Automatica 57 (2015) 105–112

Contents lists available at ScienceDirect

Automatica

journal homepage: www.elsevier.com/locate/automatica

Brief paper

Computationally simple sub-optimal filtering for spacecraft motion estimation[☆]

Anton H.J. de Ruiter[1]

Department of Aerospace Engineering, Ryerson University, 350 Victoria Street, Toronto, ON, Canada, M5B 2K3

ARTICLE INFO

Article history:
Received 20 October 2014
Received in revised form
5 February 2015
Accepted 5 April 2015
Available online 16 May 2015

Keywords:
Space vehicles
Navigation
Kalman filters
Extended Kalman filters
Computational simplicity

ABSTRACT

This paper presents a computationally simple near-optimal filter for spacecraft motion estimation. This is particularly important in applications where the computational resources are very limited, such as in cube-satellite and nano-satellite missions. The proposed filter consists of two scalar gains, and has analytically guaranteed performance under given bounds on the process and measurement noise covariances. Unlike the Kalman filter or its variants, there is no associated covariance propagation. Favorable performance of the presented filter, compared with a conventional extended Kalman filter, is demonstrated via a hardware-in-the-loop simulation of a dual spacecraft formation navigation problem.

1. Introduction

There are increasing efforts to make spacecraft more autonomous in general. An enabling technology for this is the ability to estimate the spacecraft motion onboard in real-time. In particular, much of the current effort is focused on absolute and relative orbital motion estimation for spacecraft formations (Ardaens, D'Amico, & Cropp, 2013; D'Amico & Montenbruck, 2010; Ebinuma, Bishop, & Lightsey, 2003; Ebinuma, Montenbruck, & Lightsey, 2002; Eyer & Damaren, 2009; Gill, D'Amico, & Montenbruck, 2007). The benchmark algorithm for on-board motion estimation is the Kalman filter and its variants (Crassidis & Junkins, 2012; El-Sheimy, Nassar, Shin, & Niu, 2006).

A common present day assumption is that computational power is no longer a limitation. Consequently, there is a general trend toward increasing complexity of on-board state estimation algorithms, with the goal of improving both estimation robustness and performance. However, there is still real value in developing and utilizing computationally simple algorithms when the spacecraft performance requirements allow for it. First, a number of past space-system failures can be attributed to software errors (Harland & Lorenz, 2005; Tafazoli, 2009). With the ever increasing complexity of flight software, these failures are becoming more difficult to predict, detect, isolate and mitigate. Second, simple algorithms are more amenable to analysis with the ability to make conclusions based on rigorous mathematical arguments, while this is more difficult or impossible to do with more complex algorithms. For example, with some highly complex implementations of the extended Kalman filter (D'Amico & Montenbruck, 2010) or its variants such as the unscented Kalman filter (Crassidis & Junkins, 2012; Wolfe, Speyer, Hwang, Lee, & Lee, 2007), there are no analytical guarantees of filter stability or consistency. As such, engineers must embark on extensive numerical simulation campaigns in order to assess filter convergence and steady-state performance. Finally, in very small satellite applications (for example, cube-satellites and nano-satellites), the onboard processors that are used are still typically quite primitive, with limited computational resources.

In this paper, a novel computationally simple continuous-discrete filter is developed for spacecraft orbital motion estimation based on position measurements. The filter contains two scalar gains, and requires no covariance propagation. Each gain is separated into two parts: a time-varying transient part, and a constant steady-state part. The transient part of the gains is applied for a pre-determined period of time, with the objective of achieving rapid filter convergence similar to the Kalman filter. After the transient period, the gains are switched to the constant steady-state part, and they provide the long-term filter stability and performance. The filter is similar to the Kalman filter in that it contains a linear correction term based on the measurement

[☆] The material in this paper was not presented at any conference. This paper was recommended for publication in revised form by Associate Editor Andrey V. Savkin under the direction of Editor Ian R. Petersen.

E-mail address: aderuiter@ryerson.ca.

[1] Tel.: +1 416 979 5000x4878; fax: +1 416 979 5056.

http://dx.doi.org/10.1016/j.automatica.2015.04.018
0005-1098/

innovation. However, unlike the Kalman filter, the correction is applied directly inside the state estimate propagation step, rather than after the propagation step, resulting in an a-priori filter (a one-step ahead predictor). This key difference allows the filter's stability and steady-state performance to be analyzed, and analytically guaranteed steady-state performance bounds are obtained. The effectiveness of the proposed filter is demonstrated by applying it to a hardware-in-the-loop simulation for GPS-based relative navigation for a close spacecraft formation.

2. Problem formulation

In this paper, the $n \times n$ identity matrix will be denoted by $\mathbf{I}_n$, the $n \times n$ matrix of zeros by $\mathbf{0}_n$, and $\|\mathbf{x}\|$ denotes the Euclidean 2-norm of the vector $\mathbf{x} \in \mathbb{R}^n$, while $\|\mathbf{X}\|$ denotes the induced 2-norm of the matrix $\mathbf{X} \in \mathbb{R}^{n\times m}$.

It is assumed that position measurements are available (for example, from a GPS receiver), of the form

$$\mathbf{r}^m(t_k) = \mathbf{r}(t_k) + \mathbf{n}_k, \tag{1}$$

expressed in some reference frame of interest $\mathcal{F}_x$ (common choices are inertial or Earth-fixed frames). The vectors $\mathbf{r}^m(t_k)$ and $\mathbf{r}(t_k)$ represent respectively the measured and true spacecraft position vectors at time instant t_k, and $\mathbf{n}_k$ is a zero-mean white noise sequence with covariance $E\{\mathbf{n}_k\mathbf{n}_j^T\} = \mathbf{R}_k\delta_{kj}$, where $E\{\cdot\}$ denotes the expectation operator, δ_{kj} is the discrete delta function and $\mathbf{R}_k$ is a symmetric positive definite matrix.

In $\mathcal{F}_x$ coordinates, the spacecraft translational equations of motion are given by

$$\begin{aligned}\dot{\mathbf{r}}(t) &= \mathbf{v}(t) - \boldsymbol{\omega}_x^\times(t)\mathbf{r}(t),\\ \dot{\mathbf{v}}(t) &= \mathbf{a}_g(\mathbf{r}(t), t) - \boldsymbol{\omega}_x^\times(t)\mathbf{v}(t) + \mathbf{a}_{ng}(t) + \mathbf{w}_a(t),\end{aligned} \tag{2}$$

where $\mathbf{v}(t)$ is the spacecraft inertial velocity, $\boldsymbol{\omega}_x(t)$ is the known inertial angular velocity of $\mathcal{F}_x$, $\mathbf{a}_g(\mathbf{r}(t), t)$ is the spacecraft gravitational acceleration, $\mathbf{a}_{ng}(t)$ is the modeled spacecraft non-gravitational acceleration, $\mathbf{w}_a(t)$ represents un-modeled accelerations and modeling errors. For any $\mathbf{a} \in \mathbb{R}^3$, the matrix $\mathbf{a}^\times \in \mathbb{R}^{3\times3}$ is the skew symmetric matrix such that $\mathbf{a}^\times\mathbf{b}$ gives the cross-product between $\mathbf{a} \in \mathbb{R}^3$ and any other $\mathbf{b} \in \mathbb{R}^3$. It is assumed that $\mathbf{a}_g(\mathbf{r}, t)$ is continuously differentiable in $\mathbf{r}$. Furthermore, the gravitational acceleration is derivable from a potential function $\phi_g(\mathbf{r}, t)$ as $\mathbf{a}_g(\mathbf{r}, t)^T = \nabla\phi_g(\mathbf{r}, t)$ (Vallado, 2004). Consequently, the Jacobian $\partial\mathbf{a}_g(\mathbf{r}, t)/\partial\mathbf{r}$ is symmetric. It is assumed that $\mathbf{w}_a(t)$ is a zero-mean white noise process with covariance $E\{\mathbf{w}_a(t)\mathbf{w}_a(\tau)^T\} = \bar{\mathbf{Q}}_t\delta(t-\tau)$ where $\bar{\mathbf{Q}}(t)$ is a symmetric positive semi-definite matrix, and $\delta(t-\tau)$ is the Dirac delta function. The following filter structure is now imposed

$$\begin{aligned}\dot{\hat{\mathbf{r}}}(t) &= \hat{\mathbf{v}}(t) - \boldsymbol{\omega}_x^\times(t)\hat{\mathbf{r}}(t) + \mathbf{u}_r(t),\\ \dot{\hat{\mathbf{v}}}(t) &= \mathbf{a}_g(\hat{\mathbf{r}}(t), t) - \boldsymbol{\omega}_x^\times(t)\hat{\mathbf{v}}(t) + \mathbf{a}_{ng}(t) + \mathbf{u}_v(t),\end{aligned} \tag{3}$$

where $\hat{\mathbf{r}}$ and $\hat{\mathbf{v}}$ are the estimates of $\mathbf{r}$ and $\mathbf{v}$, respectively, and $\mathbf{u}_r$ and $\mathbf{u}_v$ are control-like inputs, which are yet to be defined. The position and velocity estimation errors are defined as $\tilde{\mathbf{r}}(t) = \mathbf{r}(t) - \hat{\mathbf{r}}(t)$ and $\tilde{\mathbf{v}}(t) = \mathbf{v}(t) - \hat{\mathbf{v}}(t)$ respectively. Using Eq. (1), the difference between the measured and estimated position becomes

$$\mathbf{r}^m(t_k) - \hat{\mathbf{r}}(t_k) = \tilde{\mathbf{r}}(t_k) + \mathbf{n}_k, \tag{4}$$

which may be used as an input for the filter in (3). It is assumed that $\tilde{\mathbf{r}}$ and $\tilde{\mathbf{v}}$ are small, such that their dynamics are well described by linearizing their dynamics about the estimates, $\hat{\mathbf{r}}(t)$ and $\hat{\mathbf{v}}(t)$. Using Eqs. (2) and (3), the linearized error dynamics take the form

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \tilde{\mathbf{A}}(t)\mathbf{x}(t) - \mathbf{u}(t) + \mathbf{B}\mathbf{w}_a(t), \tag{5}$$

together with a discrete-time measurement equation from (4) of the form

$$\mathbf{y}_k = \mathbf{H}\mathbf{x}(t_k) + \mathbf{n}_k, \tag{6}$$

where

$$\mathbf{x} \triangleq \begin{bmatrix}\tilde{\mathbf{r}}\\ \tilde{\mathbf{v}}\end{bmatrix}, \qquad \mathbf{y}_k = \mathbf{r}^m(t_k) - \hat{\mathbf{r}}(t_k), \qquad \mathbf{A} = \begin{bmatrix}\mathbf{0}_3 & \mathbf{I}_3\\ \mathbf{0}_3 & \mathbf{0}_3\end{bmatrix},$$

$$\tilde{\mathbf{A}}(t) = \begin{bmatrix}-\boldsymbol{\omega}_x^\times(t) & \mathbf{0}_3\\ \partial\mathbf{a}_g(t) & -\boldsymbol{\omega}_x^\times(t)\end{bmatrix}, \qquad \partial\mathbf{a}_g(t) \triangleq \partial\mathbf{a}_g(\hat{\mathbf{r}}(t), t)/\partial\mathbf{r},$$

$$\mathbf{B} = \begin{bmatrix}\mathbf{0}_3 & \mathbf{I}_3\end{bmatrix}^T, \qquad \mathbf{u} = \begin{bmatrix}\mathbf{u}_r^T & \mathbf{u}_v^T\end{bmatrix}^T, \qquad \mathbf{H} = \begin{bmatrix}\mathbf{I}_3 & \mathbf{0}_3\end{bmatrix}.$$

From this point on, for a time-dependent vector or matrix $\mathbf{X}(t)$, we will denote $\mathbf{X}(t_k) = \mathbf{X}_k$. It is assumed that the sample period $T = t_{k+1} - t_k$ is short enough such that the linearized error dynamics in (5) can be discretized as

$$\mathbf{x}_{k+1} = e^{(\mathbf{A}+\tilde{\mathbf{A}}_k)T}\mathbf{x}_k - \int_{t_k}^{t_{k+1}} e^{(\mathbf{A}+\tilde{\mathbf{A}}_k)(t_{k+1}-\tau)}\mathbf{u}(\tau)d\tau + \mathbf{w}_k, \tag{7}$$

where $\mathbf{w}_k$ is a zero-mean white noise sequence with covariance

$$\mathbf{Q}_k = \int_{t_k}^{t_{k+1}} e^{(\mathbf{A}+\tilde{\mathbf{A}}_k)(t_{k+1}-\tau)}\mathbf{B}\bar{\mathbf{Q}}_\tau\mathbf{B}^T e^{(\mathbf{A}+\tilde{\mathbf{A}}_k)^T(t_{k+1}-\tau)}d\tau. \tag{8}$$

In typical problems involving spacecraft, $\tilde{\mathbf{A}}_k$ is small compared with $\mathbf{A}$, and it may be viewed as a perturbation to $\mathbf{A}$. One readily finds that

$$e^{\mathbf{A}t} = \begin{bmatrix}\mathbf{I}_3 & t\mathbf{I}_3\\ \mathbf{0}_3 & \mathbf{I}_3\end{bmatrix}. \tag{9}$$

Using (9), if $\bar{\mathbf{Q}}_t = q(t)\mathbf{I}_3$ (i.e. isotropic) where $q(t) \geq 0$, then in the unperturbed case (i.e. $\tilde{\mathbf{A}}_k = \mathbf{0}$) the discrete process noise in (8) takes the form

$$\mathbf{Q}_k = \int_{t_k}^{t_{k+1}} \begin{bmatrix}(t_{k+1}-\tau)^2\mathbf{I}_3 & (t_{k+1}-\tau)\mathbf{I}_3\\ (t_{k+1}-\tau)\mathbf{I}_3 & \mathbf{I}_3\end{bmatrix} q(\tau)d\tau. \tag{10}$$

Considering the matrix exponential expansion for $\exp[(\mathbf{A}+\tilde{\mathbf{A}}_k)t]$, and keeping only first order terms in $\tilde{\mathbf{A}}_k$ gives

$$e^{(\mathbf{A}+\tilde{\mathbf{A}}(t_k))t} \approx e^{\mathbf{A}t} + \begin{bmatrix}\dfrac{\partial\mathbf{a}_{g,k}t^2}{2} & \dfrac{\partial\mathbf{a}_{g,k}t^3}{6}\\ \partial\mathbf{a}_{g,k}t & \dfrac{\partial\mathbf{a}_{g,k}t^2}{2}\end{bmatrix} - \begin{bmatrix}\boldsymbol{\omega}_{x,k}^\times t & \boldsymbol{\omega}_{x,k}^\times t^2\\ \mathbf{0}_3 & \boldsymbol{\omega}_{x,k}^\times t\end{bmatrix}. \tag{11}$$

The control term in (7) will now be examined. Since the measurements in (6) are available only at the sample times, the control-like input in Eqs. (3) and (5) is implemented through a zero-order hold

$$\mathbf{u}(t) = \bar{\mathbf{u}}_k, \quad t_k \leq t < t_{k+1}, \tag{12}$$

for some yet to be chosen $\bar{\mathbf{u}}_k$. Temporarily neglecting the perturbing terms in (11), one obtains

$$\int_{t_k}^{t_{k+1}} e^{\mathbf{A}(t_{k+1}-\tau)}\mathbf{u}(\tau)d\tau = \begin{bmatrix}T\mathbf{I}_3 & (T^2/2)\mathbf{I}_3\\ \mathbf{0}_3 & T\mathbf{I}_3\end{bmatrix}\bar{\mathbf{u}}_k =: \breve{\mathbf{u}}_k. \tag{13}$$

Using Eqs. (9) and (13), the ideal discrete system corresponding to (7) becomes

$$\mathbf{x}_{k+1} = \boldsymbol{\Phi}\mathbf{x}_k - \breve{\mathbf{u}}_k + \mathbf{w}_k, \tag{14}$$

where $\boldsymbol{\Phi} \triangleq \exp[\mathbf{A}T]$, and $\breve{\mathbf{u}}_k$ is a discrete control-like input, which is yet to be defined. Eq. (13) is readily inverted to obtain

$$\bar{\mathbf{u}}_k = \begin{bmatrix}(1/T)\mathbf{I}_3 & -(1/2)\mathbf{I}_3\\ \mathbf{0}_3 & (1/T)\mathbf{I}_3\end{bmatrix}\breve{\mathbf{u}}_k, \tag{15}$$