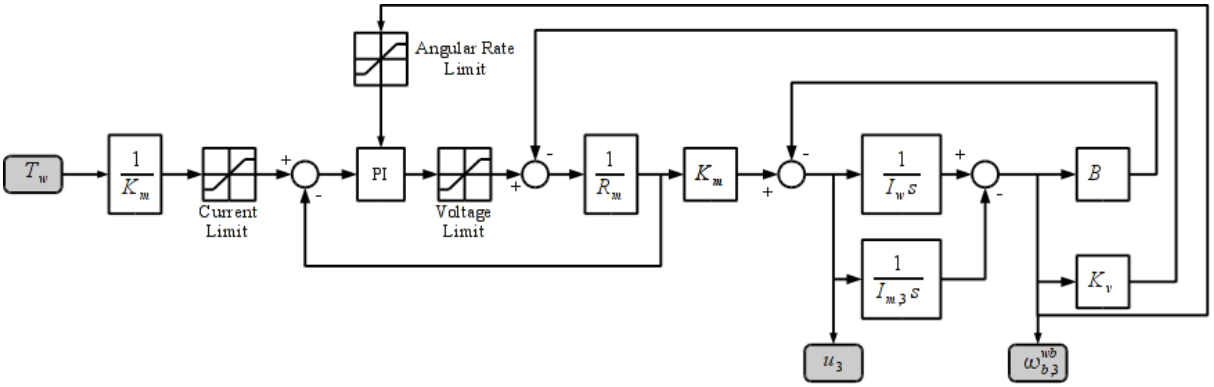

Figure 3. Reaction wheel block diagram

The reaction wheel was modeled as in Sidi (1997) for the purpose of validating estimation and control algorithms. This ground-truth model has included wheel motor dynamics, current and voltage limits, viscous friction, back-emf, and the maximum angular rate limit. Zero-crossing dead-band has been discarded from this model because it can be easily avoided by assuming that the reaction wheel can be initialized with non-zero angular rate, hence becoming a biased momentum wheel, and such an initial condition does not affect the results.

The corresponding block diagram can be seen at Fig. 3, where I_m is the wheel inertia, $I_{m,3}$ is the table inertia around the Z_b axis, K_m, K_v, R_m and B are electromechanical wheel parameters, T_w is the commanded torque, and u_3 is the real torque.

The wheel angular rate with respect to the air-suspended table can be measured by an embedded tachometer in the device. This measurement is ω_{tac} and it is composed of $\omega_{b,3}^{wb}$ plus white-Gaussian noise.

D. Dynamic System Model

The dynamic system model was adapted from Sidi (1997). The table inertia matrix without the reaction wheel $\mathbf{I}_{\mathrm{m,b}}$, and the reaction wheel inertia matrix $\mathbf{I}_{\mathrm{w,b}}$, both represented in the body-fixed table coordinate frame *b*, are shown in Eqs. 4.

$$\mathbf{I}_{\mathrm{m,b}} = \begin{bmatrix} I_{m,b,1} & I_{m,b,1,2} & I_{m,b,1,3} \\ I_{m,b,2,1} & I_{m,b,2} & I_{m,b,2,3} \\ I_{m,b,3,1} & I_{m,b,3,2} & I_{m,b,3} \end{bmatrix}$$
$$\mathbf{I}_{\mathrm{w,b}} = \begin{bmatrix} I_{w,b,1} & I_{w,b,1,2} & I_{w,b,1,3} \\ I_{w,b,2,1} & I_{w,b,2} & I_{w,b,2,3} \\ I_{w,b,3,1} & I_{w,b,3,2} & I_{w,b,3} \end{bmatrix} \tag{4}$$

The table angular rate vector with respect to the inertial frame represented in the *b* coordinate frame $\boldsymbol{\omega}_b^{bi}$, and the reaction wheel angular rate vector with respect to the table represented in the same coordinate frame $\boldsymbol{\omega}_b^{wb}$ are shown in Eqs. 5:

$$\boldsymbol{\omega}_b^{bi} = \begin{bmatrix} \omega_{b,1}^{bi} & \omega_{b,2}^{bi} & \omega_{b,3}^{bi} \end{bmatrix}^T \qquad \boldsymbol{\omega}_b^{wb} = \begin{bmatrix} 0 & 0 & \omega_{b,3}^{wb} \end{bmatrix}^T \tag{5}$$

Based on a Newtonian approach, the dynamic model is represented in the table coordinate frame *b* as in Eq. 6:

$$\begin{bmatrix} \dot{\omega}_{b,1}^{bi} \\ \dot{\omega}_{b,2}^{bi} \\ \dot{\omega}_{b,3}^{bi} \end{bmatrix} = \begin{bmatrix} I_{m,b,1} + I_{w,b,1} & I_{m,b,1,2} + I_{w,b,1,2} & I_{m,b,1,3} \\ I_{m,b,2,1} + I_{w,b,2,1} & I_{m,b,2} + I_{w,b,2} & I_{m,b,2,3} \\ I_{m,b,3,1} + I_{w,b,3,1} & I_{m,b,3,2} + I_{w,b,3,2} & I_{m,b,3} \end{bmatrix}^{-1} \cdot$$
$$\cdot\left(-\mathbf{I}_{w,b} \cdot \begin{bmatrix} 0 \\ 0 \\ \dfrac{u_3}{I_{w,b,3}} \end{bmatrix} - \boldsymbol{\omega}_b^{bi} \times \mathbf{H}_b + \mathbf{T}_{c,b} + \mathbf{T}_{d,b}\right) \tag{6}$$

Where $\mathbf{T}_{d,b}$ is the disturbance torque, $\mathbf{T}_{c,b}$ is the control torque output by the pneumatic actuators and shown in Eq. 7, $\mathbf{H}_b$ is the total angular momentum of the table and the reaction wheel as in Eq. 8, and u_3 is the actual torque acting on the reaction wheel as shown at Fig. 3. Unbalance torque due to gravity has been disconsidered since the testbed is assumed to have undergone a balancing procedure to align its mass center with the table air bearing.

$$\mathbf{T}_{c,b} = \begin{bmatrix} T_{c,b,1} & T_{c,b,2} & 0 \end{bmatrix}^T \tag{7}$$

$$\mathbf{H}_b = (\mathbf{I}_{m,b} + \mathbf{I}_{w,b}) \cdot \boldsymbol{\omega}_b^{bi} + \mathbf{I}_{w,b} \cdot \boldsymbol{\omega}_b^{wb} \tag{8}$$

The table angular rate vector $\boldsymbol{\omega}_b^{bi}$ relates to the attitude kinematics given by the Euler angles ψ, θ, and ϕ time derivatives according to Eqs. 9:

$$\begin{aligned}\dot{\Phi} &= \omega_{b,1}^{bi} + \sin(\phi)\tan(\theta)\omega_{b,2}^{bi} + \cos(\phi)\tan(\theta)\omega_{b,3}^{bi} \\ \dot{\theta} &= \cos(\phi)\omega_{b,2}^{bi} - \sin(\phi)\omega_{b,3}^{bi} \\ \dot{\psi} &= \frac{\sin(\phi)}{\cos(\theta)}\omega_{b,2}^{bi} + \frac{\cos(\phi)}{\cos(\theta)}\omega_{b,3}^{bi}\end{aligned} \quad (9)$$

Hence, the ground-truth model has been constructed using Eqs. 6, 8, 9, and the reaction wheel model seen in Fig. 3.

E. Model States and Measurement Vector

Analyzing the model equations in the previous section, a vector state with seven real components has been defined: the three Euler angles that rotate from the reference frame to the body-fixed table frame, the three components of the angular rate vector of the table with respect to the inertial frame, and the reaction wheel speed with respect to the table. Static friction torque in the vicinity of the reaction wheel zero speed yields a steady-state pointing error about the Z_b axis. Therefore, the integral of such pointing error, shown in Eq. 10, has been selected to augment the state vector as the eighth state component as seen in Eq. 11.

$$\varepsilon = \int_0^t (\psi_{ref} - \psi)dt \rightarrow \dot{\varepsilon} = \psi_{ref} - \psi \quad (10)$$

$$\mathbf{x} = \begin{bmatrix}\phi & \theta & \psi & \omega_{b,1}^{bi} & \omega_{b,2}^{bi} & \omega_{b,3}^{bi} & \omega_{b,3}^{wb} & \varepsilon\end{bmatrix}^T \quad (11)$$

The reference state is given by Eq. 12. Thus, the controller should align the table with the local horizontal plane, and likewise the on-board magnetometer measurement components with those of the external magnetometer.

$$\mathbf{x}_{ref} = \begin{bmatrix}0 & 0 & 0 & 0 & 0 & 0 & 0 & 0\end{bmatrix}^T \quad (12)$$

Recalling Equation 3, the measurement vector concatenates accelerometers, magnetometers and tachometer data as in Eq. 13.

$$\mathbf{y} = \begin{bmatrix} -\dfrac{Asp_{b,1}}{9{,}81} \\ \dfrac{Asp_{b,2}}{9{,}81} \\ \mathrm{M1_{d,2}.M2_{h,1}} - \mathrm{M1_{d,1}.M2_{h,2}} \\ \omega_{tac} \end{bmatrix} \quad (13)$$

3 Control strategy

The main focus is to investigate and compare the performance of three nonlinear estimators. Consequently, a straightforward control technique based on state feedback has been used. Firstly, the system has been linearized around $\mathbf{x}_{ref}$. As a result, the horizontal plane dynamics given by state components ϕ, θ, $\omega_{b,1}^{bi}$ and $\omega_{b,2}^{bi}$ has become decoupled from the vertical dynamics embedded in the remaining state components. Such decoupling allowed for the design of two separate state feedback control laws for the horizontal and vertical dynamics, respectively. Then, the closed-loop poles in Eqs. 14 have been located to avoid actuator saturation while still yielding an acceptable settling time.

$$\begin{aligned}\mathbf{p}_{horizontal} &= \begin{bmatrix}-1 & -1 & -1{,}5 & -1{,}5\end{bmatrix} \\ \mathbf{p}_{vertical} &= \begin{bmatrix}-0{,}2 + j0{,}2 & -0{,}2 - j0{,}2 & -0{,}15\end{bmatrix}\end{aligned} \quad (14)$$

Additionally, each horizontal axis control is turned off when the corresponding Euler angle error norm is less than 0,25°, and the control is switched back on when this error is higher than 0,5°. This avoids high-frequency switching in actuators when the system is near the reference.

4 Estimators

Given the nonlinear model, an extended Kalman filter (EKF), an unscented Kalman filter (UKF) and a regularized particle filter (RPF) have been compared to gauge the performance in terms of estimation accuracy and computational load. This section describes details about the implementation of the estimators. The filters assumed a set of model simplifications. The disturbance torques have been considered nonexistent, the inertia matrices of table and reaction wheel considered diagonal, i.e., without inertia products, and the reaction wheel friction and back-emf have been neglected, i.e. $T_w = u_3$.

The continuous mathematical model, omitting model and measurement noise, can be written as in Eqs. 15:

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x},\mathbf{u}) \qquad \mathbf{y} = \mathbf{h}(\mathbf{x}) \tag{15}$$

where $\mathbf{u}$ is a vector containing commanded torques for both pneumatic actuators and the reaction wheel, and $\mathbf{f}(.)$ is a function concatenating Eqs. 4 to 10 considering the aforementioned simplifications.

The particle filter algorithm needs a discrete system model. The discretization can be done, also omitting model and measurement noise, using Euler approximation as in Eqs. 16 Ristic et al.(2004):

$$\begin{aligned} \mathbf{x}_k &\approx \mathbf{x}_{k-1} + \mathbf{f}(\mathbf{x}_{k-1},\mathbf{u}_{k-1}).\Delta \\ \mathbf{y}_k &= \mathbf{h}(\mathbf{x}_k) \end{aligned} \tag{16}$$

where Δ denotes the sample time.

The mathematical model of the real system does not have any approximation; it has been built using Newton's law. So it doesn't have any modeling noise. However implementation artifacts contribute to the mismatch between the ground-truth model and the one embedded in the estimators. These errors, including those arising from numerical round-off, are merged into a virtual, additive, white noise sequence approximated by a Gaussian probability density with zero mean and a tuned covariance.

A. Extended Kalman Filter

Regarding Eqs. 16 augmented with respective additive noise, the EKF performs the linearization of the dynamics about the updated state estimate, whereas the linearization of the measurement equation is about the propagated state estimate Ristic et al.(2004). The continuous-discrete approach for the EKF has been used here. Being a slightly different algorithm with respect to the usual implementation, the approach offers improved performance in continuous systems than the discrete approach, because propagation is done by directly integrating the continuous system model equations using, for example, the 4th order Runge-Kutta numerical integration algorithm, whereas the measurements are discrete in time. Propagation is shown in Eqs. 17:

$$\begin{aligned} \hat{\mathbf{x}}_{k|k-1} &= \hat{\mathbf{x}}_{k-1|k-1} + \int_{t_{k-1}}^{t_k} \mathbf{f}(\mathbf{x},\mathbf{u})dt \\ \mathbf{P}_{k|k-1} &= \mathbf{P}_{k-1|k-1} + \\ &+ \int_{t_{k-1}}^{t_k} (\mathbf{J}_f(\mathbf{x},\mathbf{u})\mathbf{P}(t) + \mathbf{P}(t)\mathbf{J}_f(\mathbf{x},\mathbf{u})^T + \mathbf{Q}(t))dt \end{aligned} \tag{17}$$

where $\mathbf{J}_f(\mathbf{x},\mathbf{u})$ is the Jacobian matrix of the function $\mathbf{f}(.)$ at the point $(\mathbf{x},\mathbf{u})$, and $\mathbf{Q}(t)$ is the continuous-time model noise P.S.D. matrix.

The update step is performed as usual, and the above EKF has been implemented previously Chagas and Waldmann (2010).

B. Unscented Kalman Filter

The UKF uses the unscented transform to achieve improved estimation relative to the EKF in case Eq. 15, in our case augmented with additive noise, is highly nonlinear. The unscented transform calculates a set of σ-points that are propagated using the nonlinear model and measurement equations to estimate the mean and covariance of the stochastic state vector Ristic et al.(2004). Unlike the EKF, it does not need computation of Jacobians.

Nevertheless, computing σ-points requires a great amount of computational effort, which makes this method slower than the EKF in almost every practical situation. The UKF has been also used with the continuous-discrete approach described in Särkkä. The UKF has been also implemented in Chagas and Waldmann (2010).

C. Regularized Particle Filter

Particle filters perform sequential Monte Carlo estimation based on point masses, also called particles, to represent the *a posteriori* state probability density. Each particle is located in a point in the state-space and its corresponding weight measures its importance with respect to the current measurement. There are several algorithms designed to execute this method Ristic et al.(2004), Doucet (1998). Each one has some advantages but also introduces a weakness, so the choice is problem-specific. The most common choice due to its straightforward implementation is the SIR (Sequential Importance Resampling) filter, also called bootstrap Ristic et al.(2004).

The generic structure of a particle filter consists in the generation of a high number of samples from an importance probability density that should correctly describe the probability density of the state at instant *k* given the measurement and the set of particles at instant *k-1*. Then, each particle is weighed according to the measurement probability density, and the set of particles and weights at instant *k* should provide an estimate of the state *a posteriori* probability density. Thus, one can generate the state estimate by, for example, computing the mean of the estimated *a posteriori* state probability density, which provides an approximation to the minimum mean square error state estimate.

The importance probability density choice is crucial. However, the optimal one, which minimizes the importance weights variance, is available only in restrictive problems Ristic et al.(2004). In most cases, a widely used approximation is expressed in Eq. 18 Ristic et al.(2004), Doucet (1998):

$$q(\mathbf{x}_k \mid \mathbf{p}_{k-1}^i, \mathbf{y}_k) = p(\mathbf{x}_k \mid \mathbf{p}_{k-1}^i) \qquad i = 1,2,\ldots,N \tag{18}$$

where $q(.)$ is the importance probability density and $\mathbf{p}_{k-1}^i$ denotes the *i*-th particle at instant *k-1*. Sampling from this p.d.f is carried out by sampling from the p.d.f. of the model noise and applying the nonlinear model equations, i.e. Eqs. 16, using $\mathbf{p}_{k-1}^i$ as the previous state.

Particle filters also need to perform a resampling stage, which has been devised to avoid wasting computational resources on low-weight, unimportant, particles. Basically, it eliminates particles with low weight and multiplies particles with higher weight. There is a myriad of algorithms for the resampling stage. This work has used the systematic resample described in Ristic et al.(2004) with a slightly modification proposed in Doucet (1998). The original SIR algorithm performs resampling at every iteration, but it is only needed when the particles become severely degraded, i.e., only few particles presenting a significant normalized weight. Therefore, the resampling stage is performed only when index N_{eff} shown in Eq. 19 is smaller than a threshold named N_{thres}.

$$N_{eff} = \frac{1}{\sum_{i=1}^{N} (w_k^i)^2}, \quad 1 \le N_{eff} \le N \tag{19}$$

where N is the number of particles and w_k^i denotes the normalized weight of the *i*-th particle at instant *k*. For this work, N_{thres} has been set to $0.9 \cdot N$.

Finally the SIR algorithm at instant *k* for this problem can be written:

$$[\{\mathbf{p}_k^i, w_k^i\}_{i=1}^N] = \mathrm{SIR}[\{\mathbf{p}_{k-1}^i, w_{k-1}^i\}_{i=1}^N, \mathbf{y}_k]$$

- FOR i = 1 : N
  - Draw $\mathbf{v} \sim N(0, \mathbf{Q})$ (sampling from model noise p.d.f)
  - $\mathbf{p}_k^i = \mathbf{p}_{k-1}^i + f(\mathbf{p}_{k-1}^i, \mathbf{u}_{k-1}) + \mathbf{v}$
  - Calculate $\tilde{w}_k^i = p(\mathbf{y}_k \mid \mathbf{p}_k^i)$
- END FOR
- Normalize weights generating $[\{w_k^i\}_{i=1}^N]$

- Calculate N_{eff}
- IF $N_{eff} < N_{thres}$
  - $[\{\mathbf{p}_k^i, w_k^i\}_{i=1}^N]$=RESAMPLE$[\{\mathbf{p}_k^i, w_k^i\}_{i=1}^N]$
- END IF
- Estimate the state with empirical mean: $\hat{\mathbf{x}}_{k|k} = \sum_{i=1}^{N} w_k^i . \mathbf{p}_k^i$

END

where $p(\mathbf{y}_k \mid \mathbf{p}_k^i)$ is a Gaussian p.d.f evaluated at $\mathbf{y}_k$ with mean $\mathbf{h}(\mathbf{p}_k^i)$, and covariance $\mathbf{R}$.

For this work, the above algorithm had an acceptable performance at steady-state with a low number of particles. However, it always diverges when a disturbance is applied even with a high number of particles. It is caused by a known phenomenon called particles collapse Ristic et al.(2004). The model noise here is very low, so the particles tend to stay in a very close state-space region, which decrease the diversity. When a disturbance occurs, the weights become very small and numerical round-off often set them all to 0. At this point, the filter should be fully reinitialized to amplify the diversity, but it was found that this process severely degraded estimation performance. The literature provides a set of modified algorithms that tries to minimize particle collapse of SIR filter Ristic et al.(2004). Here, it has been used the regularized particle filter (RPF) to circumvent the problem. This approach jitters the particles at every resampling stage, which helps to increase the diversity. The new algorithm for this stage is:

- IF $N_{eff} < N_{thres}$
  - Calculate empirical covariance $\mathbf{S}_k$ of $[\{\mathbf{p}_k^i, w_k^i\}_{i=1}^N]$
  - Compute $\mathbf{D}_k$ such that $\mathbf{D}_k \mathbf{D}_k^T = \mathbf{S}_k$ using, for example, the lower Cholesky decomposition.
  - $[\{\mathbf{p}_k^i, w_k^i\}_{i=1}^N]$=RESAMPLE$[\{\mathbf{p}_k^i, w_k^i\}_{i=1}^N]$
  - FOR i = 1 : N
    - Draw $\varepsilon \sim N(0,1)$
    - $\mathbf{p}_k^i = \mathbf{p}_k^i + h_{opt} . \mathbf{D}_k . \varepsilon$
  - END FOR
- END IF

where h_{opt} is calculated as in Eq. 20 Ristic et al.(2004):

$$h_{opt} = \left[\frac{4}{nx+2} \right]^{\frac{1}{nx+4}} . N^{-\frac{1}{nx+4}} \tag{20}$$

where nx is the number of state vector components.

It is well known that the number of particles for a good estimation quality increases exponentially with the number of states. So, for the RPF implementation, the eighth state has been suppressed from the filter and the integration of pointing error about Z-axis has been performed with the rectangle rule using ψ angle estimates from the filter.

5 Simulations

A. Parameters

The simulations have been carried out using the same parameters from Chagas and Waldmann (2010). Ground-truth inertia matrices of the table and reaction wheel have included inertia products to account for a residual mass unbalance. Equation 21 shows the measurement noise covariance matrix used in all filters:

$$\mathbf{R} = diag(\sigma_{accel.}^2 / 9{,}81 \quad \sigma_{accel.}^2 / 9{,}81 \quad 4.\sigma_{mag}^2 / 500 \quad \sigma_{tac}^2) \tag{21}$$

where *diag*(.) means a diagonal matrix. Magnetometer noise is not Gaussian on account of the measurements mathematical manipulation. This can be seen analyzing Eq. 3. Nevertheless, it has been verified by an *ad-hoc* procedure that its probability density is very similar to a Gaussian with zero mean and the variance described in Eq. 21.

The model noise covariance needed to be separately tuned for each filter to avoid divergence during the simulation. The selected values for the EKF, UKF and RPF are in Eqs. 22:

$$\mathbf{Q}_{ext} = 0{,}5.\mathbf{I}_{8x8} \qquad \mathbf{Q}_{usc} = 0{,}45.\mathbf{I}_{8x8}$$
$$\mathbf{Q}_{rpg} = \left[5 \quad 5 \quad 5 \quad 100 \quad 100 \quad 100 \quad 1000\right]10^{-4} \tag{22}$$

The number of particles in the RPF has been set to 2.10^6.

B. Filter Performance

Two metrics have been defined to gauge filter performance. The first computes the rotation angle about the

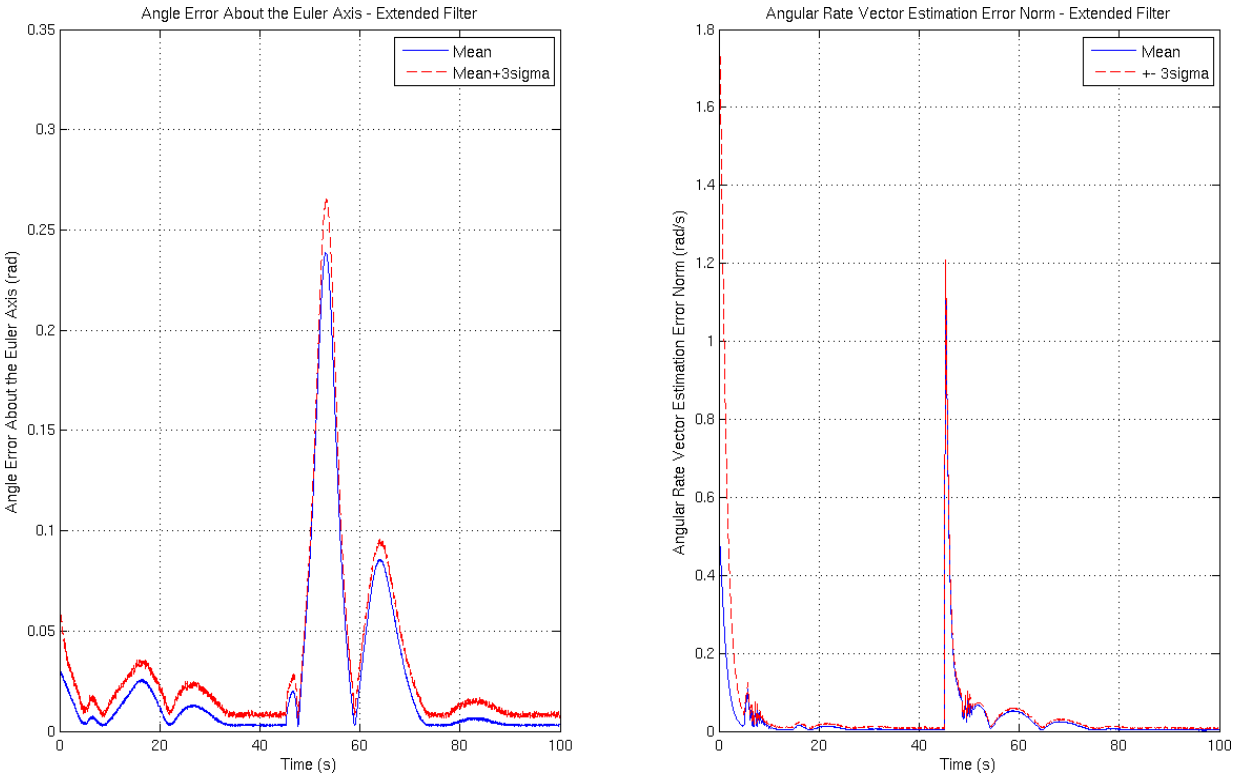

Figure 4. Scenario 01 results – Extended Kalman Filter

Euler axis that is related to the attitude estimation error at each iteration as in Eq. 23. It has been used to ascertain the attitude estimation accuracy of each filter. The second computes the norm of the angular rate vector estimation error at each iteration as in Eq. 24.

$$\Theta_{error,k} = a\cos\left(\frac{1}{2}.trace(\hat{\mathbf{D}}_{d,k|k}^{b,T}.\mathbf{D}_d^b) - \frac{1}{2}\right) \tag{23}$$

$$\omega_{\text{error,k}} = \sqrt{(\hat{\omega}_{\text{k|k}} - \omega_{\text{k}})^{\text{t}}(\hat{\omega}_{\text{k|k}} - \omega_{\text{k}})} \tag{24}$$

where $\hat{\mathbf{D}}_{d,k|k}^b$ and $\mathbf{D}_d^b$ are the estimated and ground-truth direction cosine matrices, respectively, at instant *k* that rotate from the reference coordinate frame to the table coordinate frame, and $\hat{\boldsymbol{\omega}}_{k|k}$ and $\boldsymbol{\omega}_k$ are the estimated and ground-truth table angular rate vectors, respectively, at instant *k*. Both metrics have been computed at each

iteration over a large number *M* of Monte Carlo simulations. At the end, the mean and standard deviation have been computed.

C. Results

For the EKF and UKF simulations, 100 Monte Carlo simulations have been carried out in a time interval from 0s to 100s with a filter step of 0,01s. For RPF, only 30 Monte Carlo simulations have been performed and the step has been set to 0,05s because of time constraints. The initial ground-truth state vector has been kept fixed and given in Eq. 25 with SI units. For the simulations, the filter initial estimate and the particles have been set equal to the initial ground-truth state plus a random vector in which each component was a random Gaussian variable with zero mean and variance 0.1.

$$\mathbf{x}_0 = \begin{bmatrix} 25 & -30 & 20 & 3 & -3 & -2 & 0 & 0 \end{bmatrix} \pi / 180 \, [SI] \tag{25}$$

An unexpected, deterministic disturbance torque has been applied at *t=45s* with *0.3s* duration to investigate filter robustness and convergence. The disturbance torque vector is given in Eq. 26.

$$\mathbf{T}_{d,b} = \begin{bmatrix} -0{,}7 & 0{,}7 & 0{,}3 \end{bmatrix}^T \quad N.m \tag{26}$$

The first scenario used the EKF as the estimator. The results are plotted in Fig. 4. The second scenario used the UKF and the results are shown in Fig. 5. The third scenario used the RPF and the results are shown in Fig. 6. The mean filter algorithm simulation time in the first approach was 4.8136s with a standard deviation of 1.096s. The mean filter algorithm simulation time in the second approach was 9.6889s with a standard deviation of 0.8601s.

The third approach had a mean filter algorithm simulation time of $3.5518 \cdot 10^4$ s with a standard deviation of 119.8495s. Finally, estimation error means of all scenarios are plotted together in Fig. 7 for comparison. Figure 7 shows that RPF yielded more accurate estimates after the unexpected disturbance, followed by the UKF and EKF. It also presented improved convergence rate with respective to the EKF and UKF approaches. At its peak in attitude estimation error, the RPF error was only 17% of the EKF error and 25% of the UKF error. Moreover, the RPF converged to steady-state estimation accuracy almost 10s in advance of the other two filters.

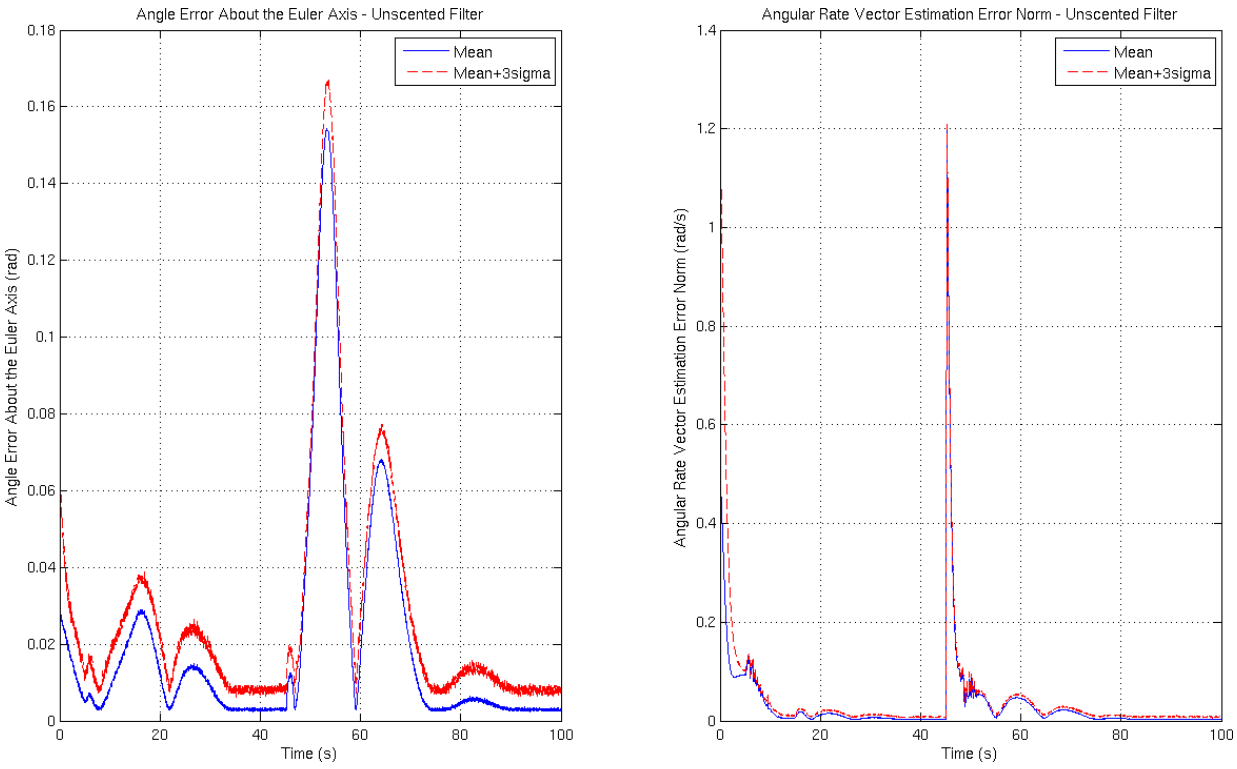

Figure 5. Scenario 02 results – Unscented Kalman Filter

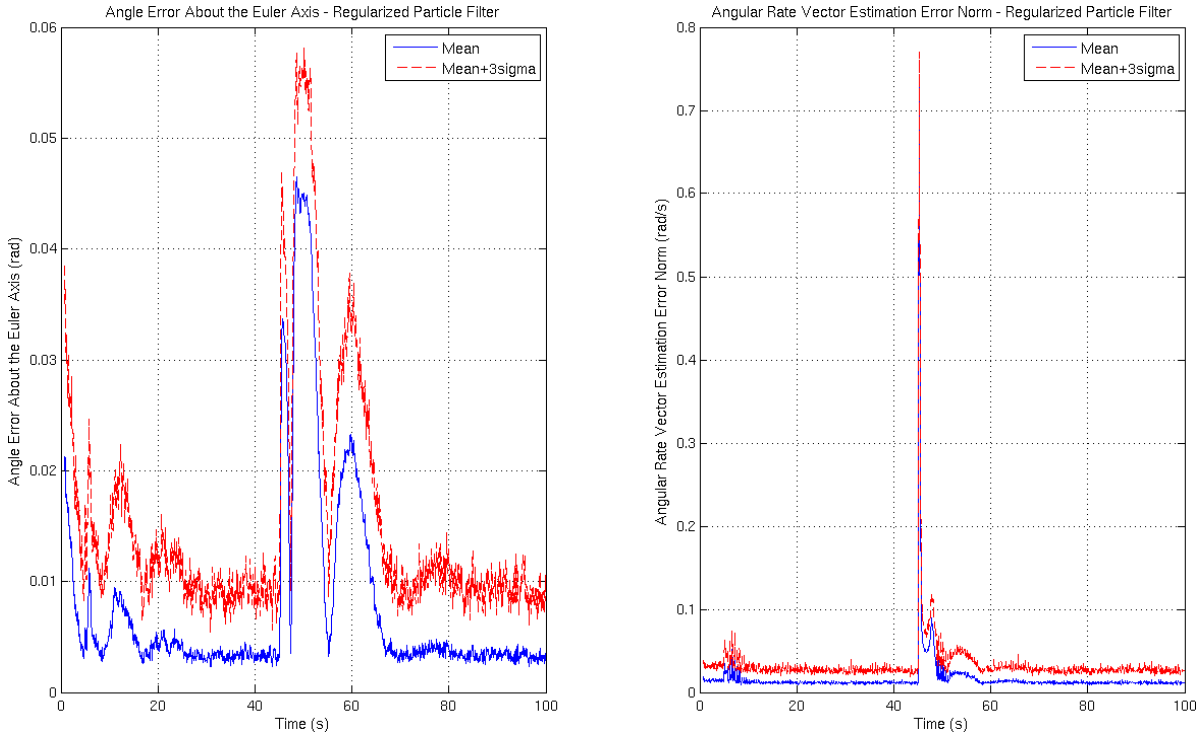

Figure 6. Scenario 03 results – Regularized Particle Filter

From Fig. 7, one also concludes that the estimation accuracy in steady-state is quite similar in all three approaches. This is a limitation set by measurement noise, model mismatch, and sensors biases. Note that the RPF yields the angular rate estimation accuracy slightly degraded relative to the EKF and UKF. It may have been caused by insufficient number of particles, or just because the RPF sample time is five times higher than EKF and UKF.

4 Conclusions

Three nonlinear estimation techniques have been investigated in a simulated air-suspended table with a straightforward, linearized state feedback law for attitude control. The table uses two pneumatic actuators for alignment with the local horizontal plane, and one reaction wheel for azimuth alignment. The sensors consist of two accelerometers to estimate the local gravity vector direction, and two magnetometers - one on board the table, and the other fixed to the reference coordinate frame to provide azimuth alignment about the local vertical.

The nonlinear system dynamics have been linearized about the desired state, thus decoupling horizontal and vertical dynamics. The state feedback control has successfully allocated the closed-loop poles in the desired locations for each of the decoupled dynamics.

Three scenarios have been simulated with an unexpected deterministic torque disturbance. The first one used an extended Kalman Filter (EKF), the second used an unscented Kalman filter (UKF), whereas the third used the regularized particle filter (RPF).

When the unexpected disturbance occurred the EKF approach resulted in the fastest algorithm, with the most degraded estimation accuracy though. The UKF produced twice the computational burden with respect to the EKF, but the attitude estimation error about the residual misalignment Euler vector was significantly lower. The RPF presented a superb gain of accuracy relative to the other algorithms, but at the price of a much heavier computational load. With respect to angular rate estimation, all three approaches yielded very similar accuracy.

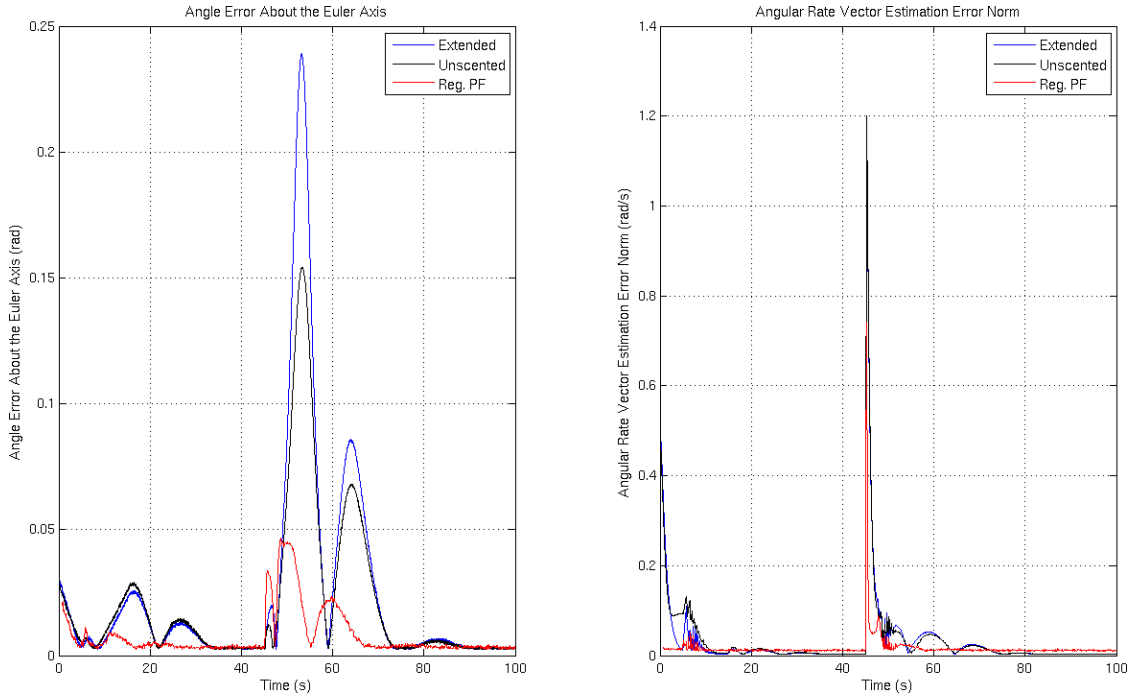

Figure 7. Comparison between the three scenarios

For a practical implementation on board a satellite, the RPF is not feasible with present microelectronics. Since the modeled dynamics has seven state components to be estimated, the resulting computational load is almost impractical even in desktop computers. The UKF should be a good compromise for on-board implementation in case the attitude estimation and control system design needs an estimator with lower sensitivity to a torque disturbance in comparison with the EKF. With the purpose of reducing computational workload, Chagas and Waldmann (2010) suggested the use of a hybrid approach, in which resorts some kind of fault detection mechanism should indicate the occurrence of a disturbance and thus switch from the EKF in steady-state to the UKF in the unexpected situation. Nevertheless, some implementation issues have arisen with the UKF Chagas and Waldmann (2010) involving the computation of the square root of the estimation error covariance matrix for generating the σ-points. Sometimes the positive-semi-definiteness of this matrix is lost because of numerical errors, hence rendering the UKF likely to be unfeasible for such a real-time, critical application.

Finally, investigation of Monte Carlo-based methods for attitude estimation and control are bound to carry on. Even if actual implementations appear today to be faraway into the future due to limited computational throughput, the multi-core processors technology is advancing at a fast pace. Today, it remains a challenge to fulfill the potential of such technology and parallelize the computations involved in particle filtering for use in embedded systems with complex, nonlinear high order dynamics.

5 Acknowledgments

The authors acknowledge the support provided by project FINEP/CTA/INPE SIA (Sistemas Inerciais para Aplicação Aeroespacial). We also thank INPE researchers Dr. Valdemir Carrara and Dr. Helio Koiti Kuga, for the information, data and help provided.

6 References

A., Doucet, On Sequential Simulated-Based Methods for Bayesian Filtering. Technical report CUED/F-INFENG/TR.310, 1998.

B., Ristic, S., Arulampalam, N., Gordon, Beyond the Kalman filter: particle filters for tracking applications. Ed. Artech House, Boston, United States of America, 2004.

M., J., Sidi, Spacecraft Dynamics and Control: a practical engineering approach. Ed. Cambridge University Press, New York, United States of America, 1997.

R., A., J., Chagas, J., Waldmann, Nonlinear Filtering in a Simulated Three-Axis Testbed for Satellite Attitude Estimation and Control. In Proc. IV Congresso Nacional de Engenharia Mecânica, Campina Grande, Paraíba, Brasil, 2010.

S., Särkkä, On Unscented Kalman Filtering for State Estimation of Continuous-Time Nonlinear Systems, In IEEE Transactions on Automatic Control, Vol. 52, No. 9, pp. 1631-1641.

V., Carrara, P., G., Milani, Controle de uma Mesa de Mancal a Ar de um Eixo Equipada com Giroscópio e Roda de Reação, In Proc. V Simpósio Brasileiro de Engenharia Inercial, Vol. 31, pp. 97-102.