COMPUTATION OF EXTENDED ROBUST KALMAN FILTER FOR REAL-TIME ATTITUDE AND POSITION ESTIMATION

Gaurav R. Yengera*, Roberto S. Inoue†, Mundla Narasimhappa‡, Marco H. Terra‡

* *Department of Electrical Engineering, Indian Institute of Technology (Banaras Hindu University), Varanasi and also with Department of Electrical Engineering, University of São Paulo at São Carlos, São Paulo, Brazil*

† *Department of Electrical Engineering, Federal University of São Carlos at São Carlos, São Paulo, Brazil*

‡ *Department of Electrical Engineering, University of São Paulo at São Carlos, São Paulo, Brazil*

Emails: g.yengera@gmail.com, rsinoue@ufscar.br, mr.narasimha08@gmail.com, terra@sc.usp.br

Abstract— This paper deals with the implementation of the extended robust Kalman filter (ERKF) which was developed considering uncertainties in the parameter matrices of the underlying state-space model. A key contribution of this work is the demonstration of a method for real-time computation of the filter on parallel computing devices. The solution of the filter is expressed as a set of simultaneous linear equations, which can then be evaluated based on QR decomposition using Givens rotation. This paper also presents the application of the ERKF in the development of an attitude and position reference system for a cargo transport vehicle. This work concludes by analyzing the performance of the ERKF and verifying the validity of the Givens rotation method.

Keywords— Extended Robust Kalman Filter, Givens Rotation, QR Decomposition, Attitude and Position Estimation, Uncertainties.

1 Introduction

The Kalman filter (Kalman, 1960; Anderson and Moore, 1979; Kailath et al., 2000) has played an important role in solving estimation problems appearing in navigation, economics, communications, control and other areas. The real-time computation of the Kalman filter has been an important and recurring feature in many of its applications. A fundamental assumption in the Kalman filter is that the underlying state-space model is accurate and does not contain uncertainties. When this condition is violated, the performance of the filter could deteriorate drastically (Sayed, 2001). As a result robust estimation is necessary in several real world applications and certain algorithms have been developed for this purpose. However, very little research has been carried out on the real time implementation of these robust estimation algorithms on parallel computing devices such as FPGAs and GPUs.

A detailed explanation of the robust optimal filtering approach utilized in the ERKF has been presented in (Terra et al., 2011; Ishihara et al., 2015; Inoue et al., 2016). Also its advantages over other robust filtering approaches have been discussed in those papers. To summarize the important features of the ERKF: it does not require any auxiliary parameter to be tuned while for linear systems, stability and convergence are guaranteed for all steady-state estimates. Thereby this filtering approach is preferable for real time implementation as no offline computations are necessary. Additionally the ERKF assumes the existence of uncertainties in all parameter matrices of the state-space model.

FPGAs have been a popular choice for the real time implementation of the Kalman filter. The most significant prior work on floating-point FPGA implementations have been developed based on direct mapping of the equations on the FPGA which either involves explicit matrix inversion, see for instance (Bonato et al., 2007; Lee and Salcic, 1997), or representation of the equations in Schur complement form and then applying Fadeev's algorithm as done in (Chen and Guo, 2005).

This paper will focus on the implementation of the ERKF presented in (Inoue et al., 2016). For the evaluation of the ERKF, a matrix of particularly large dimensions needs to be inverted. Hence, the matrix inversion approach would be computationally intensive and not ideal for real time applications. It will be shown that the proposed method is computationally more efficient than the conventional matrix inversion approach. As compared to Fadeev's algorithm, it is more straightforward to evaluate the filter as a set of simultaneous linear equations by applying QR decomposition using Givens rotation, see (Francis, 1962; Givens, 1958), followed by back substitution to obtain desired state vector and state covariance matrix.

The application of the ERKF in an attitude and position reference system for a cargo transport vehicle utilizing a global positioning system (GPS) along with an inertial measurement unit (IMU) is presented. To accommodate for the

lower measurement update frequency of the GPS, as discussed in (Farrel, 2008), an inertial navigation system based on attitude estimates is used to estimate position in the absence of GPS measurements. At the same time the IMU measurements contain residual errors even after calibration and the ERKF is used to ensure that state estimates are robust to such errors or uncertainties. In (Inoue et al., 2016) the performance improvement of the ERKF over the extended Kalman filter in the presence of uncertainties has been discussed. The role of the ERKF is especially crucial when working with low-cost IMUs which tend to possess substantial uncertainties (Xing and Gebre-Egziabher, 2008). A method to model the uncertainties present in the IMU rate gyros and accelerometers measurements, and incorporate them into the ERKF has been shown. This application is of general importance for developing more sophisticated navigation systems and it also highlights the necessity for real time computation of the ERKF. This paper concludes by using sensor data collected from this experimental setup to verify the Givens rotation based computation approach.

The organization of the paper is as follows: the extended robust Kalman filter is presented in Section 2. In Section 3, the proposed algorithm and its computational complexity are discussed. The vehicle attitude and position estimation system is presented in Section 4. In Section 5, the experimental setup and results are discussed. Finally, Section 6 presents the conclusion of the paper.

2 Extended Robust Kalman Filter

For the development of the robust Kalman filter, the underlying state-space model of the dynamic discrete-time system is modified to incorporate parametric uncertainties:

$$\begin{aligned} \mathbf{x}_{k+1} &= (F_k + \delta F_k)\,\mathbf{x}_k + (G_k + \delta G_k)\,\mathbf{w}_k, \\ \mathbf{z}_k &= (H_k + \delta H_k)\,\mathbf{x}_k + (K_k + \delta K_k)\,\mathbf{v}_k, \end{aligned} \tag{1}$$

for $k \geq 0$, where $\mathbf{x}_k \in \mathbb{R}^n$ is the state vector, $\mathbf{z}_k \in \mathbb{R}^p$ is the measurement vector, $\mathbf{w}_k$ and $\mathbf{v}_k$ are the noise vectors corresponding to the state update and measurement equations respectively. Typically, $\mathbf{x}_0$, $\mathbf{w}_k$ and $\mathbf{v}_k$ are considered as mutually independent zero-mean Gaussian random variables with respective variances $\mathbb{E}\{\mathbf{x}_0\mathbf{x}_0^T\} = \Pi_0 \geq 0$, $\mathbb{E}\{\mathbf{w}_k\mathbf{w}_k^T\} = Q_k \geq 0$ and $\mathbb{E}\{\mathbf{v}_k\mathbf{v}_k^T\} = R_k \geq 0$. $F_k \in \mathbb{R}^{n\times n}$, $G_k \in \mathbb{R}^{n\times m}$, $H_k \in \mathbb{R}^{p\times n}$, $K_k \in \mathbb{R}^{p\times m}$ are nominal parameter matrices, and $\delta F_k \in \mathbb{R}^{n\times n}$, $\delta G_k \in \mathbb{R}^{n\times m}$, $\delta H_k \in \mathbb{R}^{p\times n}$, $\delta K_k \in \mathbb{R}^{p\times m}$ are uncertainty matrices which are modeled as:

$$\begin{aligned} \begin{bmatrix} \delta F_k & \delta G_k \end{bmatrix} &= M_{1_k}\Delta_1 \begin{bmatrix} N_{F_k} & N_{G_k} \end{bmatrix} \\ \begin{bmatrix} \delta H_k & \delta K_k \end{bmatrix} &= M_{2_k}\Delta_2 \begin{bmatrix} N_{H_k} & N_{K_k} \end{bmatrix} \end{aligned} \tag{2}$$

where $||\Delta_1|| < 1$ and $||\Delta_2|| < 1$ are arbitrary contractions. Matrices M_{1_k}, M_{2_k}, N_{F_k}, N_{G_k}, N_{H_k}, and N_{K_k} are known *a-priori*.

The ERKF is developed considering the solution of the following unconstrained optimization problem:

$$\min_{\mathbf{x}_k,\mathbf{x}_{k+1}} \max_{\delta_k} \{\mathcal{J}_k^{\mu}(\mathbf{x}_k, \mathbf{w}_k, \mathbf{v}_k, \mathbf{x}_{k+1}, \delta_k)\}, \tag{3}$$

where $\delta_k := \{\delta F_k, \delta G_k, \delta K_k, \delta H_k\}$. The cost function $\mathcal{J}_k^{\mu}(\mathbf{x}_k, \mathbf{w}_k, \mathbf{v}_k, \mathbf{x}_{k+1}, \delta_k)$ is given in (4), in which $F_{\delta,k} = F_k + \delta F_k$, $G_{\delta,k} = G_k + \delta G_k$, $H_{\delta,k} = H_k + \delta H_k$ and $K_{\delta,k} = K_k + \delta K_k$. The detailed derivation of the filter is presented in (Inoue et al., 2016).

$$\begin{aligned} \mathcal{J}_k^{\mu}(\mathbf{x}_k, \mathbf{w}_k, \mathbf{v}_k, \mathbf{x}_{k+1}, \delta_k) := & \begin{bmatrix} \mathbf{x}_k - \widehat{\mathbf{x}}_{k|k-1} \\ \mathbf{w}_k \\ \mathbf{v}_k \\ \mathbf{x}_k \\ \mathbf{x}_{k+1} \end{bmatrix}^T \begin{bmatrix} P_{k|k-1}^{-1} & 0 & 0 & 0 & 0 \\ 0 & Q_k^{-1} & 0 & 0 & 0 \\ 0 & 0 & R_k^{-1} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \mathbf{x}_k - \widehat{\mathbf{x}}_{k|k-1} \\ \mathbf{w}_k \\ \mathbf{v}_k \\ \mathbf{x}_k \\ \mathbf{x}_{k+1} \end{bmatrix} + \\ & \left(\begin{bmatrix} F_{\delta,k} & G_{\delta,k} & 0 & 0 & -I \\ 0 & 0 & K_{\delta,k} & H_{\delta,k} & 0 \end{bmatrix} \begin{bmatrix} \mathbf{x}_k - \widehat{\mathbf{x}}_{k|k-1} \\ \mathbf{w}_k \\ \mathbf{v}_k \\ \mathbf{x}_k \\ \mathbf{x}_{k+1} \end{bmatrix} - \begin{bmatrix} -F_{\delta,k}\widehat{\mathbf{x}}_{k|k-1} \\ \mathbf{z}_k \end{bmatrix} \right)^T \mu I \left(\bullet \right), \end{aligned} \tag{4}$$

where, the notation $Y^T XY = Y^T X(\cdot)$ is used for convenience and μ is a penalty parameter.

The ERKF is essentially in the form of a predicted estimator filter where the measurement update is computed first and the time update step later. Thereby the filter recursively computes $\widehat{\mathbf{x}}_{k+1|k}$ and $P_{k+1|k}$ from $\widehat{\mathbf{x}}_{k|k-1}$ and $P_{k|k-1}$ respectively.

The optimal estimates for linear systems are obtained by substituting $\mu \to \infty$. The recursive algorithm for obtaining the optimal filtered estimates is presented in Table 1.

The ERKF is applied to nonlinear system models by defining $\mathbf{b}_k = \begin{bmatrix} -F_k\widehat{\mathbf{x}}_{k|k-1} \\ \mathbf{z}_k - \mathbf{h}(\widehat{\mathbf{x}}_{k|k-1}) \end{bmatrix}$ and is used with linear system models by defining $\mathbf{b}_k = \begin{bmatrix} -F_k\widehat{\mathbf{x}}_{k|k-1} \\ \mathbf{z}_k - H_k\widehat{\mathbf{x}}_{k|k-1} \end{bmatrix}$.

3 Computation using Givens Rotation

The ERKF given in Table 1 can be rewritten as a linear system $A\mathbf{y} = \mathbf{b}$, as shown in (5). The μ and λ terms are values we are not interested in, while all the other terms are as defined in Table 1.

$$\underbrace{\begin{bmatrix} P_{k|k-1} & 0 & 0 & 0 & I & 0 & 0 \\ 0 & \mathcal{R}_k & 0 & 0 & 0 & I & 0 \\ 0 & 0 & 0 & 0 & \mathcal{F}_k & \mathcal{G}_k & \mathcal{E}_k \\ 0 & 0 & 0 & 0 & N_{\mathcal{F}_k} & N_{\mathcal{G}_k} & N_{\mathcal{E}_k} \\ I & 0 & \mathcal{F}_k^T & N_{\mathcal{F}_k}^T & 0 & 0 & 0 \\ 0 & I & \mathcal{G}_k^T & N_{\mathcal{G}_k}^T & 0 & 0 & 0 \\ 0 & 0 & \mathcal{E}_k^T & N_{\mathcal{E}_k}^T & 0 & 0 & 0 \end{bmatrix}}_{A_1} \underbrace{\begin{bmatrix} \lambda_1 & \mu_1 \\ \lambda_2 & \mu_2 \\ \lambda_3 & \mu_3 \\ \lambda_4 & \mu_4 \\ \widehat{\mathbf{x}}_{k|k} - \widehat{\mathbf{x}}_{k|k-1} & * \\ \widehat{\boldsymbol{\mu}}_{k|k} & * \\ \widehat{\mathbf{x}}_{k+1|k} & P_{k+1|k} \end{bmatrix}}_{\mathbf{y}_1} = \underbrace{\begin{bmatrix} 0 & 0 \\ 0 & 0 \\ \mathbf{b}_k & 0 \\ \mathbf{b}_{\mathbf{b}_k} & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & -I \end{bmatrix}}_{\mathbf{b}_1}, \tag{5}$$

To solve for only the required elements of matrix $\mathbf{y}$, i.e. state vector $\widehat{\mathbf{x}}_{k+1|k}$ and state covariance matrix $P_{k+1|k}$, each system of linear equations corresponding to an individual column of matrix $\mathbf{y}$ is evaluated one at a time using QR decomposition based on Givens rotation. And only the required values are calculated using back-substitution. The following equations describe the proposed solution:

$$A\mathbf{y}_l = \mathbf{b}_l, \tag{6}$$

Table 1: Recursive Algorithm for Extended Robust Kalman filter

Uncertain Model: Consider (1) with $\Pi_0 \succ 0$, $Q_k \succ 0$, and $R_k \succ 0$.

Step 0: (Initial Conditions) $P_{0|-1} = \Pi_0$, $\widehat{\mathbf{x}}_{0|-1} = 0$.

Step k: Given $\mathbf{z}_k$, update $\{\widehat{\mathbf{x}}_{k|k}\,;\,\widehat{\mathbf{x}}_{k+1|k}\,;\,P_{k+1|k}\}$ from $\{\mathbf{z}_k\,;\,\widehat{\mathbf{x}}_{k|k-1}\,;\,P_{k|k-1}\}$ as follows:

$$\begin{bmatrix} \widehat{\mathbf{x}}_{k|k} & * \\ \widehat{\boldsymbol{\nu}}_{k|k} & * \\ \widehat{\mathbf{x}}_{k+1|k} & P_{k+1|k} \end{bmatrix} = \begin{bmatrix} \widehat{\mathbf{x}}_{k|k-1} & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 & 0 & I & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & I & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & I \end{bmatrix}$$
$$\begin{bmatrix} P_{k|k-1} & 0 & 0 & 0 & I & 0 & 0 \\ 0 & \mathcal{R}_k & 0 & 0 & 0 & I & 0 \\ 0 & 0 & 0 & 0 & \mathcal{F}_k & \mathcal{G}_k & \mathcal{E}_k \\ 0 & 0 & 0 & 0 & N_{\mathcal{F}_k} & N_{\mathcal{G}_k} & N_{\mathcal{E}_k} \\ I & 0 & \mathcal{F}_k^T & N_{\mathcal{F}_k}^T & 0 & 0 & 0 \\ 0 & I & \mathcal{G}_k^T & N_{\mathcal{G}_k}^T & 0 & 0 & 0 \\ 0 & 0 & \mathcal{E}_k^T & N_{\mathcal{E}_k}^T & 0 & 0 & 0 \end{bmatrix}^{-1} \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ \mathbf{b}_k & 0 \\ N_{\mathbf{b}_k} & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & -I \end{bmatrix},$$
$$\widehat{\boldsymbol{\nu}}_{k|k} = \begin{bmatrix} \widehat{\mathbf{w}}_{k|k} \\ \widehat{\mathbf{v}}_{k|k} \end{bmatrix}, \mathcal{F}_k = \begin{bmatrix} F_k \\ H_k \end{bmatrix}, \mathcal{G}_k = \begin{bmatrix} G_k & 0 \\ 0 & K_k \end{bmatrix}, \mathcal{E}_k = \begin{bmatrix} -I \\ 0 \end{bmatrix},$$
$$N_{\mathcal{F}_k} = \begin{bmatrix} N_{F_k} \\ N_{H_k} \end{bmatrix}, N_{\mathcal{G}_k} = \begin{bmatrix} N_{G_k} & 0 \\ 0 & N_{K_k} \end{bmatrix}, N_{\mathcal{E}_k} = \begin{bmatrix} 0 \\ 0 \end{bmatrix},$$
$$\mathbf{b}_k = \begin{bmatrix} -F_k\widehat{\mathbf{x}}_{k|k-1} \\ \mathbf{z}_k - H_k\widehat{\mathbf{x}}_{k|k-1} \end{bmatrix}, N_{\mathbf{b}_k} = \begin{bmatrix} -N_{F_k}\widehat{\mathbf{x}}_{k|k-1} \\ -N_{H_k}\widehat{\mathbf{x}}_{k|k-1} \end{bmatrix}, \mathcal{R}_k = \begin{bmatrix} Q_k & 0 \\ 0 & R_k \end{bmatrix}$$

$$QR \times \mathbf{y}_l = b_l, \tag{7}$$

$$R \times \mathbf{y}_l = Q^{-1} \times b_l = Z_l, \tag{8}$$

where $\mathbf{y}_l$ and b_l are the lth columns of $\mathbf{y}$ and $\mathbf{b}$ respectively, and $A = QR$ is the result of QR decomposition of matrix A.

The matrices R and Z_l can be obtained by the following equation:

$$[R, Z_l] = \Theta[A, b_l] = \Theta M_l, \tag{9}$$

where Θ is any sequence of Givens rotations which result in the upper triangularization of matrix A. Since matrix R is a triangular matrix, the elements of vector $\mathbf{y}_l$, specifically the ones corresponding to state vector $\widehat{\mathbf{x}}_{k+1|k}$ or state covariance matrix $P_{k+1|k}$, can be calculated from (8) using back-substitution. All the elements of $\widehat{\mathbf{x}}_{k+1|k}$ and $P_{k+1|k}$ are obtained after (9) and (8) have been carried out for all the columns of $\mathbf{y}$.

The number of floating point operations (FLOPS) required by this method, where one FLOP is counted as any individual floating point operation; has been calculated referring to (Golub and Loan, 1996) and considering the following general case: $\mathbf{x} \in \mathbb{R}^{n\times 1}$, $P \in \mathbb{R}^{n\times n}$, $A \in \mathbb{R}^{m\times m}$. It can be noticed that in (5), the matrix A is a square matrix and $m \gg n$. Thereby, $\mathbf{y} \in \mathbb{R}^{m\times(n+1)}$, $\mathbf{b} \in \mathbb{R}^{m\times(n+1)}$ and $M_l \in \mathbb{R}^{m\times(m+1)}$.

Givens rotation, in (9), requires FLOPS of an order of magnitude equal to $3(m+1)^2(m-\frac{m+1}{3})$. Since $m \gg 1$, this can be approximated as $2m^3$. Back-substitution is applied in (8) to calculate the bottom n elements of vector $\mathbf{y}_l$ as only these correspond to elements of either $\widehat{\mathbf{x}}_{k+1|k}$ or $P_{k+1|k}$. Hence, the FLOPS required in (8) is of the order of magnitude n^2.

Remembering that (9) and (8) need to be carried out $n+1$ times, once for each column of matrix $\mathbf{y}$, the total number of FLOPS required is given by:

$$\text{Total FLOPS} = (n+1) \times (2m^3 + n^2) \propto 2nm^3. \tag{10}$$

For evaluating the ERKF directly as presented in Table 1, explicitly calculating the inverse of matrix A using Gaussian elimination would involve applying Gaussian elimination and back-substitution m times. The number of FLOPS required would be of the order of magnitude $\frac{2m^4}{3}$. To only obtain $\widehat{\mathbf{x}}_{k+1|k}$ and $P_{k+1|k}$ after having obtained the inverse, requires matrix multiplication considering the bottom n rows of A^{-1} alone. The number of FLOPS for the matrix multiplication step would be $2nm(n+1)$. The computationally intensive step in this method is the matrix inversion step.

Comparing the computational cost of the matrix inversion approach, $\frac{2m^4}{3}$, with (10) and noticing from the structure of matrix A that $n < \frac{m}{3}$; it can be concluded that the Givens rotation approach is computationally more efficient.

An important remark to be added here is that instead of the QR decomposition approach, the more efficient LU decomposition approach to solve the linear system given in (5) would appear to be a better choice. However, it was seen that the result of the LU decomposition method was unstable and did not converge with the results obtained from the matrix inversion approach. It can further be noticed that matrix A is sparse and that is the reason for Givens rotation being preferred over Householder reflections. Additionally, it is important to note that Givens rotation based QR decomposition easily lends itself to parallel implementations (Wang and Leeser, 2009). Hence, the computation speed can further be increased on a parallel computing device such as an FPGA.

4 Vehicle State Determination

In this section the vehicle state determination system (attitude, position and velocity) is presented. The system is composed of an IMU and a GPS module. The IMU is based on uncertain output models of rate gyros and accelerometers $\boldsymbol{\omega}_g$,

$\boldsymbol{a}_{\mathrm{a}}$. In Figure 1 the model of the IMU considers rate gyro bias $\boldsymbol{b}_{\mathrm{g}}$, accelerometer bias $\boldsymbol{b}_{\mathrm{a}}$, Gaussian white noise in the rate gyros and accelerometers, $\mathbf{w}_{\mathrm{g}}$ and $\mathbf{w}_{\mathrm{a}}$, respectively, and uncertain terms due to scale factor and axes misalignment of the rate gyros, accelerometers, $\delta\boldsymbol{\omega}_{\mathrm{g}}$ and $\delta\boldsymbol{a}_{\mathrm{a}}$, respectively. The IMU also measures the tilt angles ϕ_{IMU} and θ_{IMU}, as well as the yaw angle ψ_{IMU}. The GPS module provides geodetic position $\boldsymbol{p}_{GPS}$ and yaw angle ψ_{GPS}.

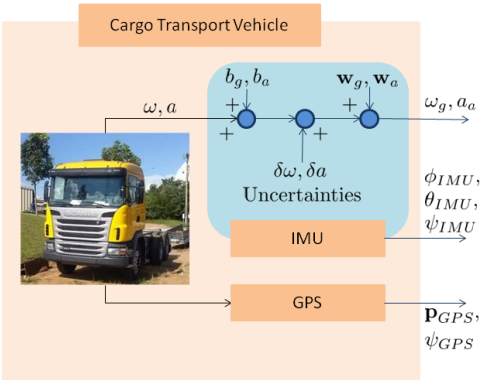

Figure 1: Cargo transport vehicle experimental setup.

The estimation of the state is split in two filters: (1) an attitude estimator and (2) a position estimator, in order to obtain a trade-off between accuracy and processing power (Bijker and Steyn, 2008). In Figure 2, the method of obtaining attitude and position estimates has been illustrated. This is essentially a pictorial representation of the systems described in Sections 4.1 and 4.2. Further, Figure 2 shows how the inertial navigation system updates position estimates in the absence of GPS measurements by integrating accelerometer readings. The states of the attitude and position system are represented by $\widehat{\mathbf{x}}^a$, and $\widehat{\mathbf{x}}^p$, respectively.

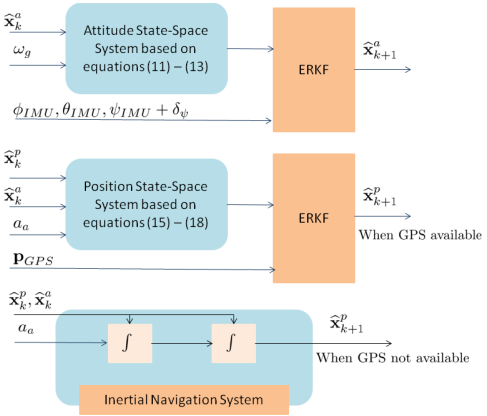

Figure 2: Estimation Diagram.

4.1 *Attitude system*

The dynamic equations of the attitude model are given by (Farrel, 2008; Kim, 2011):

$$[\dot{\phi} \;\; \dot{\theta} \;\; \dot{\psi}]^T = \Omega(\phi,\theta,\psi)[p \;\; q \;\; r]^T, \tag{11}$$

$$[p \;\; q \;\; r]^T = \boldsymbol{\omega}_{\mathrm{g}} + \delta\boldsymbol{\omega}_{\mathrm{g}} - \boldsymbol{b}_{\mathrm{g}} - \mathbf{w}_{\mathrm{g}}, \tag{12}$$

$$\dot{\boldsymbol{b}}_{\mathrm{g}} = -\frac{1}{\tau_{\mathrm{g}}}\boldsymbol{b}_{\mathrm{g}} + \mathbf{w}_{b_{\mathrm{g}}}, \tag{13}$$

where ϕ and θ are the tilt angles roll and pitch, respectively; ψ is the yaw angle, p, q and r are the angular velocities in the body frame; τ_{g} is the correlation time of the Gauss Markov process; $\mathbf{w}_{b_{\mathrm{g}}}$ is Gaussian white noise of the rate gyros bias; and $\Omega(\phi,\theta,\psi)$ is the transformation matrix between angular velocities, which is given by:

$$\Omega(\phi,\theta,\psi) = \begin{bmatrix} 1 & \sin\phi\tan\theta & \cos\phi\tan\theta \\ 0 & \cos\phi & -\sin\phi \\ 0 & \sin\phi\sec\theta & \cos\phi\sec\theta \end{bmatrix}. \tag{14}$$

The filters presented in this paper are based on discrete-time systems. In this regard, Equations (11) - (13) are represented in the form of (1) after being linearized and discretized considering a sample time T. The terms of (1) are chosen as; $\mathbf{x}^a = [\phi \;\; \theta \;\; \psi \;\; \boldsymbol{b}_{\mathrm{g}}^T]^T \in \mathbb{R}^{6\times1}$ is the state vector, $\mathbf{w}^a = [\mathbf{w}_{\mathrm{g}}^T \;\; \mathbf{w}_{b_{\mathrm{g}}}^T]^T \in \mathbb{R}^{6\times1}$ is the vector Gaussian process with zero mean and covariance Q^a, $\mathbf{z}^a = [\phi_{IMU} \;\; \theta_{IMU} \;\; \psi_{IMU} + \delta_\psi]^T \in \mathbb{R}^{3\times1}$ is the measurement vector, $\delta_\psi = \psi_{GPS} - \psi_{IMU}$ is the yaw error between GPS and IMU computed when GPS is available, $\mathbf{v}^a \in \mathbb{R}^{3\times1}$ is the vector Gaussian process with zero mean and covariance R^a of the measured angles in $\mathbf{z}^a$, F_k^a is the state transition matrix, G_k^a is the input noise matrix, and $H_k^a = [I_{3\times3} \;\; 0_{3\times3}]$ is the measurement matrix.

4.2 *Position system*

The dynamic equations of the position model are given by (Farrel, 2008; Bijker and Steyn, 2008):

$$[\dot{\lambda} \;\; \dot{\varphi} \;\; \dot{h}]^T = \Psi(\lambda,\varphi,h)[\upsilon_N \;\; \upsilon_E \;\; \upsilon_D]^T, \tag{15}$$

$$[\dot{\upsilon}_N \;\; \dot{\upsilon}_E \;\; \dot{\upsilon}_D]^T = \boldsymbol{g}_{\mathrm{e}} + A^T(\phi,\theta,\psi)\boldsymbol{a}, \tag{16}$$

$$\boldsymbol{a} = \boldsymbol{a}_{\mathrm{a}} + \delta\boldsymbol{a}_{\mathrm{a}} - \boldsymbol{b}_{\mathrm{a}} - \mathbf{w}_{\mathrm{a}}, \tag{17}$$

$$\dot{\boldsymbol{b}}_{\mathrm{a}} = -\frac{1}{\tau_{\mathrm{a}}}\boldsymbol{b}_{\mathrm{a}} + \mathbf{w}_{b_{\mathrm{a}}}, \tag{18}$$

where $\boldsymbol{p} = [\lambda \;\; \varphi \;\; h]^T$ are geodetic positions in the LLA (Latitude, Longitude and Altitude) frame ; $\boldsymbol{\upsilon} = [\upsilon_N \;\; \upsilon_E \;\; \upsilon_D]^T$ are the velocities in the NED (*North*, *East* and *Down*) frame; R_λ is the radius of meridian curvature at a given latitude; R_ϕ is the transverse radius of curvature, $\boldsymbol{g}_{\mathrm{e}}$ is the Earth's gravity vector; $\boldsymbol{a}$ is the actual linear acceleration; $A(\phi,\theta,\psi)$ is the rotation matrix from Inertia frame to Body frame; τ_{a} is the correlation

time of the Gauss Markov process; $\mathbf{w}_{b_a}$ is Gaussian white noise of the accelerometer bias; and $\Psi(\lambda, \varphi, h)$ is the transformation matrix between linear velocities, which is given by:

$$\Psi(\lambda,\varphi,h) = \begin{bmatrix} \frac{1}{R_\lambda + h} & 0 & 0 \\ 0 & \frac{1}{(R_\phi + h)cos\lambda} & 0 \\ 0 & 0 & -1 \end{bmatrix}. \quad (19)$$

Equations (15)-(18) are written in the state-space form of (1) after being linearized and discretized, such that state vector $\mathbf{x}^p = [\boldsymbol{p}^T \ \ \boldsymbol{v}^T \ \ \boldsymbol{b}_a^T]^T \in \mathbb{R}^{6\times 1}$, $\mathbf{w}^p = [\mathbf{w}_a^T \ \mathbf{w}_{b_a}^T]^T \in \mathbb{R}^{6\times 1}$ is the vector Gaussian process with zero mean and covariance Q^p, $\mathbf{z}^p = [\boldsymbol{p}_{GPS}^T]^T \in \mathbb{R}^{3\times 1}$ is the measurement vector, $\mathbf{v}^p \in \mathbb{R}^{3\times 1}$ is the vector Gaussian process with zero mean and covariance R^p of the measured position and velocity in $\mathbf{z}^p$, F_k^p is the state transition matrix, G_k^p is the input noise matrix, and $H_k^p = [I_{3\times 3} \ \ 0_{3\times 6}]$ is the measurement matrix.

5 Experimental Results

5.1 Description of Experimental Setup

An IMU and GPS were used to track the attitude and position of the cargo transport vehicle shown in Figure 1. The IMU used was the Xsens MTi-300-AHRS-2A5G4, which utilized MEMS based sensors. It comprised of a 3-axial accelerometer, 3-axial gyroscope and a 3-axial magnetometer. The update rate of the IMU was 400 Hz and it had an in-built algorithm which computed orientation, angular velocity and linear velocity from sensor readings. For this experimental setup, orientation was measured in terms of Euler angles and both the angular and linear velocities were measured about the body reference frame of the vehicle.

Septentrio AsteRx2eH PRO GPS was used in the cargo transport vehicle. The update rate of the GPS was 10 Hz and this was considerably slower than the IMU update rate. The measurements provided by the GPS were latitude, longitude, altitude, linear velocity as well as heading or yaw angle.

The weighting matrices Q^s and R^s for the ERKF were chosen based on the method described in (Xing and Gebre-Egziabher, 2008), where $s = a, \ p$. And the parameter matrices $N_{F_k}^s$, $N_{G_k}^s$, and $N_{H_k}^s$ are modeled in a manner to attenuate the uncertain terms $\delta\boldsymbol{\omega}_g$ presented in (12), $\delta\boldsymbol{a}_a$ presented in (17) and others sources of uncertainties occurring in the matrices F_k^s, G_k^s, and H_k^s. This is done by taking the average of each uncertain position present in rows i along the corresponding columns l of matrices F_k^s, G_k^s, and H_k^s; see (Inoue et al., 2016). The obtained matrices are as follows:

$$\begin{aligned} N_{F_k}^s &= 10^2 \begin{bmatrix} f_1^s & \dots & f_{n_s}^s \end{bmatrix}, \\ N_{G_k}^s &= 10^2 \begin{bmatrix} g_1^s & \dots & g_{m_s}^s \end{bmatrix}, \\ N_{H_k}^s &= \begin{bmatrix} 0 & \dots & 0 \end{bmatrix}, N_{K_k}^s = \begin{bmatrix} 0 & \dots & 0 \end{bmatrix}, \end{aligned} \quad (20)$$

where n_s is the number of variables in the state vector $\mathbf{x}^s$; m_s is the of variables in the noise vector $\mathbf{w}^s$; $n_a = 6$; $m_a = 6$; $n_p = 9$; $m_p = 6$; $f_l^s = \frac{\sum_{i=1}^{n_s} \overline{F}_k^s(i,l)}{n_s}$, $\overline{F}_k^s = F_k^s - I_{n_s \times n_s}$, for $l = 1, 2, .., n_s$; $g_l^s = \frac{\sum_{i=1}^{n_s} G_k^p(i,l)}{n_s}$, for $l = 1, \dots, m_s$.

5.2 Plot of Attitude and Position Estimates

In this section, the graphs for the attitude and position estimates are presented. Figure 5 shows the attitude estimates while Figure 6 shows the position estimates.

5.2.1 Attitude Estimates

The majority of variation in attitude is seen in the yaw angle as shown in Figure 5(c), while the total variation in roll, Figure 5(a), and pitch, Figure 5(b), are considerably lower. This is as expected for a ground vehicle.

The yaw angle estimates provided by the filter, Figure 5(c), show greater certainty in GPS measurements by following them more closely than IMU measurements, which are observed to have errors due to the presence of uncertainties. It should be noted that due to the ERKF the yaw estimates are robust to the uncertainties present in the IMU measurements.

5.2.2 Position Estimates

The position estimates are a good fit with GPS readings as shown in Figure 6 as well as in Figure 3, which is a 3 dimensional plot showing the route followed by the cargo transport vehicle. An important observation to be made here is that by implementing an inertial navigation system when GPS measurements are not available, the position estimates are updated at a frequency of 400 Hz, which is greater than the 10 Hz update rate of the GPS. This is depicted in Figure 4.

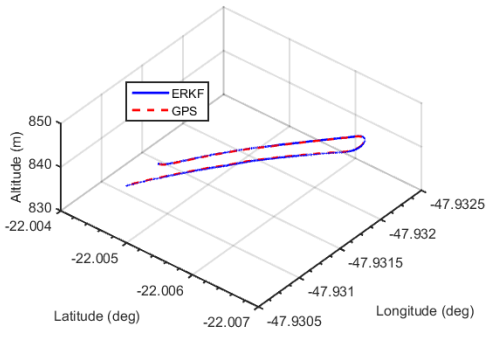

Figure 3: 3D Plot of Vehicle Position.

5.3 Numerical analysis

The maximum and minimum singular values of the state covariance matrix of the attitude sys-

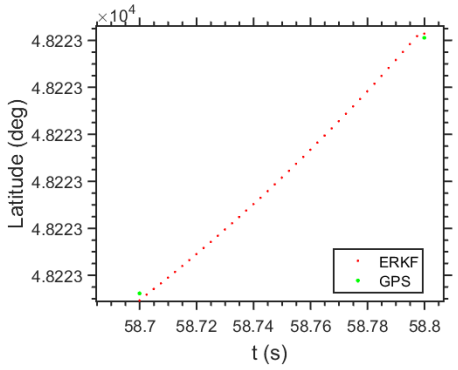

Figure 4: Update Rates of GPS and Estimates.

tem are shown in Figure 7. The ERKF has been implemented through the conventional matrix inversion approach as well as the proposed Givens rotation approach. The corresponding maximum and minimum singular values of the state covariance matrix of the position system are shown in Figure 8.

These figures clearly show that the singular values of the covariance matrices obtained from the standard matrix inversion and Givens rotation implementations are very nearly the same. It was seen that absolute value of the difference between the singular values was smaller than 10^{-13} when 64 bit precision floating point arithmetic was used.

6 Conclusions

In this paper we have presented an attitude and heading reference system, based on IMU and GPS data, using the ERKF. The yaw estimates provided by the ERKF followed the accurate GPS measurements. The position estimates were obtained at a higher update frequency, due to the utilization of an inertial navigation system based on attitude estimates. And the ERKF ensured that the estimates are robust to uncertainties in both system models.

Additionally, we have presented and verified a method for computing the ERKF in real-time based on QR decomposition using Givens rotation. The increased computational efficiency of this method over the conventional matrix inversion approach has been discussed.

Acknowledgement

This work was supported by grants #2014/08432-0, #2014/50851-0 and #2015/18085-8, São Paulo Research Foundation (FAPESP) and by grants #484095/2013-7 and 465755/2014-3, Brazilian National Council for Scientific and Technological Development (CNPq).

References

Anderson, B. D. O. and Moore, J. B. (1979). *Optimal filtering*, Prentice-Hall.

Bijker, J. and Steyn, W. (2008). Kalman filter configurations for a low-cost loosely integrated inertial navigation system on an airship, *Control Engineering Practice* **16**(12): 1509–1518.

Bonato, V., Peron, R., Wolf, D. F., de Holanda, J. A. M., Marques, E. and Cardoso, J. M. P. (2007). An FPGA implementation for a Kalman filter with application to mobile robotics, *International Symposium on Industrial Embedded Systems*, Lisbon, Portugal, pp. 148–155.

Chen, G. and Guo, L. (2005). The FPGA implementation of Kalman filter, *International Conference on Signal Processing, Computational Geometry & Artificial Vision,*, Malta, pp. 61 – 65.

Farrel, J. A. (2008). *Aided navigation GPS with high rate sensors*, The McGRaw-Hill Companies, New York.

Francis, J. G. F. (1962). The QR transformation part 2, *The Computer Journal* **4**(4): 332–345.

Givens, W. (1958). Computation of plane unitary rotations transforming a general matrix to triangular form, *Journal of the Society for Industrial and Applied Mathematics* **6**(1): 26–50.

Golub, G. H. and Loan, C. F. V. (1996). *Matrix Computations*, The Johns Hopkins University Press.

Inoue, R. S., Terra, M. H. and Cerri, J. P. (2016). Extended robust Kalman filter for attitude estimation, *IET Control Theory Applications* **10**(2): 162–172.

Ishihara, J. Y., Terra, M. H. and Cerri, J. P. (2015). Optimal robust filtering for systems subject to uncertainties, *Automatica* **52**(1): 111–117.

Kailath, T., Sayed, A. H. and Hassibi, B. (2000). *Linear Estimation*, Prentice-Hall, New Jersey, USA.

Kalman, R. E. (1960). A new approach to linear filtering and prediction problems, *Transactions of the ASME - Journal of Basic Engineering* **82**(Series D): 35–45.

Kim, P. (2011). *Kalman Filter for Beginners: with MATLAB Examples*, CreateSpace Independent Publishing Platform.

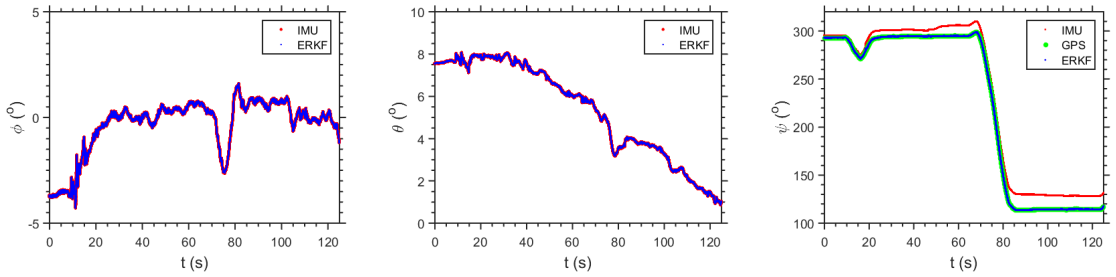

Figure 5: Attitude estimates of the ERKF: roll (ϕ), pitch (θ) and yaw (ψ).

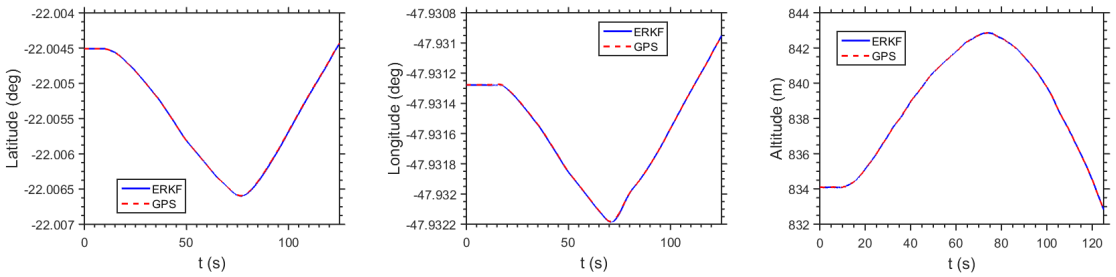

Figure 6: Position estimates of the ERKF: latitude, longitude and altitude.

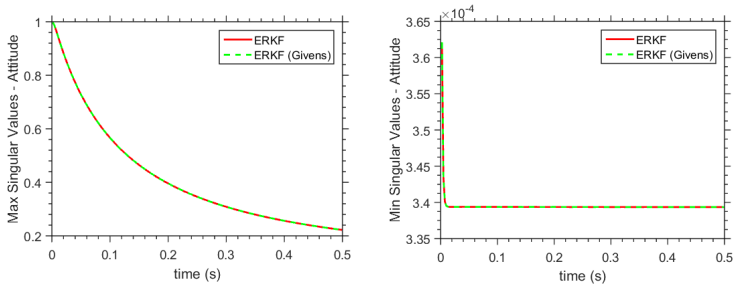

Figure 7: Maximum and minimum singular values of state covariance matrix P of the attitude system model.

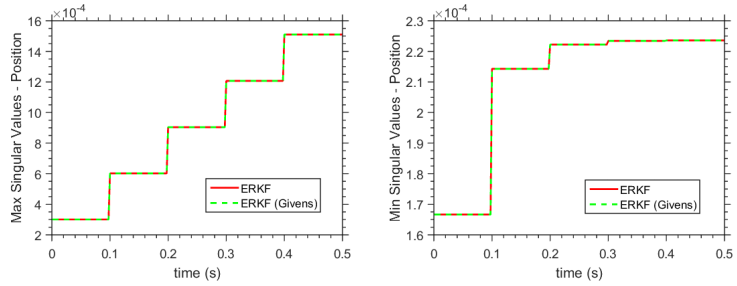

Figure 8: Maximum and minimum singular values of state covariance matrix P of the position system model.

Lee, C. R. and Salcic, Z. (1997). High-performance FPGA-based implementation of Kalman filter, *Microprocessors and Microsystems* **21**(4): 257 – 265.

Sayed, A. H. (2001). A framework for state-space estimation with uncertain models, *IEEE Transactions on Automatic Control* **46**(1): 998–1013.

Terra, M. H., Ishihara, J. Y. and Inoue, R. S. (2011). A new approach to robust linear filtering problems, *IFAC World Congress*, Milano, Italy.

Wang, X. and Leeser, M. (2009). A truly two-dimensional systolic array FPGA implementation of QR decomposition, *ACM Trans. Embed. Comput. Syst.* **9**(1): 3:1–3:17.

Xing, Z. and Gebre-Egziabher, D. (2008). Modeling and bounding low cost inertial sensor errors, *IEEE/ION Position, Location and Navigation Symp.*, Monterey, California, USA.