Seven State Kalman Filtering for LEO Microsatellite Attitude Determination

A.M. SI MOHAMMED\textsuperscript{a}, M. BENYETTOU\textsuperscript{b},
A. BOUDJEMAI\textsuperscript{a}, Y. HASHIDA\textsuperscript{c},
\textsuperscript{a}Division de Mécanique Spatiale
Centre National des Techniques Spatiales
1, Avenue de la Palestine, Arzew, Algeria
\texttt{Arezki_s@yahoo.fr, A_boudjemai@yahoo.fr}

\textsuperscript{b}Université des Sciences et de la Technologie «Mohamed Boudiaf »
Laboratoire de Modélisation et Simulation, Oran, Algeria
\texttt{Med_Benyettou@yahoo.fr}

\textsuperscript{c}Surrey Satellite Technology Limited
Guilford, Surrey GU2 7XH, United Kingdom
\texttt{Y.Hashida@sstl.co.uk}

\textit{Abstract:} The attitude determination and control subsystem (ADCS) requirement of payload on the Algerian satellite is stringent enough to demand some type of data processing in order to meet attitude determination requirements. This paper details one data filter, the Kalman filter, and more specifically, the 7-state Kalman filter. For the 7-state filter, the state vector defines not only the attitude of the satellite, but also the rates at which the attitude is changing. This paper examines the seven state Kalman filter from a theoretical standpoint before examining practical implementation.

\textit{Keywords:} LEO, Microsatellite, Attitude, Kalman, Filtering.

\section{Introduction}

The proposed microsatellite is 3-axis stabilized, using a pitch momentum wheel and yaw reaction wheel, with dual redundant 3-axis magnetorquers. A gravity gradient boom is employed to provide a high degree of platform stability. Two vector magnetometers and four dual sun sensors are carried in order to determine the attitude [1]. As part of the ADCS task, a Kalman filter will be implemented to process sensor attitude data. Generally speaking, the Kalman filter is a recursive optimization algorithm that generates an estimate based upon potentially noisy observation data. At the most basic level, the Kalman filter is fundamentally an optimization problem that can be applied across many disciplines to predict the behaviour of systems. In astronautics, the Kalman filter is often implemented to simplify and expedite the ADCS task [6].

The Kalman filter is ideal for processing large amounts of data in this fashion. Attitude determination presents another difficulty that is addressed by the Kalman filter. In the attitude determination task, three independent reference parameters are needed to determine attitude. Each vector measurement provided by the satellite sensors yields two reference parameters. Therefore, the typical requirement for three-axis attitude determination is two vector measurements. Attitude determination with two measurements is over determined while attitude determination with one parameter is underdetermined. The Kalman filter can solve either the over determined or the underdetermined cases, allowing three-axis attitude determination with either both sun sensors and magnetometers or solely magnetometers. The Kalman filter is able to solve the underdetermined case because an on-board orbit propagation model is part of the filter. This filter utilizes this model to allow three-axis attitude determination with only one vector measurement. In short, the robustness of the Kalman filter enables it to be applied to the ADCS task [6].

For ADCS, Kalman filtering involves propagation of the satellite attitude and covariance matrices using both Euler’s moment equations and a basic knowledge of the disturbance torques acting upon the satellite. Subsequent to this propagation, the Kalman filter adjusts the propagated attitude and covariance matrices based upon the measurement vector(s). Because the attitude motion of the proposed microsatellite will be nonlinear, extended Kalman filter (EKF) will be necessary to accommodate nonlinearities. This paper will detail the various step to design the seven state Kalman filter.
2 System Equation:
A Kalman filter is an optimal, recursive, data processing algorithm [5], [7], [8] and [9] all address Kalman filtering for spacecraft attitude estimation. The 7-state EKF state vector is comprised of the four-element quaternion attitude vector combined with the three-element body rates vector, with respect to the inertial frame. The state vector to be estimated is 7 dimensional such that
\[
x = [q \omega_B]^T = [q_1 q_2 q_3 q_4 \omega_x \omega_y \omega_z]^T (1)
\]
From [2], [3], [4], [11] and [13] the system equation is given as follows
\[
\begin{bmatrix}
\dot{q}_1 \\
\dot{q}_2 \\
\dot{q}_3 \\
\dot{q}_4
\end{bmatrix} = \frac{1}{2} \begin{bmatrix}
0 & -o_{oz} & o_{oy} & o_{ox} \\
-o_{oz} & 0 & -o_{ox} & o_{oy} \\
o_{oy} & o_{ox} & 0 & -o_{oz} \\
o_{ox} & -o_{oy} & o_{oz} & 0
\end{bmatrix} \begin{bmatrix}
q_1 \\
q_2 \\
q_3 \\
q_4
\end{bmatrix} + \begin{bmatrix}
h_x \\
h_y \\
h_z
\end{bmatrix} \omega - w
\]
\[
\dot{\omega} = \begin{bmatrix}
\frac{N_x^{MT}}{I_x} = \alpha(3o_{o2}a_{23}a_{33} - o_yo_z) - \frac{h_x}{I_x} \\
\frac{N_y^{MT}}{I_y} = \beta(3o_{o2}a_{23}a_{33} - o_xo_y) - \frac{h_y}{I_y} \\
\frac{N_z^{MT}}{I_z} = \gamma(3o_{o2}a_{23}a_{33} - o_yo_x) - \frac{h_z}{I_z}
\end{bmatrix} + w
\]
(2)

3 State Transition Matrix:
The state transition matrix is defined by [13]:
\[
\Phi \approx I_{7 \times 7} + \begin{bmatrix}
\frac{\partial \hat{q}}{\partial q_1} \\
\frac{\partial \hat{q}}{\partial q_2} \\
\frac{\partial \hat{q}}{\partial q_3} \\
\frac{\partial \hat{q}}{\partial q_4}
\end{bmatrix}\Delta t (3)
\]
\[
\Phi \approx I_{7 \times 7} + \begin{bmatrix}
F_{11} \\
F_{12}
\end{bmatrix} \Delta t (4)
\]

3.1 (4x4) Matrix F11 Computation:
From [4], [13]
\[
\dot{q} = \frac{1}{2} \Omega q = \frac{1}{2} \Lambda(\omega_B - \omega_o) (5)
\]
The partial derivative of \(\dot{q}\) with respect to each \(q_i\) becomes:
\[
\frac{\partial \dot{q}}{\partial q_1} = \frac{1}{2} \Omega \frac{\partial q}{\partial q_1} + \frac{1}{2} \Lambda \frac{\partial \omega_B}{\partial q_1} q (6)
\]
\[
\frac{\partial \dot{q}}{\partial q_i} = \frac{1}{2} \Omega \frac{\partial q}{\partial q_i} - \frac{1}{2} \Lambda \frac{\partial \omega_B}{\partial q_i} \omega_0 (7)
\]
Then
\[
F_{11} = \frac{1}{2} \Omega + \begin{bmatrix}
\zeta_1 & \zeta_2 & \zeta_3 & \zeta_4
\end{bmatrix} (8)
\]
Where
\[
\zeta_i = -\frac{1}{2} \Lambda \frac{\partial \omega}{\partial q_i} \omega_0 (9)
\]
We find easily
\[
\begin{bmatrix}
\zeta_1 \\
\zeta_2 \\
\zeta_3 \\
\zeta_4
\end{bmatrix} = \begin{bmatrix}
q_1 q_1 \\
q_2 q_2 \\
q_3 q_3 \\
q_4 q_4
\end{bmatrix} (10.a)
\]
\[
\begin{bmatrix}
\zeta_1 \\
\zeta_2 \\
\zeta_3 \\
\zeta_4
\end{bmatrix} = \begin{bmatrix}
q_1 q_1 - q_2 q_2 \\
q_2 q_2 - q_3 q_3 \\
q_3 q_3 - q_4 q_4
\end{bmatrix} (10.b)
\]
\[
\begin{bmatrix}
\zeta_1 \\
\zeta_2 \\
\zeta_3 \\
\zeta_4
\end{bmatrix} = \begin{bmatrix}
1 - q_1^2 \\
-2q_1 q_2 \\
-q_1 q_3 \\
-q_1 q_4
\end{bmatrix} (10.c)
\]
\[
\begin{bmatrix}
\zeta_1 \\
\zeta_2 \\
\zeta_3 \\
\zeta_4
\end{bmatrix} = \begin{bmatrix}
q_1 q_1 \\
q_2 q_2 \\
q_3 q_3 \\
q_4 q_4
\end{bmatrix} (10.d)
\]

\[\text{ISSN: 1790-5117} \quad 152 \quad \text{ISBN: 978-960-474-157-1}\]
3.2 (4x3) Matrix F_{12} Computation:
The partial derivative can be found as follows
\[
\frac{\partial \mathbf{q}}{\partial \mathbf{w}_B} = \frac{1}{2} \Lambda \frac{\partial}{\partial \mathbf{w}_B} (\mathbf{w}_B - \Lambda \omega_0) = \frac{1}{2} \Lambda
\]  
(11)

Then
\[
\mathbf{F}_{12} = \frac{1}{2} \begin{bmatrix}
q_4 & -q_3 & q_2 \\
q_3 & q_4 & -q_1 \\
-q_2 & q_1 & q_4 \\
-q_1 & -q_2 & -q_3
\end{bmatrix}
\]  
(12)

3.3 (3x4) Matrix F_{21} Computation:
The partial derivative can be found as follows
\[
\mathbf{F}_{21} = \begin{bmatrix}
\lambda_1 & \lambda_2 & \lambda_3 & \lambda_4
\end{bmatrix}
\]  
(13)

Where
\[
\lambda_1 = \begin{bmatrix}
6 \omega_0^2 \alpha (-a_{23} q_1 + a_{33} q_4)
\end{bmatrix}
\]  
(14.a)

\[
\lambda_2 = \begin{bmatrix}
6 \omega_0^2 \beta (+a_{33} q_3 - a_{13} q_1)
\end{bmatrix}
\]  
(14.b)

\[
\lambda_3 = \begin{bmatrix}
6 \omega_0^2 \alpha (+a_{23} q_3 + a_{33} q_2)
\end{bmatrix}
\]  
(14.c)

\[
\lambda_4 = \begin{bmatrix}
6 \omega_0^2 \beta (-a_{33} q_2 + a_{13} q_1)
\end{bmatrix}
\]  
(14.d)

3.4 (3x3) Matrix F_{22} Computation:
The partial derivative can be found as follows
\[
\mathbf{F}_{22} = \begin{bmatrix}
\eta_1 & \eta_2 & \eta_3
\end{bmatrix}
\]  
(15)

Where
\[
\eta_1 = \begin{bmatrix}
0 \\
-\beta \omega_z + \frac{h_z}{I_y} \\
-h_y & \frac{h_z}{I_z}
\end{bmatrix}
\]  
(16.a)

4 Observation Equation:
The sensor such as magnetometer and sun sensor observation model is given as follows [3], [5]
\[
\hat{\mathbf{S}}_B = \hat{\mathbf{S}}_{LO} + \mathbf{n}
\]  
(17)

Where \( \mathbf{n} \) is a zero mean measurement noise vector and \( \hat{\mathbf{S}} \) is a normalised sensor observation vector with respect to the body axis and orbit coordinate.

The sensor observation (3x7) matrix \( \mathbf{H} \) is given as follows
\[
\mathbf{H} = \begin{bmatrix}
\partial \hat{\mathbf{S}}_B / \partial (q_1 q_2 q_3 q_4) \\
\partial \hat{\mathbf{S}}_B / \partial \omega_x \\
\partial \hat{\mathbf{S}}_B / \partial \omega_y \\
\partial \hat{\mathbf{S}}_B / \partial \omega_z
\end{bmatrix}
\]  
(18)

The partial derivative can be found as follows
\[
\mathbf{H} = \begin{bmatrix}
\mu_1 & \mu_2 & \mu_3 & \mathbf{O}_{3x3}
\end{bmatrix}
\]  
(19)

Where
\[
\mu_i = \frac{\partial \hat{\mathbf{S}}_{LO}}{\partial q_i} \quad 1 \leq i \leq 3
\]
\( \mathbf{O}_{3x3} \) is a zero 3x3 matrix

Therefore
\[
\frac{\partial \mathbf{A}}{\partial q_1} = \begin{bmatrix}
q_1 & q_2 & q_3 \\
q_2 & -q_1 & q_4 \\
q_3 & -q_4 & -q_1
\end{bmatrix}
\]  
(20.a)

\[
\frac{\partial \mathbf{A}}{\partial q_2} = \begin{bmatrix}
-q_2 & q_1 & -q_4 \\
q_1 & q_2 & q_3 \\
q_4 & q_3 & -q_2
\end{bmatrix}
\]  
(20.b)

\[
\frac{\partial \mathbf{A}}{\partial q_3} = \begin{bmatrix}
-q_3 & q_4 & q_1 \\
-q_4 & -q_3 & q_2 \\
q_1 & q_2 & q_3
\end{bmatrix}
\]  
(20.c)

\[
\frac{\partial \mathbf{A}}{\partial q_4} = \begin{bmatrix}
q_4 & q_3 & -q_2 \\
-q_3 & q_4 & q_1 \\
q_2 & -q_1 & q_4
\end{bmatrix}
\]  
(20.d)
4 Process and Observation Noise Matrix:
A process noise covariance will compensate the uncertainty of the model used in the system equation. The process covariance matrix $Q_{id}$ defined by [3], [13]

$$Q_k = \int_{t_k}^{t_{k+1}} \Phi(\Delta t, u) E[w(u)w(v)]^T \Phi(\Delta t, v) dt$$

For our application, it is assumed that only the angular velocity terms have process noise, then the (7x7) $E$ matrix is given by

$$E_{22} = \begin{bmatrix} \sigma_z^2 \delta(u-v) & 0 & 0 \\ 0 & \sigma_y^2 \delta(u-v) & 0 \\ 0 & 0 & \sigma_y^2 \delta(u-v) \end{bmatrix}$$

Where $\delta$ is Dirac delta function.

Then

$$Q = Q_1 \Delta t + Q_2 \frac{(\Delta t)^2}{2} + Q_3 \frac{(\Delta t)^3}{3}$$

5 7-State Kalman Filter Mathematical Process:
From [3], [6] and [12] the 7-state extended Kalman filter Algorithm is given as follows

- Propagation Cycle
  - Covariance Propagation
    $$P_{k+1}^- = \Phi_k \Phi_k^T + Q_k$$
    $$\Phi \approx I_{7 \times 7} + F\Delta t$$

- Propagation State
    $$\hat{x}_{k+1} = \hat{x}_k + \int_{t_k}^{t_{k+1}} f(\{x(t), t\}) dt$$

Specifically

$$\hat{q}_{k+1} = \hat{q}_k + \int_{t_k}^{t_{k+1}} (Qq) dt$$

$$\hat{\omega}_{k+1} = \hat{\omega}_k + \int_{t_k}^{t_{k+1}} \left[ N - \omega_x (I_{\omega} + h) - \dot{h} \right] dt$$

- Correction Cycle
  - Compute Observation Matrix
    $$H = \begin{bmatrix} \frac{\partial \hat{S}_B}{\partial q_1} & \frac{\partial \hat{S}_B}{\partial q_2} & \frac{\partial \hat{S}_B}{\partial q_3} \end{bmatrix}$$

  - Compute Kalman Gain $K_k$
    $$K_k = P_k^{-1} H_k (H_k P_k^{-1} H_k^T + R_k)^{-1}$$

  - Update Estimate with Measurement $z_k$
    $$\hat{x}_k = \hat{x}_k + K_k b_k$$

  - Compute Error Covariance for Updated Estimate
    $$P_k = (I - K_k H_k) P_k^-$$

Where

- $\hat{x}_k$: The computed state estimate for this step (using measurement data from this step);
- $\hat{x}_k$: The predicted state estimate from the previous step;
- $x$: The actual value of the state. We never know this value but seek to estimate it;
- $P_k$: The predicted error covariance (7x7) matrix from the previous step;
- $P_k$: The computed error covariance (7x7) matrix;
- $H_k$: The measurement noise (3x7) matrix;
- $b_k$: The measurement residual vector.
6 Simulations Results:
The following initialization parameters were utilized for 7-state analysis

Orbit:
Inclination [degree] : 98
Altitude [km] : 680

Initial Uncertainty of Rate
(Initial Covariance Matrix P):
q1 = q2 = q3 = q4 : (0.3)^2
ω_{ox} = ω_{oy} = ω_{oz} [degree/sec] : (0.5)^2

Process Noise Intensity of Euler Angle and Rate
(System Noise Covariance Matrix Q)
σ_x = σ_y = σ_z [degree/sec] : 510^5
Q_2 = Q_3 : O_{7x7}

Inertial Tensor (Satellite configuration I)
I [kgm^2] : diag [159.2 158.2 4.91]^t

Measurement Error Variance
(Measurement Noise Covariance Matrix R)
Magnetometer measurement error variance in X/Y/Z axis [microTesla]^2 : (0.3)^2
Sun sensor measurement error variance in X/Y/Z axis [degree]^2 : (0.1)^2

Miscellaneous
Simulation time [orbit] : 2
Integration step [sec] : 1
Sampling time [sec] : 5

4.1 Dynamic Simulations
The initial attitude error is 70° roll and the pitch, 40° for the yaw.
Figures 1 and 3 shows the true attitude and the performance of the EKF 7 state simulation. Figures 4 and 6 shows the state error estimation. It is clear that the attitude errors (or residuals) converge to zero quickly.
It can be seen from Table 1 and Table 2 that the magnitude of the RMS error estimation indicates that the angular error is approximately 0.086 deg during magnetometer measurement and 0.06 deg during hybrid measurement. The rate error is about 0.11 mdeg/sec during magnetometer measurement and 0.12 mdeg/sec during hybrid measurement.

Table 1: State error compilation using magnetometer measurement

<table>
<thead>
<tr>
<th></th>
<th>RMS using magnetometer measurement</th>
</tr>
</thead>
<tbody>
<tr>
<td>Roll [deg]</td>
<td>0.04</td>
</tr>
<tr>
<td>Pitch [deg]</td>
<td>0.07</td>
</tr>
<tr>
<td>Yaw [deg]</td>
<td>0.03</td>
</tr>
<tr>
<td>$\omega_x$ [deg/sec]</td>
<td>7.76 *10^{-5}</td>
</tr>
<tr>
<td>$\omega_y$ [deg/sec]</td>
<td>7.92 *10^{-5}</td>
</tr>
<tr>
<td>$\omega_z$ [deg/sec]</td>
<td>6.21*10^{-5}</td>
</tr>
<tr>
<td>Magnitude of error</td>
<td>0.086</td>
</tr>
<tr>
<td>Angles [deg]</td>
<td>1.11*10^{-4}</td>
</tr>
<tr>
<td>Angular velocity [deg/sec]</td>
<td></td>
</tr>
</tbody>
</table>

Table 2: State error compilation using hybrid measurement

<table>
<thead>
<tr>
<th></th>
<th>RMS using hybrid measurement</th>
</tr>
</thead>
<tbody>
<tr>
<td>Roll [deg]</td>
<td>0.03</td>
</tr>
<tr>
<td>Pitch [deg]</td>
<td>0.05</td>
</tr>
<tr>
<td>Yaw [deg]</td>
<td>0.02</td>
</tr>
<tr>
<td>$\omega_x$ [deg/sec]</td>
<td>9.13*10^{-5}</td>
</tr>
<tr>
<td>$\omega_y$ [deg/sec]</td>
<td>8.07*10^{-5}</td>
</tr>
<tr>
<td>$\omega_z$ [deg/sec]</td>
<td>7.98*10^{-6}</td>
</tr>
<tr>
<td>Magnitude of error</td>
<td>0.06</td>
</tr>
<tr>
<td>Angles [deg]</td>
<td>1.22*10^{-4}</td>
</tr>
<tr>
<td>Angular velocity [deg/sec]</td>
<td></td>
</tr>
</tbody>
</table>

7 Conclusion

This paper detailed Kalman filtering, and more specifically, the 7-state. The extended Kalman filter described in this paper was able to extract full attitude, body rate from noisy vector measurements for a LEO 3-axis stabilized. The magnitude of the RMS error results indicates that the angular error is approximately 0.086 deg and the rate error is about 0.12 mdeg/sec. These results fit within the bounds specified by the derived attitude requirements.

A low cost method of full satellite state determination such as 7-state was proposed to be used for LEO microsatellite.
References:


