REPORT OF THE COURSE “PROGETTO DI MICROELETTRONICA”

English reduced version

D. Comotti , M. Ermidoro

Contents

Introduction

This report is an english reduced version of the report published on the site project:

$$http : //code.google.com/p/9dof - orientation - estimation/$$

the project aims to achieve robust, correct and efficient orientation estimation algorithms using measurements provided by an IMU (Inertial Measurement Unit).

This brief report is organized as follows.
In the first chapter will be presented the Complementary Filter, the easiest way to get orientation from an IMU.
Chapter two will shows a second solution we have achieved, that is an implementation of a Kalman Filter based on quaternions.

The inertial measurement unit used in this project is the iNemo (iNErtial MOdule), produced by ST Microelectronics, release STEVAL-MKI06V2. This device is equipped with the following sensors:

- 3 Axis Gyroscope;
- 3 Axis Accelerometer;
- 3 Axis magnetometer;
- Temperature sensor;
- Pressure sensor.

For more information see [6, 1] and sensors datasheets.

This project is developed in cooperation with ST Microelectronics.

Chapter 1

Complementary Filter

1.1 Introduction

In this chapter a first algorithm briefly described in [3] will be presented. This algorithm is based on angles and it uses two static gain components to merge orientation achieved from gyroscope measurements with the one provided by accelerometer and magnetometer. Since these static gain components sum to 1, this kind of filter is named Complementary Filter.

This first solution uses angles as state variables and it has been developed in order to handle ±90 degrees only rotations. Indeed, it is easier to achieve good algorithms using angles and trigonometric functions, but this kind of approach suffer of laks due to the same angles (i.e. codomain of inverse trigonometric functions, angles fusion, gimbal lock, etc.).

1.2 Complementary Filter: Implementation

Complementary Filter uses two main components to estimate orientation:

- Gyroscope measurements;
- Accelerometer and magnetometer measurements.

Since datas of these two components are different each other, two methods has been used to estimate angles.

1.2.1 Accelerometer and Magnetometer Measurements

Measurements provided by accelerometer and magnetometer are first filtered through an IIR low pass filter. Then, roll and pitch angles in Euler convention can be calculated as follows [7, 5]:

$$\gamma = asin\left(\frac{A_y}{|\vec{g}|}\right);\ \varphi = asin\left(\frac{A_x}{|\vec{g}|}\right) \tag{1.1}$$

where A_y and A_x are filtered measurements of acceleration on y and x axis, respectively. Once γ and φ has been achieved, yaw angle, ψ, is computed by the following equation:

$$\begin{gathered} M_x = X_h \cdot cos(\phi) + Z_h \cdot sin(\phi); \\ M_y = X_h \cdot sin(\gamma) \cdot sin(\phi) + Y_h \cdot cos(\gamma) - Z_h \cdot sin(\gamma) \cdot cos(\phi); \\ \psi = atan2(M_y, M_x) \end{gathered} \tag{1.2}$$

where X_h, Y_h and Z_h are the magnetic components on x, y, z axis respectively. Finally, rotation angles computed from acceleromtere and magnetometer measurement, expressed in Euler convention are the following:

$$\vec{\Phi}(t) = \begin{bmatrix} \gamma(t) \\ \varphi(t) \\ \psi(t) \end{bmatrix} \tag{1.3}$$

1.2.2 Gyroscope Measurement

Gyroscope measurements can be integrated over time in order to get angles. However, angles achieved by this solution are expressed in the current frame; to get angles expressed in Euler convention the following equation has been used [7]:

$$\begin{bmatrix} \frac{\delta\psi}{\delta t} \\ \frac{\delta\varphi}{\delta t} \\ \frac{\delta\gamma}{\delta t} \end{bmatrix} = \frac{1}{cos\,(\gamma)} \cdot \begin{bmatrix} 0 & sin\,(\gamma) & cos\,(\gamma) \\ 0 & cos\,(\gamma)\,cos\,(\varphi) & -sin\,(\gamma)\,cos\,(\varphi) \\ cos\,(\phi) & sin\,(\gamma)\,sin\,(\varphi) & cos\,(\gamma)\,cos\,(\varphi) \end{bmatrix} \cdot \begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix} \tag{1.4}$$

where γ, φ and ψ are Euler angles achieved during the previous acquisition $(t-1)$. Euler angles related to gyroscope measurements can be simply obtained by the following equation:

$$\vec{\Phi}_{Gyro}(t) = \begin{bmatrix} \psi(t) \\ \phi(t) \\ \gamma(t) \end{bmatrix} = \begin{bmatrix} \psi(t-1) \\ \phi(t-1) \\ \gamma(t-1) \end{bmatrix} + \begin{bmatrix} \frac{\delta\psi}{\delta t} \\ \frac{\delta\phi}{\delta t} \\ \frac{\delta\gamma}{\delta t} \end{bmatrix} \cdot \Delta t \tag{1.5}$$

This integration operation is feasible since sampling period is small enough.

In figure 1.1 is shown an example of computing angles from gyroscope measurements.

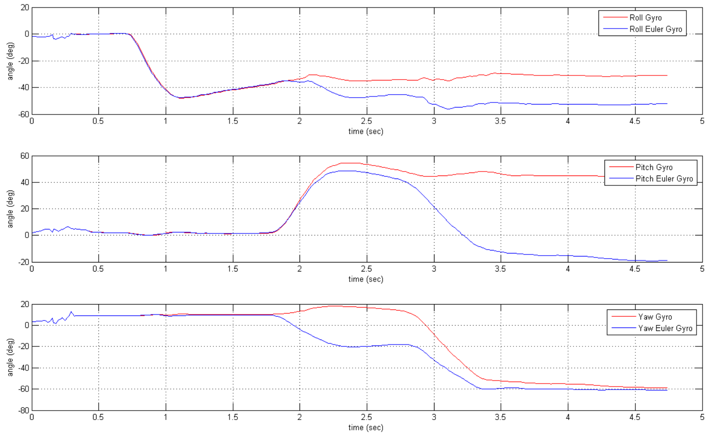

Figura 1.1: Roll, Pitch an Yaw rotations expressed in the current frame (red) and in Euler convention (blue).

1.2.3 Filtering Algorithm

Once the orientations from the system components are available and they are expressed in the same convention, filtering algorithm is defined taking into account the following behaviors:

- Fast changes related to gyroscope are reliable than those achieved by accelerometer and magnetometer.
- In order to avoid divergences due to gyroscope bias, steady convergence must be achieved using acceleromter and magnetometer measurements rather than gyroscope ones.

The merging equation is the following:

$$\hat{\vec{\Phi}}(t) = 0.98 \cdot \vec{\Phi}_{Gyro}(t) + 0.02 \cdot \vec{\Phi}(t) \tag{1.6}$$

The highest coefficient 0.98 allows the system to respond against fast changes in measurements. The second point of the previous list is achieved replacing equations 1.4 and 1.5 with the following ones:

$$\begin{bmatrix} \dfrac{\delta\psi_{gyroFilt}}{\delta t} \\ \dfrac{\delta\phi_{gyroFilt}}{\delta t} \\ \dfrac{\delta\gamma_{gyroFilt}}{\delta t} \end{bmatrix} = \frac{1}{cos\left(\hat{\gamma}\right)} \cdot \begin{bmatrix} 0 & sin\left(\hat{\gamma}\right) & cos\left(\hat{\gamma}\right) \\ 0 & cos\left(\hat{\gamma}\right) cos\left(\hat{\phi}\right) & -sin\left(\hat{\gamma}\right) cos\left(\hat{\phi}\right) \\ cos\left(\hat{\phi}\right) & sin\left(\hat{\gamma}\right) sin\left(\hat{\phi}\right) & cos\left(\hat{\gamma}\right) cos\left(\hat{\phi}\right) \end{bmatrix} \cdot \begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix} \tag{1.7}$$

$$\vec{\Phi}_{gyroFilt} = \begin{bmatrix} \psi_{gyroFilt}(t) \\ \phi_{gyroFilt}(t) \\ \gamma_{gyroFilt}(t) \end{bmatrix} = \begin{bmatrix} \hat{\psi}(t-1) \\ \hat{\phi}(t-1) \\ \hat{\gamma}(t-1) \end{bmatrix} + \begin{bmatrix} \frac{\delta\psi_{gyroFilt}}{\delta t} \\ \frac{\delta\phi_{gyroFilt}}{\delta t} \\ \frac{\delta\gamma_{gyroFilt}}{\delta t} \end{bmatrix} \cdot \Delta t \tag{1.8}$$

where $\hat{\gamma}$, $\hat{\phi}$, $\hat{\psi}$ are the filtered Euler angles of $\hat{\vec{\Phi}}$. In this way, a feedback has been introduced in the system, and equation 1.6 become the following:

$$\hat{\vec{\Phi}}(t) = 0.98 \cdot \vec{\Phi}_{gyroFilt}(t) + 0.02 \cdot \vec{\Phi}(t) \tag{1.9}$$

This feedback allows to get a low response both to accelerometer and magnetometer dinamics and to steady convergence of gyroscope. As shown in , time constant of the system is the following:

$$\tau = \frac{0.98 \cdot \delta t}{0.02}$$

where δt is the sampling period. In our case, $\delta t = 0.2\,sec$, therefore $\tau \simeq 1\,sec$. With respect to accelerometer and magnetometer datas, τ represents the time constant of a LP filter, while it acts quite like an HP filter for gyroscope measurements.

1.2.4 Block Diagram

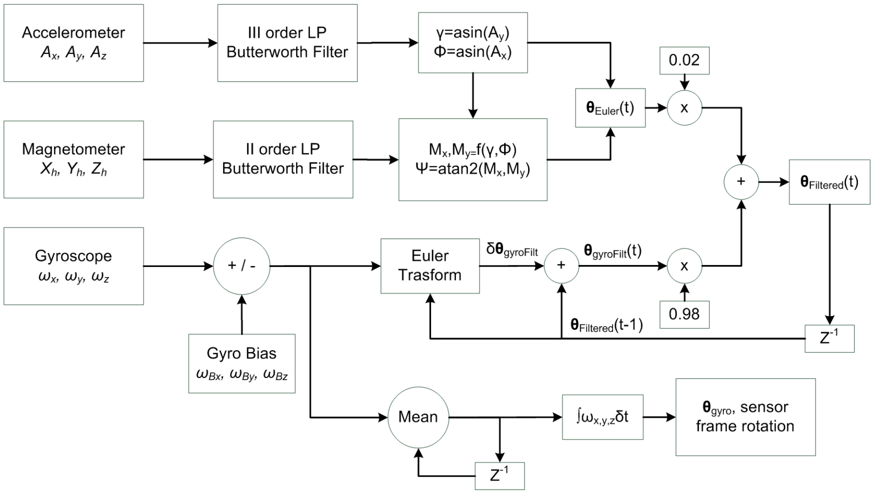

Figure 1.2: Block diagram of the Complementary Filter.

1.3 Results

With regards to experimental results, two kind of measurements have been executed:

- Static measurements, during which precision of Complementary Filter has been compared with the real orientation (achieved using an inclinometer). Results are shown in figures 1.3, 1.4 and 1.5.

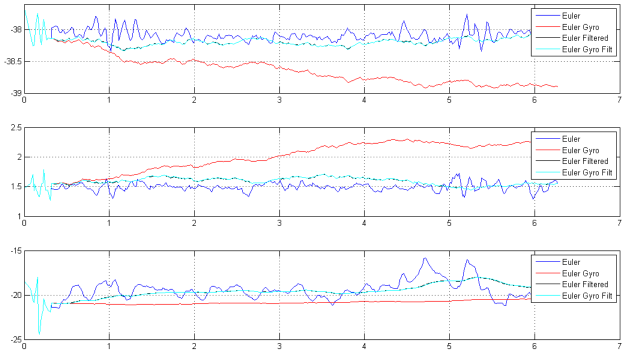

Figura 1.3: Roll angle experiment: 36.3 degrees on Roll angle.

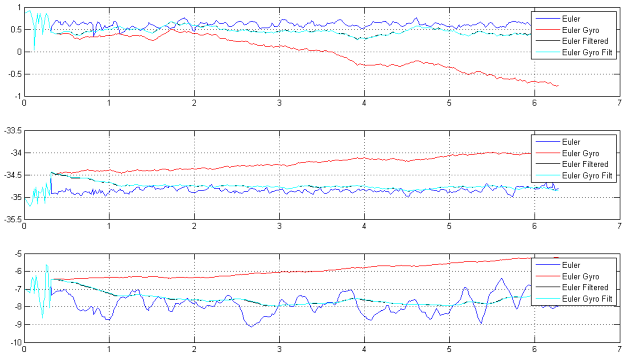

Figura 1.4: Pitch angle experiment, $-36,1$ degrees on Pitch angle.

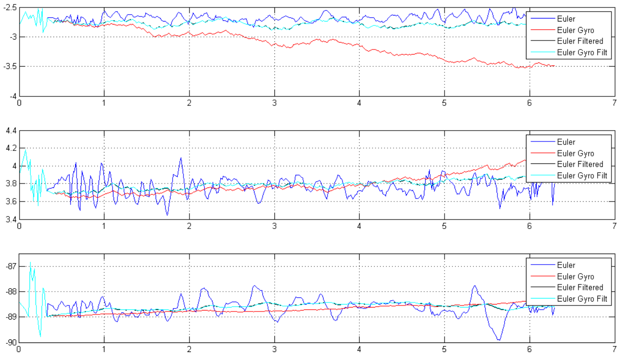

Figura 1.5: Yaw angle experiment, -90 degrees on Yaw angle.

- Dynamic measurements, which tested time response of the filtering algorithm. The goal of the experiment was to compare Complementary filter with Kalman Filter. Anyway, we post Complementary filter results only, shown in figures 1.6, 1.7 and 1.8.

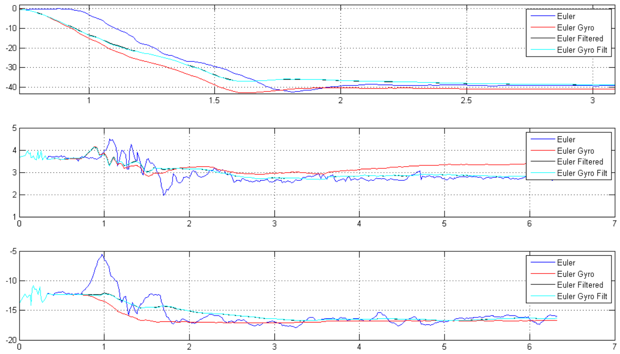

Figura 1.6: 37 degrees rotation around x axis.

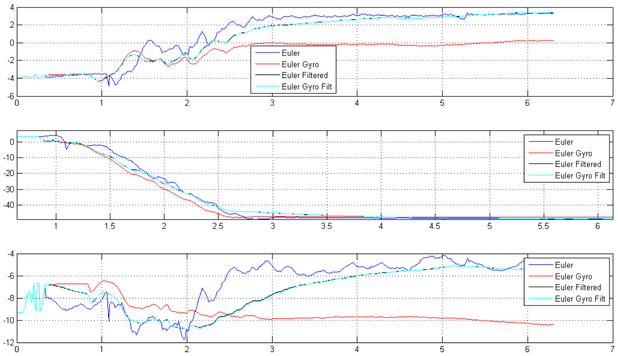

Figura 1.7: -46 degrees rotation around y axis.

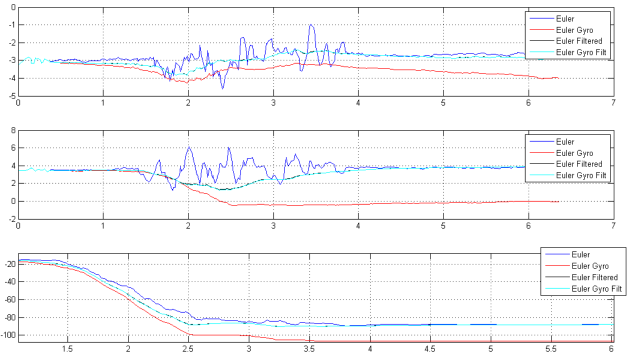

Figura 1.8: -90 degrees rotation around z axis.

Chapter 2

Quaternion Kalman Filter

2.1 A brief introduction to Kalman Filter

Kalman Filter is an efficient recoursive filter that estimate an unobservable noisy state. There's a lot of stuff on the internet which treat this orientation problem ([4, 5, 6, 7]), however we decided to implement our specific version of this filter. To represent orientation in 3D space quaternions have been used ([7, 8]).

Kalman filter uses a system's dynamics model (i.e., physical laws of motion), known control inputs to that system, and measurements (such as from sensors) to form an estimate of the system's varying quantities (its state) that is better than the estimate obtained by using any one measurement alone. As such, it is a common sensor fusion algorithm (Wikipedia).
The model of the generic dynamic system is shown in 2.1

$$\begin{cases} \vec{x}_t = F \cdot \vec{x}_{t-1} + B \cdot \vec{u}_{t-1} + \vec{w}_{t-1} \\ \vec{z}_t = H \cdot \vec{x}_t + \vec{v}_t \end{cases} \tag{2.1}$$

Where $\vec{w_t} \sim N(\mu_w, \sigma_w^2)_{t-1}$is the state innovation and $\vec{v_t} \sim N(\mu_v, \sigma_v^2)$ the state observation noise.
The Kalman filter is based on two steps:

1. Prediction step: it predicts the state on the basis of the previous realization and using the matrix F wich represent the state evolution matrix.

2. Update step: The state is updated using the observation $\vec{z_t}$.

2.2 Quaternion Kalman Filter: Implementation

We decided to implement the filter in the simplest way, so we choose as state of the system the values of quaternion components, as shown in 2.2, where q_1 is the real-part while q_2, q_3 and q_4 are the three immaginary-part of the quaternion.

$$\vec{x} = \begin{bmatrix} q_1 \\ q_2 \\ q_3 \\ q_4 \end{bmatrix} \tag{2.2}$$

we'll now proceed to describe in deatil the main features of our implementation of this filter.

2.2.1 Prediction Step

As shown in [4, 6, 7] the equation that links the angular velocity to quaternion variation is shown in 2.3

$$\dot{q} = \frac{1}{2} \cdot \vec{x} \otimes \vec{\omega} \tag{2.3}$$

So we integrate over time (δt is the sample period) the variation shown in 2.3 in order to predict the value of the state, using the following equation:

$$\vec{x}_t = \vec{x}_{t-1} + \dot{q}_t \cdot \delta t = F \cdot \vec{x}_{t-1} \tag{2.4}$$

From these considerations we define the matrix F, the state transition model:

$$F = \begin{bmatrix} 1 & -\frac{1}{2} \cdot \omega_x \cdot \delta t & -\frac{1}{2} \cdot \omega_y \cdot \delta t & -\frac{1}{2} \cdot \omega_z \cdot \delta t \\ \frac{1}{2} \cdot \omega_x \cdot \delta t & 1 & \frac{1}{2} \cdot \omega_z \cdot \delta t & -\frac{1}{2} \cdot \omega_y \cdot \delta t \\ \frac{1}{2} \cdot \omega_y \cdot \delta t & -\frac{1}{2} \cdot \omega_z \cdot \delta t & 1 & \frac{1}{2} \cdot \omega_x \cdot \delta t \\ \frac{1}{2} \cdot \omega_z \cdot \delta t & \frac{1}{2} \cdot \omega_y \cdot \delta t & -\frac{1}{2} \cdot \omega_x \cdot \delta t & 1 \end{bmatrix} \tag{2.5}$$

To model the process noise $\vec{w}$ we estimated the variance of the gyroscope on each axis and then we computed the Q matrix, simply considering the axis used in the estimation. 2.6

$$Q = E\left[\begin{bmatrix} -\omega_x - \omega_y - \omega_z \\ \omega_x - \omega_y + \omega_z \\ \omega_x + \omega_y - \omega_z \\ -\omega_x + \omega_y + \omega_z \end{bmatrix} \cdot \begin{bmatrix} -\omega_x - \omega_y - \omega_z & \omega_x - \omega_y + \omega_z & \omega_x + \omega_y - \omega_z & -\omega_x + \omega_y + \omega_z \end{bmatrix}\right] \tag{2.6}$$

Assuming $E[\omega_i] = 0$ (gyroscope offsets have been calculated and removed from signals in order to have a very small bias, $0.1 \div 0.2\, ^{deg}/_{s}$) and $E\left[\omega_i \cdot \omega_j\right] = 0,\ \forall i \neq j$, the innovation covariance Q matrix can be computed as follows:

$$Q = \begin{bmatrix} \sigma_x^2+\sigma_y^2+\sigma_z^2 & -\sigma_x^2+\sigma_y^2-\sigma_z^2 & -\sigma_x^2-\sigma_y^2+\sigma_z^2 & \sigma_x^2-\sigma_y^2-\sigma_z^2 \\ -\sigma_x^2+\sigma_y^2-\sigma_z^2 & \sigma_x^2+\sigma_y^2+\sigma_z^2 & \sigma_x^2-\sigma_y^2-\sigma_z^2 & -\sigma_x^2-\sigma_y^2+\sigma_z^2 \\ -\sigma_x^2-\sigma_y^2+\sigma_z^2 & \sigma_x^2-\sigma_y^2-\sigma_z^2 & \sigma_x^2+\sigma_y^2+\sigma_z^2 & -\sigma_x^2+\sigma_y^2-\sigma_z^2 \\ \sigma_x^2-\sigma_y^2-\sigma_z^2 & -\sigma_x^2-\sigma_y^2+\sigma_z^2 & -\sigma_x^2+\sigma_y^2-\sigma_z^2 & \sigma_x^2+\sigma_y^2+\sigma_z^2 \end{bmatrix} \tag{2.7}$$

At this point the prediction equation can be defined by 2.8, where $\vec{x}_{t|t-1}$is the prediction vector and $\vec{x}_{t-1|t-1}$ is the output of the filter at the previous step (the filtered state).

$$\vec{x}_{t|t-1} = F \cdot \vec{x}_{t-1|t-1} \tag{2.8}$$

2.2.2 Observations

The observations consist of four components of a quaternion, as the state of the system, which are achieved from accelerometer and magnetometer signals. As shown in [6], to compute the orientation quaternion using accelerometer and magnetometer measurments an optimization problem can be solved, in order to have the quaternion which minimized an error. Such a problem is often a minimization one, which can be solved using Gradient Descent or Gauss-Newton methods. We implement both the methods and comparison of results is shown in [9].

In order to reduce errors due to electromagnetic interferences, the magnetic compensation shown in [6] has been also developed while magnetometer calibration helped us to avoid measurement errors[10].

H matrix is the observation model which maps the true state space into the observation space. In our case H is an identity just because the states space corresponds to the observations space (they're both the quaternion representing current 3D space orientation).

2.2.3 Update

At this step the filter updates the predicted state $\vec{x}_{t|t-1}$, using the observations just computed. This update is performed by mean of a weighted sum as shown in 2.9,where K is the Kalman filter gain calculated using equation 2.10.

$$\vec{x}_{t|t} = \vec{x}_{t|t-1} + K \cdot \left(\vec{x}_t - \vec{x}_{t|t-1}\right) \tag{2.9}$$

$$K = P_{t|t-1} \cdot H \cdot \left(H \cdot P_{t|t-1} \cdot H^T + R\right)^{-1} \tag{2.10}$$

where $P_{t|t-1}$ is the prediction error:

$$P_{t|t-1} = \left(F \cdot Q \cdot F^T\right) + Q \tag{2.11}$$

The last matrix we haven't showed yet is the observations covariance matrix, R. Since the signal of accelerometer or magnetometer are not directly involved as for the state model, we decide to try several value for this matrix in order to acheve the best results. In the literature R is often represented as a $I_{4\times4} \cdot 0.001$; however, since R matrix is involved in the computing of the kalman gain we decide to use a $I_{4\times4} \cdot 0.05$ in order to avoid small oscillation around the optimal point, due to a too small variance.

2.2.4 Initialization

In order to get good results it is important to choose right values during the algorithm initialization. At the first step of iterations, Kalman filter needs an initial value of the state updated and update variance and covariance matrix, P_0. For what concerne the initial state we decided an arbitrary value which must only be normalized, $\begin{bmatrix} 0.5 & 0.5 & 0.5 & 0.5 \end{bmatrix}$. The initial value of P_0 has been chosen after several tuning operations, and in the end we used $P_0 = I_{4\times4} \cdot 2$. With this condition, and without esternal stress, the filter reach a steady state after 5-6 seconds (TODO PRESTAZIONI CON NUOVO GAUSS NEWTON).

2.2.5 Block diagram

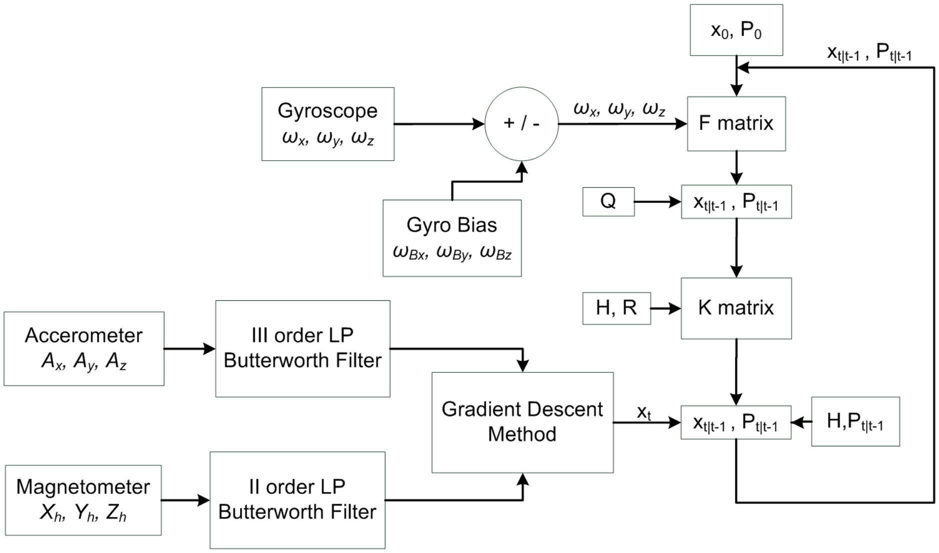

Figura 2.1: BLock diagram of the quaternion Kalman filter.

2.3 Results

In figure 2.2 Kalman Filter quaternions output is shown, while in figure 2.3 trend of the related $Roll - Pitch - Yaw$ angles is shown.

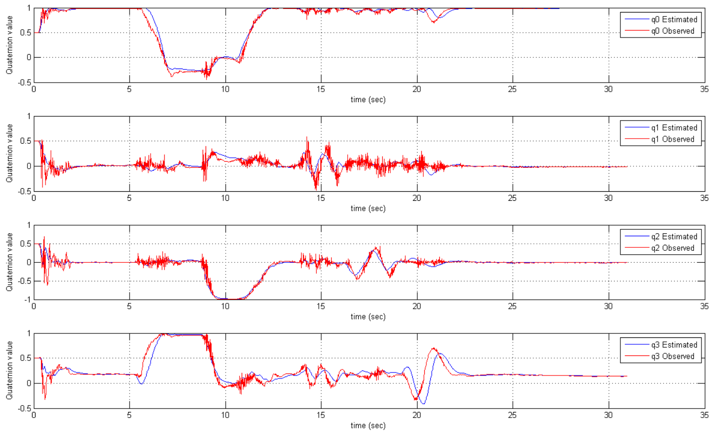

Figura 2.2: In red we can see the results of the Gradient Descent method (the observation). In blue the putput of the Kalman Filter (the state updated). The output of the Gradient Descent presents the typical oscillations around the convergence point, this fluctuations are removed by the filter. In the first 2/3 seconds the filter reach the steady state.

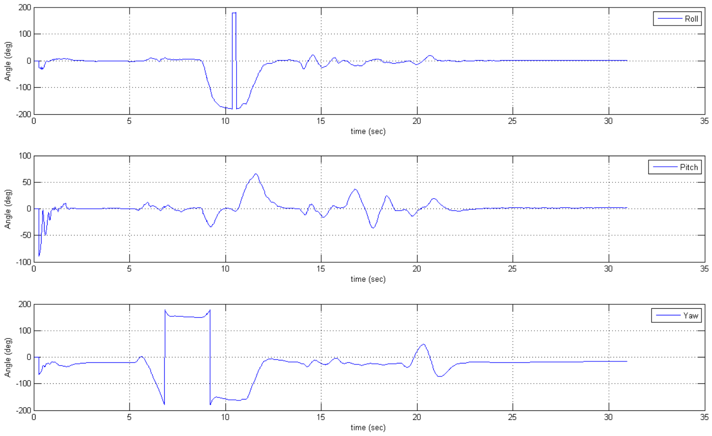

Figura 2.3: Angles with the RPY convenctions associated to quaternion of figure 2.2. The *iNemo* is initially rotated of 180° around z-axis, then about 180° around x-axis and then 180° around y. The last rotation is less visible because it's represented as composition of three rotations. After these rotations there's other roll, pitch, yaw operations.

With regards to comparison between Kalman filter and Complementary filter, table 2.1 shows some experimental results.

	Real Meas.	Complem. Filter [°]	Kalman Filter [°]	Complem. Filter Error [°]	Kalman Filter Error [°]
Roll	36.3	38.0	35.2	1.7	1.1
Pitch	-36.1	-34.9	-36.2	1.2	-0.1
Yaw	-90.0	-88.6	-91.0	1.4	-1.0

Table 2.1: Complementary filter and Kalman filter precision comparison.

Conclusions

The aim of this report is to provide an english and brief version of topics treated in [11], in order to provide an alternative paper to be consulted. We have spent little attention to results because they are already shown in form of figure in [11] and because, with respect to the same [11], more improvements has been achieved during the month following its release [9].

Bibliography

[1] ST Microelectronics, “STEVAL-MKI062V2 User Manual”.

[2] ST Microelectronics, “STEVAL-MKI062V2 Data brief”.

[3] Shane Colton, “The Balance Filter A Simple Solution for Integrating Accelerometer and Gyroscope Measurements for a Balancing Platform”.

[4] João Luís Marins, Xiaoping Yun, Eric R. Bachmann, Robert B. McGhee, and Michael J. Zyda, An Extended Kalman Filter for Quaternion-Based Orientation Estimation Using MARG Sensors.

[5] Demoz Gebre-Egziabher, Gabriel H. Elkaim, J. D. Powell, Bradford W. Parkinson, A gyrofree quaternion based attitude determination system suitable for implementation using low cost sensors.

[6] Sebastian O.H. Madgwick, An efficient orientation filter for inertial and inertial/magnetic sensor arrays.

[7] DEMOZ GEBRE-EGZIABHER, ROGER C. HAYWARD , J. DAVID POWELL, Design of Multi-Sensor Attitude Determination Systems.

[8] Verena Elisabeth Kremer, Quaternions and SLERP.

[9] D. Comotti, “Orientation Estimation Based on Gauss-Newton Method and Implementation of a Quaternion Complementary Filter”.

[10] Michael J. Caruso, “Applications of Magnetoresistive Sensors in Navigation Systems”.

[11] D. Comotti, M. Ermidoro, “Relazione del Corso di Progetto di Microelettronica”.