An online filter study for inertial properties estimation based on low-cost sensors

Bruno Miguel Almeida Tibério
bruno.tiberio@tecnico.ulisboa.pt

Instituto Superior Técnico, Lisboa, Portugal

November 2017

Abstract

There is a clear trend in automotive industry towards autonomous driving. The knowledge of variables such as vehicle positioning, speed and its orientation are vital aspects for the correct trajectory control. Also important is the inertia, especially the moment associated with rotations about the vertical axis in light vehicles. Following this idea, this work, as part of a bigger project consisting on control of an autonomous car, aims at providing estimates for orientation, position, speed and vertical moment of inertia. The orientation will be derived based on the Attitude and Heading Reference System (AHRS) algorithm, known from the literature but with some innovative contributions, using low cost inertial sensors. The positioning and velocity will be estimated using an extended Kalman filter by fusing inertial sensor readings with the output of a couple of Global Navigation Satellite System (GNSS) receivers. The estimative of ratio between vertical moment of inertia and the mass will use the recursive least squares method, taking as information the readings of inertial sensors.

Keywords: GNSS, AHRS, Kalman, Inertia, Localization

1. Introduction

Current trend in autonomous driving for civilian use, while promising a significant reduction in accidents and traffic flow, also opens new fields about smart cities, software reliability, communications security between vehicles and/or infrastructures that need to be studied, researched and tested.

With that in mind, this work is inserted in a major project consisting in the conversion of an old electric car (Fiat Elettra) property of Instituto Superior Técnico (IST) as an autonomous vehicle to serve as a framework for the academic community and research in the field. The objective is to estimate of vehicle’s attitude, location, velocities and moment of inertia associated with car’s vertical axis since they have a key role in trajectory control. Also there is an effort to ensure that solutions are possible to be used in online mode.

2. Background

In this section is explained several notations and definitions used along the article important to help the reader to walk through the work. The notation system of leading superscripts and subscripts is used to denote relative frames orientation or general physical quantities (vectors, points). For frames orientations, leading subscript refers to the frame being represented with respect to the frame in leading superscript. For example let R be a rotation matrix. Using the notations stated before, \( a^b R \) describes orientation of frame b with respect to frame a. For physical quantities, a vector is represented in the frame defined by is leading superscript, \( a^v \) and in similar way points follow the same rule, \( a^P = [^a x, ^a y, ^a z] \).

2.1. ENU Frame, \( w \)

In order to have a clear notion of frames used along the article, it will be necessary to define it. The frames cited are Earth Centered, Earth Fixed (ECEF), World Geodetic System (WGS84), East North Up (ENU) and body frame. The ECEF, WGS84 are just auxiliary frames used to define the ENU frame which will be considered the world frame.

ENU coordinate system is a local coordinate system where the origin is located at a user defined point in ECEF coordinate system, with Y axis
pointing towards North Pole and X axis pointing towards East. The plan defined by X and Y axis is tangent to the WGS84 frame on the origin of ENU, Z axis express the altitude from defined local plane (see figure 1). The ENU frame is considered in this work as the reference frame and will be denoted with superscript or subscript w. in (1) is expressed the transformation used for converting points in ECEF frame to local ENU frame given the corresponding latitude($\varphi_r$) and longitude($\lambda_r$) of reference point ($X_r,Y_r,Z_r$) where the rotation matrix follows the implementation in [18].

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix}_{ENU} = ECEF_{ENU} R(\varphi_r, \lambda_r) \begin{bmatrix} X - X_r \\ Y - Y_r \\ Z - Z_r \end{bmatrix}_{ECEF}$$

Local ENU frame can be approximated to an inertial frame. Although this is not true, rotation of earth is about an order of 10 smaller then the minimum sensed value given by gyroscope according to [8]. Also, the expected root mean square (rms) noise in gyroscope is almost 90 times greater than earth rotation rate ($\approx 15^\circ$ per hour).

2.2. Body frame, $b$

Inertial sensors are placed as possible near the middle point of the bisector segment between the two wheel axes in such a away that YY axis is pointing towards the front of vehicle, XX axis is pointing to the right side of the car and ZZ axis is pointing to the top. This way, if the Euler angles describing the orientation of body frame related to world frame are all equal to zero, it means the axes in each frame are coincident apart from an offset in origin. Rotation angles are considered positive following the right hand rule in each axis. The origin of body frame is equal to intersection of rear wheel axis with the bisector defined above as shown in figure 2.

![Figure 2: Car scheme and body frame](image)

2.3. Euler angles and quaternions

Euler angles are a form to describe orientation of one frame relative to another by defining three angles. The convention adopted in this work is the Tait–Bryan intrinsic rotation sequence Z-Y-X, meaning referential $a$ axes, represented in figure 3, can be mapped into referential $b$ by performing sequential rotations, first along ZZ axis by an angle $\psi$, second along the resulting YY axis by an angle $\theta$ and final rotation along the resulting XX axis by an angle $\phi$.

During the work an AHRS algorithm based on [12] is implemented and it uses quaternions. That away the important definitions for understanding the filter derivations are explained following the same notation as present by [11]. Quaternion is a complex number in four dimension that can be used, similar Euler angles, to represent orientation of frames with respect to others. A quaternion can be represented by (2) where $q_1$ is the norm of it, $q_2,q_3$ and $q_4$ are complex coordinates with $i,j,k$ being the axis versors. If the quaternion is normalized, it is denoted with a circumflex accent as $\hat{q}$

$$q = q_1 + q_2 i + q_3 j + q_4 k$$

Using the same notation as stated before, $\mathbf{w}_b q$ represent the orientation of body frame with respect to world frame. The quaternion conjugate describes the inverse rotation and is defined as equation 3.

$$\mathbf{w}_b q^* = \mathbf{w}_b q = [q_1 - q_2 - q_3 - q_4]$$

Let’s define the following quaternion representation of the same vector but in each referential by using is pure quaternion as $\mathbf{w}_b v = [0 \ w_x \ w_y \ w_z]$ and $\mathbf{w}_a v = [0 \ a_x \ a_y \ a_z]$. The rotation of vector $v$ from one frame to other, using a quaternion, is performed by (4) where $\otimes$ denotes the quaternion product.

<table>
<thead>
<tr>
<th>Object (point)</th>
<th>$bX$[m]</th>
<th>$bY$[m]</th>
</tr>
</thead>
<tbody>
<tr>
<td>GPS1 antenna (GPS1)</td>
<td>0.71</td>
<td>0.4</td>
</tr>
<tr>
<td>GPS2 antenna (GPS2)</td>
<td>-0.71</td>
<td>0.4</td>
</tr>
<tr>
<td>Razor 1 (CM)</td>
<td>0</td>
<td>1.35</td>
</tr>
<tr>
<td>Origin (O)</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

Table 1: Coordinates of principal components in body frame
Using either Euler angles or quaternions, it is possible to describe the associated rotation matrix [17].

3. Sensors

In this section is briefly described the inertial sensors (Razor 9DOF IMU) used and their calibration. Additionally it was used two industrial grade GNSS receivers from Novatel, where no particular effort was made and it was assumed they produce the best reports of position and velocity they can. Location of all sensors are shown in the 2D projection of the car in figure 2. The Inertial Measurement Unity (IMU) board is composed by an accelerometer (ADXL345), gyroscope (ITG3200) and magnetometer (HMC5843) all with 3-axis and digital, companioned with a Microcontroller Unit (MCU), Atmega328p.

3.1. Accelerometer sensor model and calibration

Equation (5) describes the general output of accelerometer [15] where, \( bA_{ext} \) is the sum of real external acceleration due to linear or rotational dynamics in the body frame, \( bR \) is the rotation matrix mapping quantities in world frame into sensor frame, \( wg \) represents the fictitious acceleration due to gravity in the world frame, \( bA_0 \) is an offset and \( \delta A \) describes additive gaussian noise. Ideally, \( G \) should be equal to identity matrix but in fact it represents the product of two matrices, one describing the cross-axis influence and the other a scale factor for each axis [15].

\[
_bA_s = G[ bA_{ext} + bR wg ] + bA_0 + \delta A
\]  
(5)

For this application it will be used the suggested calibration by manufacturer [14] resulting in exposing the sensor to six position combination while resting. This means the \( bA_{ext} \) will be zero and accelerometer will be only influenced by the gravity force, 1g. This assumes that cross-axis influence is small enough and can be neglected, this way, \( G \) is a diagonal matrix with the scale factors for each axis.

3.2. Gyroscope sensor model

Gyroscope output is proportional the angular velocity sensed of that axis and is given by equation (6) where \( b\omega_s \) represents the vector output of the sensor in body frame at instant \( t \), \( b\omega_{real} \) is the real angular velocity applied to sensor in body frame, \( b\omega_0 \) is the bias term and \( \delta\omega \) additive gaussian noise. Since the sensor is described in [8] as factory calibrated, \( G \) which represent a scaled factor correction, is expected to be equal to identity matrix assuming cross-axis misalignment and linear acceleration sensitivity are small for the application purpose so they are considered irrelevant. In general, the offset value depends on the temperature value inside chip. Once internal stability is reached, offset values tends to be constant [21].

\[
b\omega_s = G b\omega_{real} + b\omega_0 + \delta\omega
\]  
(6)

Scale factor referred in [8], is equal to 14.375LSB/°.s\(^{-1}\) and factory calibrated. At power on a series of samples are collected to determine the mean bias value. For correct performance, sensor should be at rest.

3.2.1 Magnetometer model and calibration

The magnetometer sensor model follows a similar structure as two previous sensors and is given by equation (7). The \( bH_s \) is the value sensed by the sensor in body frame. \( C \) is a matrix resulting from the multiplication of matrices containing influence of cross-axis, gain scale factor and soft-iron interference [13]. \( wH_e \) represent the geomagnetic vector of Earth, \( bH_{offset} \) is an offset vector which describes the influence of zero-field offset and the ferromagnetic masses fixed relative to body frame usually denoted as hard-iron. \( \delta h \) represents gaussian additive noise.

\[
bH_s = C bR wH_e + bH_{offset} + \delta h
\]  
(7)

\( wH_e \) represent the geomagnetic vector of Earth. Let us define now, \( wH_0 \) as the horizontal projection of \( wH_e \). The angle between \( wH_0 \) and \( wH_e \) is denominated as inclination, \( I \). The horizontal projection of the geomagnetic vector points to the magnetic

<table>
<thead>
<tr>
<th>Gains and offset [LSB]</th>
<th>Razor 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Z gain</td>
<td>253.66</td>
</tr>
<tr>
<td>Z offset</td>
<td>-3.47</td>
</tr>
<tr>
<td>Y gain</td>
<td>266.27</td>
</tr>
<tr>
<td>Y offset</td>
<td>-3.18</td>
</tr>
<tr>
<td>X gain</td>
<td>266.04</td>
</tr>
<tr>
<td>X offset</td>
<td>13.83</td>
</tr>
</tbody>
</table>

Table 2: Summary of values extracted and calculated for each axis of accelerometer.
north which may not be aligned with the north axis of reference frame, \( ^w Y \). This defines the magnetic declination angle, \( D \), as the angle between horizontal projection and the \( ^w Y \) axis.

The compensation for hard and soft-iron effects can be done using an ellipsoid geometric approach as described by [13]. However since the body frame is a vehicle, it is not physically possible to collect points based in rotations of \( YY \) and \( XX \) axis. Only \( ZZ \) axis rotations are available and the data instead of belonging to an ellipsoid surfaces, will belong to an ellipse in \( Z \)-plane as a result from that plan cutting the ellipsoid in some undetermined \( z \) coordinate. Because of this restriction it will be considered that the influence in \( ZZ \) axis is zero and the scale factor is equal to one. Offset in \( ZZ \) axis will also be considered zero.

Without loss of generality, \( C^{-1}\delta h_c \) will result also in a gaussian vector so it will be rewritten as \( \delta h_c \). (7) can be arranged into (8) where \( ^b H_c \) is the corrected value after calibration process. The matrix \( C^{-1} \) and \( ^b h_{offset} \) is obtained while collecting within \( 360^\circ \) orientation as described by [4]

\[
\begin{align*}
^b R(t)^w h_c &= C^{-1}[^b H_c - ^b h_{offset} - \delta h_c] \\
^b H_c &= C^{-1}[^b H_c - ^b h_{offset}] + \delta h_c
\end{align*}
\]

5. Attitude Filter

In this section is derived the filter to be deployed on Razor MCU. All treatment is applied assuming the sensors were previously calibrated as described in section 3 and preprocessed. Derivation is based on the Sebastian Madgwick filter implementation and basis will be summarized here to understand the process and not full extension steps to arrive the final form of it, since later it will be performed changes to fit the purpose of this work. Users should look into [11] or [12] for full details of the in between steps.

5.1. Sensor orientation given angular velocities

Let us denote the pure quaternion \( ^b w = [0^b w_z^b w_y^b w_x] \) as the body frame angular velocities vector represented in the quaternion space. Continuous rate of change of reference frame, relative to body frame, is described by (10).

\[
^b w q = \frac{1}{2} \frac{\dot{\omega}}{\|\dot{\omega}\|} \otimes ^b w
\]

Let \( ^w q_{\omega,t} \) be the quaternion rate at instant \( t \) given current angular speed. Similar, let \( ^w q_{ext,t} \) be the quaternion estimate at instant \( t \). The approximated \( ^w q_{ext,t} \) can be calculate using numerical integration as described by (11) and (12), provided that previous state is known and being \( \Delta t \) the sampling period.

\[
^w q_{\omega,t} = \frac{1}{2} ^w q_{ext,t-1} \otimes ^b w_t \quad \text{(11)}
\]

\[
^w q_{ext,t} = ^w q_{ext,t-1} + ^w q_{\omega,t} \Delta t \quad \text{(12)}
\]

5.2. Sensor orientation given world observable vector

As seen in section 3 the accelerometer and magnetometer are capable of sensing world observable vectors, in this case, the gravity force and the local geomagnetic vector respectively.

Let \( ^w d \) be a generic observable vector defined as pure quaternion representation of it, in the world frame, as in (13). Now, let \( ^b s \) be the estimate of same but in body frame as defined in (14).

\[
y(n) = (1 - \alpha)y(n - 1) + \alpha x(n)
\]

\[
\text{with } \alpha = 1 - e^{-F_{cut} / F_s} \approx F_{cut} / F_s + F_{cut}
\]

\[
y(n) = (1 - \alpha)y(n - 1) + \alpha x(n)
\]
\[ w \dot{d} = [0 \ w_d \ d_\theta \ w_d] \quad (13) \]
\[ b \dot{s} = [b_x \ b_y \ b_z] \quad (14) \]

Using equation (4) and the current estimate of orientation quaternion, it is possible to define a cost function as in (15) that represents the difference between current estimate of observable vector and its measurement both represented in body frame.

\[ f(w_{\hat{q}_{est}}, w \dot{d}, b \dot{s}) = \frac{\nabla f(w_{\hat{q}_{est}}, w \dot{d}, b \dot{s})}{\nabla f(w_{\hat{q}_{est}}, w \dot{d}, b \dot{s})} \quad (15) \]

If the quaternion orientation estimate is correct, the cost function result should tend to a minimum. So, the main objective is to minimize the cost function by improving the estimate of quaternion using the gradient method as shown in equations (16) and (17). The gradient of cost function can be calculated by its Jacobian \( J \) as expressed in (18).

\[ \min \left( f(w_{\hat{q}_{est}}, w \dot{d}, b \dot{s}) \right) \quad (16) \]
\[ b_{w \dot{q}_{k+1}} = b_{\hat{q}_{k}} - \mu \nabla f(w_{\hat{q}_{k}}, w \dot{d}, b \dot{s}) \quad (17) \]
\[ \nabla f(w_{\hat{q}_{k}}, w \dot{d}, b \dot{s}) = \nabla f(w_{\hat{q}_{k}}, w \dot{d}, b \dot{s}) \quad (18) \]

Specifying for gravity vector results in set of equations (19) where \( b_g \) denotes the normalized accelerometer reading in its pure quaternion and \( w \dot{g} \) is gravity vector. The resulting matrices \( f \) and \( J \) for gravity, are denoted from now on as \( f_g \) and \( J_g \).

\[ w \dot{d} = w \dot{g} = [0 \ 0 \ 0 \ 1] \quad (19a) \]
\[ b \dot{s} = b \dot{a} = [0 \ a_x \ a_y \ a_z] \quad (19b) \]
\[ f(w_{\hat{q}_{est}}, k, w \dot{d}, b \dot{s}) = f_g(w_{\hat{q}_{est}}, k, b \dot{a}) = 2(q_2q_1 + q_3q_t) - a_x \quad (19c) \]
\[ J_g(w_{\hat{q}_{est}}, k, w \dot{d}, b \dot{s}) = \begin{bmatrix} -2q_3 + 2q_4 - 2q_2 + 2q_3 \\ 2q_2 + 2q_4 + 2q_4 + 2q_1 \\ 0 \ -4q_2 - 4q_4 \ 0 \end{bmatrix} \quad (19d) \]

As for geomagnetic vector result in equations (20) where \( b \dot{m} \) denotes the normalized magnetometer reading in its pure quaternion and similar \( w \dot{h} \) is geomagnetic field vector. The resulting matrices \( f \) and \( J \) for the geomagnetic field, are denoted from now on as \( f_h \) and \( J_h \).

\[ w \dot{d} = w \dot{h} = [0 \ 0 \ \cos(I) \ \sin(I)] \quad (20a) \]
\[ b \dot{s} = b \dot{m} = [0 \ m_x \ m_y \ m_z] \quad (20b) \]

\[ f_h(w_{\hat{q}_{est}}, k, b \dot{m}) = \begin{bmatrix} 2h_y(q_2q_4 + q_2q_3) + 2h_z(q_2q_4 - q_1q_3) - m_x \\ 2h_y(q_3q_4 - q_1q_3) + 2h_z(q_2q_4 - q_1q_3) - m_y \\ 2h_y(q_2q_4 + q_1q_3) + 2h_z(q_2q_4 - q_1q_3) - m_z \end{bmatrix} \quad (20c) \]
\[ J_h(w_{\hat{q}_{est}}, k, b \dot{m}) = \begin{bmatrix} 2h_yq_2 - 2h_zq_3 & 2h_yq_3 + 2h_zq_4 & 2h_yq_4 + 2h_zq_2 \\ 2h_zq_3 - 4h_yq_4 + 2h_zq_2 \ & -2h_yq_4 - 2h_zq_1 \ & -2h_yq_3 - 2h_zq_1 \ & 2h_yq_4 - 4h_zq_3 \\ & & 2h_yq_3 \end{bmatrix} \quad (20d) \]

\[ w \dot{h} = [0 \ w_h \ w_h \ w_h] \quad (21) \]
\[ w \dot{h} = [0 \ 0 \ \sqrt{w_{h_x}^2 + w_{h_y}^2} \ w_{h_z}] \quad (22) \]

Following, results in the final form of quaternion estimate (26), given the gradient in (25) resulting from combination of both matrices associated at each observable vector, (23) and (24). \( b_{\dot{w} \hat{q}_{k+1}} \) denotes that estimate comes from the gradient method.

\[ f_g,h(w_{\hat{q}_{est}}, k, w \dot{h}, b \dot{a}, b \dot{m}) = \begin{bmatrix} f_g(w_{\hat{q}_{est}}, k, b \dot{a}) \\ f_h(w_{\hat{q}_{est}}, k, b \dot{m}) \end{bmatrix} \quad (23) \]
\[ J_{g,h}(w_{\hat{q}_{est}}, k, w \dot{h}) = \begin{bmatrix} J_g(w_{\hat{q}_{est}}, k) \\ J_h(w_{\hat{q}_{est}}, k) \end{bmatrix} \quad (24) \]
\[ \nabla f = J_{g,h}(w_{\hat{q}_{est}}, k, w \dot{h})f_g,h(w_{\hat{q}_{est}}, k, w \dot{h}, b \dot{a}, b \dot{m}) \quad (25) \]
\[ b_{\dot{w} \hat{q}_{k+1}} = b_{\dot{w} \hat{q}_{est}, k} - \mu \frac{\nabla f}{\|\nabla f\|} \quad (26) \]

Given that we can estimate the orientation of the quaternion by using the gyroscopes (12) and using the observable vectors (26) a filter fusion algorithm is used by to handle the long term accumulation errors due to gyroscopes and rapid change response of accelerometer and magnetometer as expressed in (27).
\[ b_{\varphi_{ext,k}} = \gamma b_{\varphi_{\nabla,k}} + (1 - \gamma) b_{\varphi_{w,k}} \]  \hspace{1cm} (27)

In [11] is performed a set of simplifications of equation (27) using the starting observation that the optimal weight, \( \gamma \), should be such that the rate of convergence of \( b_{\varphi_{\nabla,k}} \) should be equal to the rate of divergence of \( b_{\varphi_{w,k}} \) caused by the magnitude of quaternion derivative due to gyroscope measurements errors. The final form is present in the set of equations (28) where \( \beta \) is the divergence rate of \( b_{\varphi_{w,k}} \). The simplifications performed in between are detailed in [11].

\[ b_{\varphi_{k}} = \frac{\nabla f}{\|\nabla f\|} \]  \hspace{1cm} (28a)

\[ b_{\varphi_{ext,k}} = b_{\varphi_{w,k}} - \beta b_{\varphi_{k}} \]  \hspace{1cm} (28b)

\[ b_{\varphi_{ext,k}} = b_{\varphi_{ext,k-1}} + b_{\varphi_{ext,k}} \Delta t \]  \hspace{1cm} (28c)

A reference value for \( \beta \) is such that the optimal step size should be equal to the rate of divergence of \( b_{\varphi_{w,k}} \) caused by zero mean error sources in gyroscope and assuming all axis have equal error, \( \ddot{\omega}_g \), it would be given by (29). \( \beta \) is user adjustable as compromise between the convergence rate and over-shooting of gradient method.

\[ \beta = \left\| \frac{1}{2} \tilde{q} \otimes \begin{bmatrix} 0 & -\ddot{\omega}_g & \ddot{\omega}_g & \ddot{\omega}_g \end{bmatrix} \right\| = \frac{\sqrt{3}}{2} \ddot{\omega}_g \]  \hspace{1cm} (29)

5.3. External accelerations and magnetic field distortions

The original filter implementation do not account for external acceleration or unexpected large change in magnetic field which can be considered normal based on the original application it was developed for, where movements can be considered smooth enough and the location confined. However, in this case, the objective is to use the sensors within a moving car where external acceleration are expected and other cars, for example, may induce a great distortion in magnetic field. This would result in error of orientation estimative if no action is taken.

Let \( extAcc \) be a binary flag to denote the presence of external accelerations. This value can be defined as an adjustable threshold as shown in (30).

\[ extAcc = \begin{cases} 1 & , \|b_a\| = 1 \pm K \sigma_{max} \\ 0 & , \text{otherwise} \end{cases} \]  \hspace{1cm} (30)

In the cases where \( extAcc \) is true, the step correction is not applied or is applied only based on magnetometer. Similar, unexpected temporary distortions caused in the magnetic field can be detected and removed using the same principle described for the external accelerations as shown in (31). Figure 6 show the simulation result for normal AHRS code and new improved algorithm when injecting external acceleration or magnetic field.

\[ extMag = \begin{cases} 1 & , \|b_m\| = \|\tilde{h}_b\| \pm K \sigma_{max} \\ 0 & , \text{otherwise} \end{cases} \]  \hspace{1cm} (31)

Using both improvements, the resulting pseudo code as shown in algorithm 1

\textbf{Algorithm 1 Correct for external influences}

1: \textit{flag} ← \( 1 \times extMag + 2 \times extAcc \)
2: \textbf{switch} \textit{flag} \textbf{do}
3: \textbf{case} 0: perform gradient step correction as usual
4: \hspace{1cm} \( b_{\varphi_{ext,k}} \leftarrow \frac{1}{2} b_{\varphi_{ext,k-1}} \otimes b_{\Delta t} - \beta \nabla f_b \) \hspace{1cm} (31a)
5: \hspace{1cm} \( b_{\varphi_{ext,k}} \leftarrow b_{\varphi_{ext,k-1}} + b_{\varphi_{ext,k}} \Delta t \) \hspace{1cm} (31b)
6: \textbf{case} 1 \hspace{1cm} \( \text{perform gradient step correction based only on accelerometer} \)
7: \hspace{1cm} \( b_{\varphi_{ext,k}} \leftarrow \frac{1}{2} b_{\varphi_{ext,k-1}} \otimes b_{\Delta t} - \beta \nabla f_b \) \hspace{1cm} (31a)
8: \hspace{1cm} \( b_{\varphi_{ext,k}} \leftarrow b_{\varphi_{ext,k-1}} + b_{\varphi_{ext,k}} \Delta t \) \hspace{1cm} (31b)
9: \textbf{case} 2 \hspace{1cm} \( \text{perform gradient step correction based only on magnetometer} \)
10: \hspace{1cm} \( b_{\varphi_{ext,k}} \leftarrow \frac{1}{2} b_{\varphi_{ext,k-1}} \otimes b_{\Delta t} - \beta \nabla f_b \) \hspace{1cm} (31a)
11: \hspace{1cm} \( b_{\varphi_{ext,k}} \leftarrow b_{\varphi_{ext,k-1}} + b_{\varphi_{ext,k}} \Delta t \) \hspace{1cm} (31b)
12: \textbf{case} 3 \hspace{1cm} \( \text{do not apply gradient correction step, use gyroscope only} \)
13: \hspace{1cm} \( b_{\varphi_{ext,k}} \leftarrow \frac{1}{2} b_{\varphi_{ext,k-1}} \otimes b_{\Delta t} \) \hspace{1cm} (31a)
14: \hspace{1cm} \( b_{\varphi_{ext,k}} \leftarrow b_{\varphi_{ext,k-1}} + b_{\varphi_{ext,k}} \Delta t \) \hspace{1cm} (31b)

Figure 6: Simulation of external accelerations injection (left) and external magnetic field injection (right)
Table 3: Advised max inclination profile for national roads in Portugal and respective angle.

<table>
<thead>
<tr>
<th>Vel. [Km/h]</th>
<th>Max Inc.[%]</th>
<th>Angle [°]</th>
</tr>
</thead>
<tbody>
<tr>
<td>40</td>
<td>8</td>
<td>4.57</td>
</tr>
<tr>
<td>60</td>
<td>7</td>
<td>4.00</td>
</tr>
<tr>
<td>80</td>
<td>6</td>
<td>3.43</td>
</tr>
<tr>
<td>100</td>
<td>5</td>
<td>2.86</td>
</tr>
<tr>
<td>120</td>
<td>4</td>
<td>2.29</td>
</tr>
</tbody>
</table>

5.4. XY improvement version

When applying the improved version described in subsection 5.3 to data collect in the field, it was clear that majority of time, extMag was flagged and subsequently magnetometer correction was almost never used. Figure 7 exposes that fact for trajectory test 1, the same used for calibration of magnetometer. In fact, comparing figure 7a with figure 5b, where only the X and Y axis readings from magnetometer were used to estimate the $\psi$ orientation, it is clear that it tracks better the assumed ground truth provided by GNSS.

![Figure 7: Magnetometer norm and yaw angle estimation for trajectory 1 using improved version](image)

The comparison lead to believe that fluctuation in ZZ axis of body frame due to impossibility of physical calibration of magnetometer in this axis, is the main cause for signaling the extMag flag. However, since in normal conditions, cars are majority of time in quasi-horizontal or small angles variation. Table 3 show the advised max inclination and respective angle with horizontal for roads in Portugal[7]. As so, a new improvement version of algorithm is derived to track only the horizontal component of geomagnetic field instead its full vector.

Recalling (20) it will now take the form of (32).

\[
\begin{align}
\dot{\mathbf{w}d} &= \mathbf{w}h = \begin{bmatrix} 0 & 0 & 1 & 0 \end{bmatrix} \\
\ddot{\mathbf{b}s} &= \dot{\mathbf{b}}\mathbf{m} = \begin{bmatrix} 0 & m_x & m_y & 0 \end{bmatrix} 
\end{align}
\]

6. Sensor fusion INS/GNSS

The Inertial Navigation System (INS) algorithm takes readings from the IMU as input to estimate the position, velocity of vehicle and $\psi$ orientation angle recurring to kinematics equations. The attitude filter developed in section 5, even in the presence of small bias errors, has a bounded output because only gyroscope readings are integrated while accelerometer and magnetometer provides correction to drift. In INS algorithm, gyroscope is integrated once and accelerometer is twice, meaning small error will be propagated to output growing with the time and time squared respectively. It is important to re-estimate the bias because initial conditions during each of the runs are not granted to be identical and those values will be also used in future. A loosely coupled extended Kalman filter, in direct form, is derived to fuse information from Razor unit and GNSS receivers. Vehicle dimensions used to collect data are presented in figure 2 and sensor locations too. Total filter fusion is composed by two distinct parts. During a limited time (initialization), a Zero Velocity Update (ZVU) filter is used to estimate the bias of sensors. When the car is static, inputs are zero which allow the

\[
\begin{align}
f_h(b_q, b_{ext,k}, \dot{b_m}) &= \begin{bmatrix} 2(q_1q_4 + q_2q_3) - m_x \\ 2(1 - q_2^2 - q_1^2) - m_y \\ 2(q_3q_1 - q_2q_2) \end{bmatrix} \\
J_h(b_q, b_{ext,k}) &= \begin{bmatrix} 2q_4 & 2q_3 & 2q_2 & 2q_1 \\ 0 & -4q_2 & 0 & -4q_1 \\ -2q_2 & -2q_1 & 2q_4 & 2q_3 \end{bmatrix}
\end{align}
\]
errors in IMU to be estimated. The bias will be modeled as a random walk process driven by white noise [10]. It is assumed they are a slow changing rate variables and during the tests time they can be approximated as a constant. After the initialization, estimated values are passed to second part of the filter and subtracted from the readings of IMU sensors. IMU readings are also compensated with the gravity given estimated orientation provided by the AHRS (see figure 9). In the kalman filters, the nomenclature followed is the same as present in [16].

6.1. Vehicle kinematics

For the needs of this application a simple 2D body frame dynamic model will be used with an Ackermann steering. A few assumptions are made: (i) The vehicle is only capable of translating along the body YY axis. (ii) The vehicle can cause rotations only around the body ZZ axis. Other rotation are consequence of road grade and suspension movement. (iii) The vehicle Euler angle $\theta$ (rotation along YY body axis) at wheel axis level is close to zero and will be considered zero. Assuming the road is close to horizontal level, majority of $\theta$ is due to suspension movement. (iv) There is no slip. Tires are capable of sustain forces generated during dynamics.

Following this assumptions, the 2D representation of the car with estimated location of the sensors in body frame is presented in figure 2. Using the center of rear axis as origin of the body frame. That said, the coordinates of important points in the 2D plane is present in table 1. The general kinematic equations that govern the 2D model in horizontal plan are defined in (33) using the assumptions listed previously and simplified bicycle model. There is no access to the angle of wheels. In this case the rate of change of Euler angle $\psi$ must be estimated from the propagation of attitude due to angular speed.

\[
\begin{bmatrix}
  wX_0 \\
  wY_0 \\
  \psi \\
  wV_y
\end{bmatrix} =
\begin{bmatrix}
  -wV_y \sin(\psi) \cos(\phi) \\
  wV_y \cos(\psi) \cos(\phi) \\
  \cos(\phi) \omega_z \\
  bA_y
\end{bmatrix}
\]

(33)

6.2. ZVU filter

During the ZVU filter time, the input of system is known and all equal to zero because the vehicle is stopped. State of filter ois defined by, origin of body frame in world frame $wO$, Euler angle $\psi$, the bias estimate of accelerometer in body frame axes $X$ and $Y$, $bA_0$ and bias estimate of gyroscope axis $Z$ in body frame, $b\omega_0$. As shown in (34). The differential equations that govern the system state are represented in (35).

\[
x = \begin{bmatrix} wX_o & wY_o & \psi & bA_x & bA_y & b\omega_0 \end{bmatrix}^T
\]

(34)

\[
x_k = A x_{k-1} + B u_{k-1} = I x_{k-1}
\]

(35)

\[
P_k = AP_{k-1}A^T + Q = P_{k-1} + \Delta T I [0, 0, 0, \sigma_{RW, Acc}^2, \sigma_{RW, Acc}^2, \sigma_{RW, gyro}^2]^T
\]

(36)

To calculate associated values for random walk that compose the process noise matrix $Q$, is used the formulation given in [22] for the theoretical limit using specification values of [8] and the configured bandwidth (10Hz for gyroscope and 50Hz for accelerometer) used in the sensors resulting in (37), assuming it is equal for all axes.

\[
\sigma_{RW, Acc} = \sigma_{RW, gyro} = \sqrt{\frac{BW \frac{\pi}{2}}{2}} \approx 0.0452 \text{ m/s}^2
\]

(37)

The measurement phase of the ZVU filter use positions reported by each GNSS receiver, corrected for the lever arm, and readings provided by the AHRS for Euler angle IMU readings. (38) represent the reading provided by GNSS receiver with respect to origin point where after, only the X and Y components are extracted. Similar, the AHRS and IMU measurements are presented in (39), where, again, only the X and Y components are extracted from accelerometer. For simplicity, from now on, quantities related to IMU will also be denoted as AHRS.

Figure 9: Fusion filter diagram
as they all come from the same physical hardware. The observation matrix, \( H \), for each sensor is defined in (40) and the respective measurement covariance matrix, \( R \), in (41). The total \( H \) and \( R \) matrix will be composed by the small matrices from each sensor as shown in (42);

\[
w_{GPS_{1,2}} - b^T R b_{GPS_{1,2}} = w_{P_s} + N(0, \sigma^2_{f_{1,2}}) \tag{38}
\]

\[
\psi_{AHRS} = \psi + N(0, \sigma^2_{\psi_{AHRS}}) \tag{39}
\]

\[
\omega_z = \omega_z + N(0, \sigma^2_{\omega_{GPS}})
\]

\[
H_{GPS_1} = H_{GPS_2} = [I_{2x2} 0_{2x4}] \tag{40}
\]

\[
R_{GPS_1} = diag \left( \sigma^2_{\phi_{GPS_{1,p}}} \sigma^2_{\phi_{GPS_{2,p}}} \right) \tag{41}
\]

\[
R_{AHRS} = diag \left( \sigma^2_{\phi_{AHRS}} \sigma^2_{\omega_{GPS_{1}}} \sigma^2_{\omega_{GPS_{2}}} \right) \tag{42}
\]

\[
R = \begin{bmatrix}
R_{GPS_1} & 0_{2x2} & 0_{2x4} \\
0_{2x2} & R_{GPS_2} & 0_{2x4} \\
0_{2x2} & 0_{2x2} & R_{AHRS}
\end{bmatrix}
\]

6.3 Move filter

After the predefined time used for the ZVU filter, accelerometers and gyroscope readings will be corrected. The newly estimated \( b A_y \) and \( \omega_z \) will be the inputs for Move filter where kinematics is governed by (33). Since they are non-linear, the Extended version of Kalman filter must be used. The first order discrete approximation of (33) is given by matrix of functions \( f \) defined in (44) where the state of move filter is (43).

\[
x = \begin{bmatrix}
w X \omega Y \psi \end{bmatrix}^T \tag{43}
\]

\[
x_k = f(x_{k-1}, u_{k-1}, \delta_{k-1}) = \begin{bmatrix}
w X_{k-1} - b V_{k-1} \sin(\psi_{k-1}) \cos(\phi) \Delta T + \delta_y \\
w Y_{k-1} + b V_{k-1} \cos(\psi_{k-1}) \cos(\phi) \Delta T + \delta_x \\
\psi_{k-1} + \cos(\phi) \omega_z \Delta T + \cos(\phi) \Delta T \delta_y \\
V_{k-1} + b A_y \Delta T + \delta_x
\end{bmatrix} \tag{44}
\]

During prediction phase, \( A \) will be the derivative of \( f \) with respect to state and \( W \) is with respect to process noises as denoted in (45)

\[
A = \frac{\partial f(x_{k-1}, u_{k-1}, 0)}{\partial x_i}
\]

\[
W = \frac{\partial f(x_{k-1}, u_{k-1}, \delta)}{\partial \delta_i} \tag{45}
\]

Regarding matrix \( Q \), for the case of position is reasonable to assume that uncertain value of acceleration cause by unpredictable sources or even road might cause unexpected displacements. Velocity should also be affected by this uncertain acceleration. Orientation, although it was assumed the tires should do not slip, in reality this can happen even during small times caused by unexpected jumps or dirt in road and keep in mind that drift still occurs and its variance grows with time. Following the recommendation given in [10] for non dirt roads, a rough value for maximum acceleration is 0.1g. Also as explained there, it is reasonable to assume that accelerations occurring in body X axis is sustainable small than in forward direction. As for orientation, the author suggests a maximum of 15°/s of angular velocity uncertainty. The resulting process noise following the suggestions is shown in (46).

\[
A_1 = \begin{bmatrix} 0 & 0 & 0 & 0 & 0.1 \end{bmatrix}^T \\
A_2 = \begin{bmatrix} 0 & 0 & 0 & 0 & 0.1 \end{bmatrix}^T \\
Q = \begin{bmatrix} 0.1 \sigma_{\omega_{max}}^2 & 0 & 0 & 0 & 0.1 \sigma_{\omega_{max}}^2 \end{bmatrix}^T \tag{46}
\]

In measurement phase for GNSS, additionally to (38) will be used also (47) and the (39) will be reduced only to \( \psi \) measurement resulting in following set of equations for each sensor.

\[
w_{GPS_{1,2}} = \Omega_x \begin{bmatrix} w X \omega Y \psi \end{bmatrix} + w Y + N(0, \sigma^2_{\psi_{1,2}}) \tag{47}
\]

\[
h_{GPS_1} = \begin{bmatrix}
w X + X_1 \cos(\psi) - Y_1 \sin(\psi) \cos(\phi) \\
w Y + X_1 \sin(\psi) + Y_1 \cos(\psi) \cos(\phi) \\
-\omega_x(X_1 \cos(\psi) + Y_1 \sin(\psi) \cos(\phi)) \\
w_z(X_1 \cos(\psi) - Y_1 \sin(\psi) \cos(\phi))
\end{bmatrix} \tag{48a}
\]

\[
h_{GPS_2} = \begin{bmatrix}
w X + X_2 \cos(\psi) - Y_2 \sin(\psi) \cos(\phi) \\
w Y + X_2 \sin(\psi) + Y_2 \cos(\psi) \cos(\phi) \\
-\omega_x(X_2 \cos(\psi) + Y_2 \sin(\psi) \cos(\phi)) \\
w_z(X_2 \cos(\psi) - Y_2 \sin(\psi) \cos(\phi))
\end{bmatrix} \tag{49a}
\]

\[
h_{GPS_1} = \frac{\partial h_{GPS_1}}{\partial x_1} \tag{48b}
\]

\[
h_{GPS_2} = \frac{\partial h_{GPS_2}}{\partial x_1} \tag{49b}
\]

\[
R_{GPS_1} = diag \left( \sigma^2_{\psi_{1x}} \sigma^2_{\psi_{1y}} \sigma^2_{\psi_{1z}} \right) \tag{48c}
\]

\[
R_{GPS_2} = diag \left( \sigma^2_{\psi_{2x}} \sigma^2_{\psi_{2y}} \sigma^2_{\psi_{2z}} \right) \tag{49c}
\]

\[
H_{AHRS} = \begin{bmatrix} 0 & 0 & 1 & 0 \end{bmatrix} \tag{50a}
\]

\[
R_{AHRS} = \sigma^2_\psi \tag{50b}
\]

\[
V_{AHRS} = 1 \tag{50c}
\]

In figure 10 is presented the simulation for a radial trajectory.

Since it is not available any solution using differential GPS techniques to serve as a reference, it is not possible to precise how good the solutions provided by the total filter are. However, by looking into a few zoomed areas in figure 12, the estimation seem smooth even in the presence of GNSS disturbances which may indicate the filter is performing correctly.
the need/acceptance of existence of forces to comply with movement rather than its full sources at tire level as described for instance in [3]. A vehicle in motion with velocity $b v$, with an angle of steering as seen in figure 13, will force the vehicle to turn. At infinitesimal level the small motion can be approximated as a small translation in the direction of body Y axis, $dv = b v dt$, and a small rotation about the center of rear axis, $d\psi = \omega_z dt$. Since a rotation body around a point as tangential velocity given by (51), it follows that center of mass must have an infinitesimal velocity in $v_x = -\omega L/2$

$$\frac{v}{R} = \omega$$  

(51)

If only the force causing rotation was applied in front axis, that would cause the vehicle to rotate about its center of mass with an angular acceleration equal to (52). This would induce a tangential acceleration in the rear axis equal to (53) in the opposite direction of $F_{r_x}$

$$\alpha_z = \frac{F_{r_x} \frac{L}{2}}{I_z}$$  

(52)

$$b a_{rrear} = \alpha_z \frac{L}{2}$$  

(53)

Since that is not the case, there must be a conjugation of forces in the rear axis, at tire level, that pull down the rotation towards to rear axis. It follows (54) and the new tangential acceleration in the rear axis is equal to (55)

$$\alpha_z = \frac{(F_{r_x} - F_{r_x}) \frac{L}{2}}{I_z}$$  

(54)

$$b a_{rrear} = \frac{(F_{r_x} - F_{r_x}) \frac{L^2}{2}}{I_z}$$  

(55)

At same time, the center of mass is accelerated by $b a_z$ as (57). At infinitesimal level, in order for rear axis to be fixed at same position during rotation results that must be a force in opposite direction of $b a_{rrear}$ with norm value equal to $b a_z$. The conjugation of all equations and the parallel axis theorem [20] results in (58) which solving in order to $I_z/m$ results in (59) where $I_0$ is the moment of inertia at center of rear axis $P_0$, $b a_z$ is the reading of accelerometer corrected for bias and gravity due to

7. **Yaw moment estimation**

When performing a curve trajectory, the vehicle will be subject to lateral force due to rotation which is relate to the vehicle’s inertial moment associated with body Z axis, $I_z$. The objective is to identify during trajectory, using the data provided by AHRS corrected for the bias estimated and compensated for suspension movement, locations where the car is clearly performing a turn and use it to: (i) estimate derivative of angular velocity in body Z axis, $\alpha_z$. (ii) estimate factor $\frac{L^2}{m}$ since the mass of vehicle (plus other sources of mass) is unknown using recursive least squares. It will be followed the implementation provided in [5] where it is described

![Figure 10: Complete filter simulation for a trajectory](image)

(a) 2D Position trajectory (b) Errors in trajectory simulation

![Figure 11: Complete filter real data from trajectory 1](image)

(a) 2D Position for real test (b) Estimatives for real test 1

![Figure 12: Complete filter real data from trajectory 1 - Zoom](image)

![Figure 13: Forces causing the vehicle rotation](image)
suspension movement and $\alpha_z$ is the approximated derivative given by (56).

$$\alpha_k = \frac{1}{T_a} (\omega_k - \omega_{k-1})$$  \hspace{1cm} (56)

$$\dot{\alpha}_z = \frac{F_{rx}}{m}$$  \hspace{1cm} (57)

$$I_0 = I_x + m \frac{L^2}{4}$$

$$F_{rx} = m \dot{\alpha}_z$$

$$F_{rz} = \frac{m \alpha - I_0 F_{rx}}{I_x}$$

$$\alpha_z = \frac{(F_{rx} - F_{rz})}{2}$$

$$I_x = \frac{-(L^3 \alpha_z + L^2)}{2 \alpha_z L^2 - 4 \dot{\alpha}_z}$$  \hspace{1cm} (59)

Rearranging (59) in a suitable form for applying the recursive least squares estimate, follows (60) where the error $e_k$ is the cost function that has to be minimized.

$$e_k = -(L^3 \alpha_z + L^2) - \frac{(2 \alpha_z L^2 - 4 \dot{\alpha}_z) I_x}{m}$$  \hspace{1cm} (60)

Figure 14 shows the same trajectory.

8. Conclusions

It is considered that proposed objectives of the work was attained. The solution for attitude estimation and respective sensor calibrations where reasonable for the application purpose, however there is room for improvement. In real data tests, without a better external reference like a high grade sensor solution, it is hard to quantify how good the results are. The only external reference available is heading Euler angle $\psi$ given the velocity vectors provided by GNSS and the AHRS produced similar results with the changes introduced to handle external accelerations and magnetic distortions. Same situation occurs for INS/GNSS fusion. Simulation produces good results, but there is lacking a precise position solution to provide a ground truth in real tests. It is possible to discuss the quality of solution in points where position signals are degraded like in zoomed areas in figure 12. Although signals from GNSS are inconsistent, the result is smooth and coherent which is an indication of the quality of results. As for yaw inertia moment factor, there is need for more investigation to provide a good conclusion. If compared to the typical cuboid approximation formula [19] used for vehicles, that factor would give $\approx 0.82$, if only distance distances between axes frames are used or $\approx 2.1$ if full dimensions of car are used. Assuming the estimated value is $\approx 4$, it is between two to five times greater, depending on the measurements used in cuboid formula. However it was assumed the center of mass was in the middle point distance between both wheel axes, but in reality, it was expected to be closer to the front part of the car due to location of the motor and two persons in front seats. Also, even with efforts, it is not granted that sensors were exactly aligned horizontally with axes. If recalling the parallel axes theorem [20] (first equation of (58)) the value grows with the square of distance between center of mass and the actual measurement axis.

9. Acknowledgments

I would like to thank Dr. João Sequeira for all the time spent and knowledge provided during the elaboration of this work.

References


[2] Analog Devices. MEMS Inertial Sen-


