

MÁSTER OFICIAL EN INGENIERÍA DE COMPUTADORES Y REDES

Departamento de Arquitectura y Tecnología de Computadores

Universidad de Granada



PROYECTO FIN DE MÁSTER

Monitorización y captura del movimiento del cuerpo humano

Realizado por:

Alberto Olivares Vicente

Dirigido por:

Dr. Gonzalo Olivares Ruiz

Dr. Dr. Juan Manuel Górriz Sáez

Dr. Javier Ramírez Pérez de Inestrosa

MASTER IN COMPUTER ENGINEERING AND NETWORKS

Departament of Computer Architecture and Computer Technology

University of Granada



MASTER THESIS

Monitoring and Capture of Human Body Motion

Written by:

Alberto Olivares Vicente

Supervised by:

Dr. Gonzalo Olivares Ruiz

Dr. Dr. Juan Manuel Górriz Sáez

Dr. Javier Ramírez Pérez de Inestrosa

Acknowledgements

Foremost I would like to thank Dr. Gonzalo Olivares Ruiz, Dr. Juan Manuel Górriz Sáez and Dr. Javier Ramírez Pérez de Inestrosa for supervising this work, their valuable comments and suggestions and specially for their continuous support and motivation. I would also like to express my gratitude to Dr. Francisco Gómez Mula for designing the hardware employed in this project and also for his time to help me to configure and use it properly.

Special thanks go to the members of the Signal Processing and Biomedical Applications (SIPBA), who are a role model for me to become a successful researcher, for being such a nice group of people.

I am deeply grateful to Mr. Dennis Huning for his contribution as a software engineer to the project, and also to Mr. Sascha Poggemann for giving life with his code to the new inertial measurement unit prototype and, not less important, for accompanying me in summertime (with its high temperatures) at the office.

Contents

Acknowledgements	i
1 Introduction	7
1.1 General vision	7
1.2 Project motivation	8
1.3 State of the art	9
1.4 Published works	11
1.5 Structure of the document	12
2 Inertial Measurement Units	13
2.1 Inertial sensors	14
2.1.1 Accelerometers	14
2.1.2 Gyroscopes	18
2.1.3 Magnetometers	22
2.1.3.1 MEMS Based Magnetometers	22
2.2 Examples of IMUs	23

2.2.1	Wagyro	23
2.2.2	WagyroMag	24
2.2.2.1	Detailed components list	24
2.2.3	Memsense's Bluetooth IMU	27
2.3	Focus of research	28
3	Datalogging Software	29
3.1	VisIMU	30
3.2	WagyLog	30
3.3	Focus of research	31
4	Calibration Procedures	33
4.1	Introduction, actual trends and classification	33
4.2	Error analysis	35
4.2.1	Accelerometers	35
4.2.2	Gyrosopes	35
4.2.3	Magnetometers	37
4.3	Sensor modeling	40
4.4	Calibration before use	42
4.4.1	Accelerometers	42
4.4.2	Gyrosopes	43
4.4.2.1	Threshold-based static detection	44
4.4.2.2	Detection based on the acceleration magnitude	44
4.4.3	Magnetometers	48
4.5	Results	54
4.5.1	Accelerometer	54
4.5.2	Gyroscope	55
4.6	Focus of research	59

5 Position Monitoring	61
5.1 Coordinate system	61
5.2 Angle computation methods	62
5.2.1 Computation using accelerometers	62
5.2.1.1 Pros and cons	63
5.2.2 Computation using gyroscopes	64
5.2.2.1 Pros and cons	64
5.2.3 Computation using accelerometers and magnetometers	65
5.2.3.1 Pros and cons	66
5.2.4 Sensor fusion	66
5.2.4.1 Definition	67
5.2.4.2 Kalman filter	67
5.2.4.3 Kalman filter sensor fusion approach	69
5.2.4.4 Least Mean Squares filters	70
5.2.4.5 Recursive Least Squares filters	72
5.2.4.6 LMS/RLS Filter Sensor Fusion Approach	75
5.2.4.7 Experiments	76
5.2.4.8 Results discussion	81
5.3 Focus of research	85
6 Applications of the System	87
6.1 Telerehabilitation	87
6.2 Gait and Posture analysis	88
6.3 Analysis of activities of daily life	89
6.4 Detection of seizures	89
6.4.1 Epilepsy	89
6.4.2 Parkinson	90

6.5 Sleep disorders	91
6.6 Sport science	91
6.7 Focus of research	91
7 Conclusions	93
A Wagyro: Schematics and Layouts	95
B Wagyromag: Schematics and Layouts	99
B.1 Schematics	100
B.2 Layouts	104
C VisIMU User Manual	109
C.1 Running the program	109
Bibliography	119

List of Figures

2.1	Generic structure of a moving mass accelerometer	15
2.2	Single axis accelerometer	15
2.3	Physical model of a moving mass accelerometer	15
2.4	Bode's Diagram of the accelerometer frequency response.	17
2.5	Diagram of the generic structure of MEMS accelerometers.	17
2.6	Resistive accelerometer architecture diagram	19
2.7	Capacitive accelerometer architecture diagram	19
2.8	Piezoelectric accelerometer architecture diagram	19
2.9	Basic structure of a vibrating mass gyroscope	20
2.10	Close-up of the internal structure of a vibrating tuning fork gyroscope .	20
2.11	Vibrating Plate and Tuning Fork Coriolis Gyroscopes.	21
2.12	Vibrating Beam and Vibrating Shell Coriolis Gyroscopes.	21
2.13	Concept of the MEMS flux concentrator	23
2.14	Internal and external appearance of WAGYRO	24
2.15	Internal appearance of Wagyromag.	26

2.16 External appearance of MemSense's Bluetooth IMU.	27
2.17 MemSense's Bluetooth IMU coordinate system.	27
3.1 VisIMU's user interface	31
3.2 WagyLog's Frontal Panel	32
4.1 The Ideal Aerosmith Model 2003HP three-axis rate table	34
4.2 Registered raw gyroscope data in static position	36
4.3 Registered raw gyroscope data in static position (zoomed)	36
4.4 Integration of registered raw gyroscope data in static position	37
4.5 Magnetic sensor output temperature variation	39
4.6 Ferrous object disturbance in uniform field.	39
4.7 Magnetic sensor outputs with no disturbances	39
4.8 Hard iron offsets when rotated horizontally in the earth's field.	40
4.9 Soft iron distortion when rotated horizontally in the earth's field.	40
4.10 Orthogonalization of the sensor frame k	41
4.11 IMU positioning to transform raw data into acceleration units (g)	43
4.12 Diagram of the experimental setup to count wheel turns	46
4.13 Acceleration effects on a tuning fork gyroscope.	47
4.14 Effects of magnetometer errors on the ideal field sphere.	49
4.15 Peak detection of the accelerometer signal	55
4.16 Wheel's turn periods obtained from the positive spin	56
4.17 Angular velocity obtained with accelerometer data (positive spin).	56
4.18 Calibration line calculated for the X sensitivity axis.	57
4.19 Study of dynamic bias variation during static position.	58
4.20 Dynamic bias correction using the moving average method.	58
4.21 Dynamic bias correction using the acceleration magnitude method.	59
5.1 Navigation Angle positioning system. θ =pitch, ϕ =roll, ψ =yaw.	62

5.2	Gravity components decomposition to obtain the pitch angle.	63
5.3	Gravity components decomposition to obtain the roll angle.	63
5.4	Effects of dynamic bias on the integrated gyroscope output signal. . . .	65
5.5	Kalman filter algorithm computational diagram.	69
5.6	Diagram of the Kalman filter sensor fusion approach.	70
5.7	Diagram of the LMS/RLS filter sensor fusion approach.	75
5.8	Synth. gyro and acc. angle signals (low intensity)	77
5.9	Synth. gyro and acc. angle signals (high intensity)	78
5.10	Average MSE of low intensity synthesized signals	78
5.11	Average MSE of low intensity synthesized signals (zoom)	79
5.12	Average MSE of high intensity synthesized signals	79
5.13	Average MSE of high intensity synthesized signals (zoom)	80
5.14	Acc. and gyro. angle signals. Knee rehab. exercise	82
5.15	Acc. and gyro. angle signals. Vertical jumps	83
5.16	Adaptive filtering output. Knee rehab. exercise	83
5.17	Adaptive filtering output. Vertical jumps	84
5.18	Adaptive filtering output. Vertical jumps (zoom)	84
6.1	Diagram of a mobile teleassistance system based on inertial sensors. . .	90
A.1	Schematics of Wagyro.	96
A.2	Components layout.	97
B.1	Schematics of Wagyromag (1/4).	100
B.2	Schematics of Wagyromag (2/4).	101
B.3	Schematics of Wagyromag (3/4).	102
B.4	Schematics of Wagyromag (4/4).	103
B.5	Components layout: top.	104
B.6	Components Layout: bottom.	105

B.7	Conductive traces layout: top	106
B.8	Conductive traces layout: bottom.	107
C.1	VisIMU booting up.	110
C.2	Main window of VisIMU	111
C.3	Selection of packages to be read	111
C.4	Connection configuration parameters	112
C.5	Start to read data from port reader	113
C.6	Read Source dialog window	114
C.7	Read process tab	115
C.8	Data exporting wizzard	116
C.9	Data writing parameters	117
C.10	File including the stored sensor data	117
C.11	Plotter preferences	118

List of Tables

4.1	Measured raw accelerometer data	54
5.1	Corrections for Heading calculation with magnetometer	66
5.2	Average MSE of 3 periods. Low intensity signals	80
5.3	Average MSE of 3 periods. High intensity signals	81

CHAPTER 1

Introduction

1.1 General vision

Human body motion has been a subject of deep study since the Thirties when Kinesiology was created. Kinesiology, from the Greek words *kinesis* (movement) and *kinein* (to move), also known as human kinetics, is the science of human movement. The advances of technology have allowed the development of tools that capture and analyze the movement of human body with a high degree of accuracy. This way, human body monitoring is present in many science fields. Within the last decade, human body motion monitoring has originated a wide field of research where experts of different study areas are taking part. Telecommunication engineers, programmers, electronic engineers, physics, doctors and other specialists are working together in different projects that are based on the study of human motion.

When analyzing human motion, acceleration and angular velocity registers from different body parts such as limbs, head and trunk are of crucial importance. Therefore, inertial sensors such as accelerometers and gyroscopes play an important role to obtain information of human body kinetics.

The use and development of inertial sensors has shown an exponential peak due to the wide range of applications, including inertial navigation systems [32], [62], au-

tomotive industry, robots and positioning devices [9], [36], and human body motion monitoring systems used in e-health solutions such as gait and posture analysis [30] and [19], telerehabilitation [58], fall detection [10] and daily activities analysis [40].

The advances in Microelectromechanical Systems (MEMS) devices manufacturing technologies have allowed to reduce considerably the price and size of sensors, making them more attractive to researchers and companies. However MEMS inertial sensors still present several undesired responses leading to poor performance if not handled properly.

Every complete human motion monitoring system has a hardware and a software part. In order to capture movement, Inertial Measurement Units (IMUs) are needed. They can either be bought choosing among the several existing commercial devices, [1, 2], or designed and then built based on the needs of the application. In the latter, we will be able to choose the quality of the sensors, the autonomy of the battery, the inclusion of memory and embedded processors and the data transmission technology depending on the constraints and the budget. Like in every electronic instrumentation system, it is mandatory to perform a calibration process to smooth the undesired effects present at the output of the sensors, due to their nature, and to convert the raw analogical or digital data into data expressed in physical units.

Once the system is designed, built and calibrated, we can start processing the gathered signals to obtain the desired measure in positioning systems, i.e. the tilt angle formed by the body part being monitored. A series of algorithms and mathematical tools need to be employed to complete this task in an accurate way.

1.2 Project motivation

This project emerged from the collaboration framework established between the Department of Computer Architecture and Computer Technology of the University of Granada, Spain, and Telefónica R&D Granada. Knowing from the experience in designing sensor networks and embedded systems of some of the members of the department, more specifically, Dr. Francisco Gómez Mula and Dr. Gonzalo Olivares Ruiz, Telefonica R&D asked them to develop a prototype of a wireless Inertial Measurement Unit (IMU). This unit had to be oriented to monitor human body movement for telerehabilitation applications. In a first stage, a prototype being able to capture knee movement and determine its position should be designed.

Once the project was running and the first stage was almost finished, I had the opportunity to join it and carry out the final project in order to get the Master Degree in Telecommunication Engineering.

During several months I developed software tools to gather and visualize data from the IMUs in real time, calibration algorithms for the sensors, and specially signal processing algorithms aiming to improve the accuracy of the measurements. Furthermore,

I made a wide study of the state of the art to observe the research trends inside the human body movement monitoring field, with the objective of finding possible applications for the developed system.

After finishing the project and getting the Master Degree, I kept working together with the aforementioned members of the department, to develop a new prototype including new features such as local data storage and the inclusion of built-in microcontrollers and new sensors in the IMU. At the same time I continued researching about new calibration methods to improve the sensor's outputs and I focused on looking for new alternatives of adaptive filtering to fuse data from different sensors. As a result of such research, we took part in several international conferences to present our work.

This document is, therefore, the culmination of the work carried out during the last year, and tries to compile all the necessary information to develop a complete system to monitor human body motion. It also has the objective of being presented as the final project of the Official Master in Computer Engineering and Networks organized by the Department of Computer Architecture and Computer Technology of the University of Granada, Spain.

After one and a half year of working as an engineer in the field of human body monitoring, I am still motivated to continue researching due to the wide range of applications of this field inside the medicine. After achieving the objective of developing a system that is able to measure and determine the position of different parts of the human body in real time, as well as offline during longer sessions, we can start to apply it in multiple medicine disciplines, hence, contributing to improve the quality of life of many people. Further research is planned in form of a PhD thesis with the objective of obtaining a Doctor degree.

1.3 State of the art

Many authors have published works in the field of human body position monitoring with different applications. We can classify them in categories attending to the structure of the present work:

- *IMU design:* Some authors use commercial devices, but there is also a high number of published works that include the design of their own prototype IMUs. By designing the IMU the sensor specifications can be chosen, we can add processors, choose how the data is sent and, more importantly, we can reduce the cost of the project, as prototypes are usually many times cheaper than commercial units. The prototypes present in literature normally combine MEMS accelerometers, gyroscopes and magnetometers depending on the degree of freedom (DOF) that is required by the application. Examples of different designs can be found in [59], [49], [7], [13], [63] and [22].

- *Calibration procedures:* There exist many different procedures to calibrate the sensor attending to the application and the type of sensor (accelerometer, gyroscope or magnetometer). We can also choose to perform robust calibration before use, or less accurate but automatic in-use algorithms. Depending on the application we will use different error models attending to the degree of accuracy we must achieve. Models including complete identification and quantification of different errors will require more complex algorithms that may not be suitable for real time applications or calibration carried out by unexperienced operators. [21] was the first work to include a complete error modeling of accelerometers and gyroscopes and to propose a calibration procedure without using complex equipment. This work inspired many authors during the following years that used it as a starting point of their research. The next remarkable advance came with [54] that proposed a multi-position calibration algorithm for triaxial gyroscopes and accelerometers. This work was extended in [57]. For magnetometers the work that has been taken as a reference in the last decade [37] introduces the ellipsoid fitting parameter estimation calibration concepts. Several papers [23], [26] have used variations of this method since its publication. The choice of a calibration algorithm depends mainly on three factors: the available equipment (such as rate tables, robot arms), the sensor being calibrated (accelerometers, gyroscopes or magnetometers), and the degree of precision required by the system where the sensors will be used.
- *Position computation:* As previously said, one of the main applications of human body monitoring is determining the position of the IMU, either for inertial navigation (airplanes and other vehicles) or to determine the position of the part of the human body where the IMU is attached. Inertial navigation has been a complement for GPS navigation to improve the robustness of navigation systems and increase their accuracy. Many works [53] are published presenting systems using inertial sensors to determine the attitude and position, where [60] stands out as the main handbook for inertial navigation.
On the other hand, in the field of human body position monitoring, [18] developed a system based on gyroscopes to evaluate the knee joint of patients with anterior cruciate ligament lesions. They used a wired IMU, and a sensor fusion strategy to calculate tilt angles. [38] developed a system to evaluate the impact of neuromuscular disorders affecting the upper extremities through the analysis of arm orientation in daily activities. [22] built a human body capture system with accelerometers and magnetometers for general purpose. [47] also developed sensor fusion techniques for data gathered from magnetometers, accelerometers and gyroscopes.

1.4 Published works

Part of the work presented here has been published and is already available for the research community:

International Conferences:

1. Alberto Olivares , Gonzalo Olivares, J. M. Gorriz, y J. Ramirez, 'High-efficiency Low-cost Accelerometer-aided Gyroscope Calibration', International Conference on Test and Measurement, Hong Kong, China, December 2009. vol.1, pp 354–360, ISBN: 978-1-4244-4699-5,
2. A. Olivares, J.M. Gorriz, G. Olivares, J. Ramirez, P. Gloesekoetter, 'A Study of Vibration-Based Energy Harvesting in Activities of Daily Living', published at 4th International ICST Conference on Pervasive Computing Technologies for Healthcare, ISBN: 987-963-9799-89-9, Munich, Germany, March 2010.
3. Alberto Olivares, J.M. Gorriz, J. Ramirez, Gonzalo Olivares, 'Accurate Human Limb Angle Measurement in Telerehabilitation: Sensor fusion through Kalman, LMS and RLS Adaptive Filtering', Ambient Inteligence and Future Trends - International Symposium on Ambient Intelligence (ISAMI 2010), Guimaraes, Portugal, June 2010. ISBN: 978-3-642-13267-4. pp 97–104,
4. Alberto Olivares, J.M. Gorriz, J. Ramirez, Gonzalo Olivares, 'Sensor Fusion Adaptive Filtering for Position Monitoring in Intense Activities', 5th International Conference on Hybrid Artificial Intelligence Systems (HAIS 2010), San Sebastian, Spain, June 2010. ISBN: 978-3-642-13768-6. pp 484-491

National Conferences:

1. Alberto Olivares, Gonzalo Olivares, J. M. Gorriz, J. Ramirez, 'Wagyromag: red inalambrica de sensores iniciales para el registro y procesamiento de movimientos corporales', I Simposio en Computación Empotrada, Congreso Español de Informática (CEDI), Valencia, Spain, September 2010.

Book chapters:

1. Alberto Olivares, Gonzalo Olivares, J. M. Gorriz, J. Ramirez, 'Human Body Position Monitoring' Chapter 16 of Recent Advances on Biomedical Signal Processing, Bentham Science Publishers, pp 77 – 92, 2010.

Articles in International Magazines:

1. *Waiting for approval:* A Olivares, J M Gorriz, J Ramirez and G Olivares. 'Accurate Human Limb Angle Measurement: Sensor fusion through Kalman, LMS and RLS Adaptive Filtering' Measurement Science and Technology. Institute of Physics Publishing.

1.5 Structure of the document

After the introduction, the reader will go through the following sections: Chapter 2 presents the general structure of Inertial Measurement Units (IMUs), as well as the working principles of the inertial sensors they include. The employed IMUs for this work are also described.

Chapter 3 shows the developed software to gather and store the data being transmitted in real time from the IMUs. Different sensor calibration procedures are explained and tested in chapter 4, along with a novel accelerometer-aided gyroscope calibration procedure.

Different strategies to calculate the position of the IMUs are explained in chapter 5, including sensor fusion approaches to improve measurement accuracy. Chapter 6 presents a series of applications that are already on test as well as other possible applications of the developed system. To end, conclusions are drawn in chapter 7.

Instead of having a *Future Work* chapter, every chapter ends with a subsection called *Focus of research* that gives the key aspects and sets the bases for further research on the concepts developed throughout the chapter. Such research will hopefully be presented in the forthcoming PhD thesis.

CHAPTER 2

Inertial Measurement Units

The miniaturization of the inertial sensors derived from the development of the Micro-electromechanical Systems (MEMS) technology has permitted the creation of Inertial Measurement Units (IMUs) that are suitable to be placed on human body. Therefore, inertial sensors that used to be heavy and bulky can now be used in human body movement and activity measurement applications due to their low size and weight.

Inertial Measurement Units (IMUs) are usually composed of inertial sensors, i.e. accelerometers, gyroscopes, sometimes magnetometers and embedded electronics that allow the transmission and storage of the gathered data. They are used to measure movement and position of persons, vehicles, robots, etc. IMUs are usually designed as sensor nodes that can be included in a sensor network. The data can be sent through wires or wirelessly using Bluetooth or IEEE 802.15.4 systems. There exist many different commercial IMUs each one offering different kinds of sensors, processors and transmission attending to their price [1, 2].

2.1 Inertial sensors

2.1.1 Accelerometers

Accelerometers sense the movement of an object in an inertial space. Their behavior differs slightly from displacement sensors, since the latter measure the relative movement between two points. However, it is essential to use an element being able to sense the displacements that are produced inside the accelerometer, so a measure of displacement or force is obtained. This measure will be used in turn to calculate the resultant acceleration.

The generic structure of an accelerometer is composed by a seismic mass m which is joint to the internal surfaces by an elastic structure in such a way that it moves with the same acceleration suffered by the test object. This structure can be seen in Figure 2.1. The mass can only move in one direction, which is set to be the sensitive axis of the accelerometer. As we can observe in Figure 2.2, the displacement of the mass in the direction of \hat{n} is proportional to the difference between the dynamic acceleration \vec{a} and the gravity static acceleration \vec{g} . The variation of the displacement will vary the voltage at the output terminals. Such a variation is interpreted as the acceleration suffered by the object, to which the sensor is attached, after a proper calibration process.

The structure and basic working principles of a displacement based accelerometer can be modeled as a system composed by a mass joint to a spring and a resort (see Figure 2.3). The resultant displacement of the mass responding to an applied acceleration is modeled as a second order system. Therefore, the system model can be obtained as a direct application of Newton's Law:

$$u(t) = m \frac{\partial^2 x(t)}{\partial t^2} + b \frac{\partial x(t)}{\partial t} + kx(t) \quad (2.1)$$

where $x(t)$ is the relative displacement of the mass m , b is the damping constant, k is the elastic constant of the spring and $u(t)$ is the applied force.

By applying the Laplace Transform on (2.1) we can calculate the system's transfer function, which input and output are the force $u(t)$ and the relative displacement $x(t)$ respectively, as follows

$$u(s) = x(s)[ms^2 + bs + k] \quad (2.2)$$

$$H(s) = \frac{x(s)}{u(s)} = \frac{1}{ms^2 + bs + k} = \frac{1/m}{s^2 + \frac{b}{m}s + \frac{k}{m}} \quad (2.3)$$

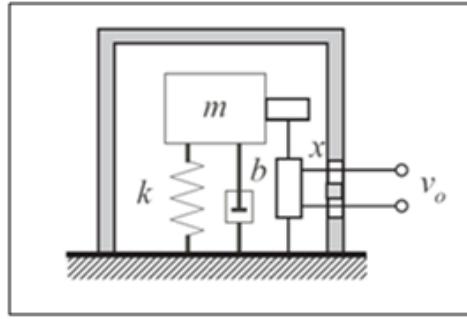


Figure 2.1: Generic structure of a moving mass accelerometer. A mass m is joint to the internal surface by a spring having an elastic constant k and a resort with a damping constant b . The displacement x causes variations in the output voltage V_o .

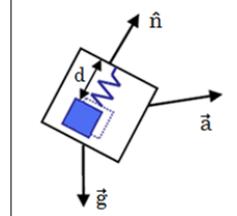


Figure 2.2: Single axis accelerometer. A mass is hanging attached to a spring. Distance d between the mass and the casing is measured as a function of the acceleration and the direction of gravity with respect to the direction of the measured displacement. Unit vector \hat{n} represents the sensitive axis of the sensor.

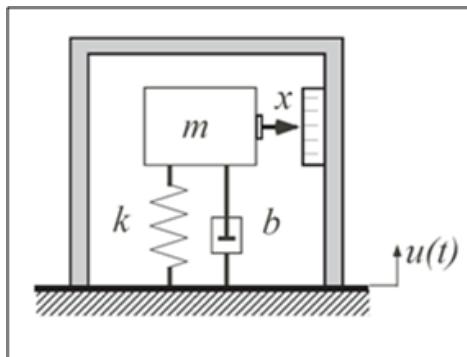


Figure 2.3: Physical model of a moving mass accelerometer. The force $u(t)$ causes a relative displacement x of the mass m , which is hanging attached to a spring with an elastic constant k and a resort with a damping constant b .

which can be finally expressed as:

$$H(s) = \frac{K_e \omega_n^2}{s^2 + 2\xi\omega_n s + \omega_n^2} \quad (2.4)$$

where $K_e = \frac{1}{k}$, $\omega_n = \sqrt{k/m}$ is the natural oscillation frequency of the spring and $\xi = b/2\sqrt{k/m}$ is the damping coefficient.

We can equally express the equations in the system's state space. For this purpose we assume the input of the system to be a sinusoidal force:

$$u(t) = u_0(t) \sin(\omega t) \quad (2.5)$$

Hence, the differential equation representing the system will be:

$$m\ddot{x}(t) + b\dot{x}(t) + kx(t) = u(t) \quad (2.6)$$

$$\ddot{x}(t) = \frac{1}{m}(-kx(t) - b\dot{x}(t) + u(t)) \quad (2.7)$$

What leads to the following system's state equation:

$$\begin{pmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ \frac{-k}{m} & \frac{-b}{m} \end{pmatrix} \begin{pmatrix} x_1(t) \\ x_2(t) \end{pmatrix} + \begin{pmatrix} 0 \\ \frac{1}{m} \end{pmatrix} u(t) \quad (2.8)$$

where $x_1(t) = x(t)$ and $x_2(t) = \dot{x}(t)$. The resulting output equation is:

$$y(t) = \begin{pmatrix} 1 & 0 \end{pmatrix} \begin{pmatrix} x_1(t) \\ x_2(t) \end{pmatrix} \quad (2.9)$$

If we represent the system's transfer function Bode Diagram ($m = 10\text{grams}$, $b = 0.05$ and $k = 300\text{N/m}$), we can clearly observe that the accelerometer has an accurate response for all frequencies under $\omega = 0.5\omega_n$, which for the previous values is:

$$\omega = 0.5\sqrt{\frac{k}{m}} = 0.5\sqrt{\frac{300}{0.01}} = 173.2 \text{ rad/s.} \quad (2.10)$$

that, as depicted on Figure 2.4, matches with the frequency value where the phase changes abruptly. Frequency values over this threshold there will likely lead to a remarkable difference between the actual acceleration and the sensed value.

Once the accelerometer's generic structure and model are described, we will do a brief description of the different kinds of accelerometers, attending to its size and physical sensing principle used to measure the mass displacement, that can be found in the market. According to size criteria, accelerometers can be classified as MEMS and non MEMS.

Microelectromechanical (MEM) devices are, as its name suggests, systems that combine mechanical elements with electronic components which are produced in micrometric or nanometric scale. This technology is applied to accelerometers so to obtain a minimization of its size, allowing them to be integrated into a circuit. Their structure is based on the generic system previously presented with some slight modifications.

The basic structure of a MEMS accelerometer consists of a suspended mass fixed to a substrate by suspension arms (see Figure 2.5). This kind of accelerometers offer a reasonable price-performance trade-off, yet their precision and robustness is many times lower than non MEMS accelerometers. Non MEMS accelerometers can't be integrated in a circuit due to their size, so they are used in applications where size is not a constraint such as industrial environments, aeronautic systems, and other systems needing great precision and robustness.

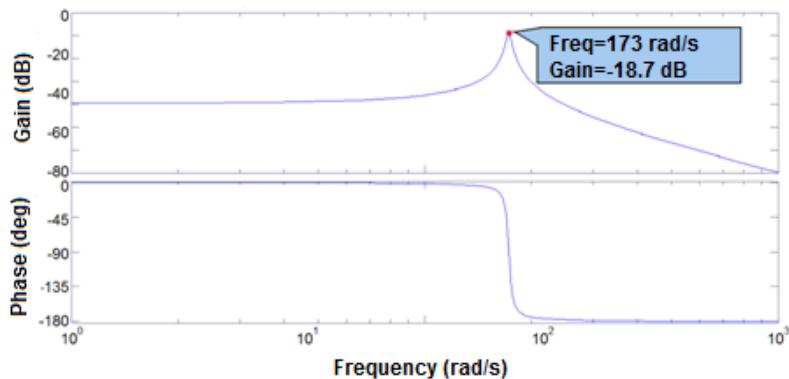


Figure 2.4: Bode's Diagram of the accelerometer frequency response.

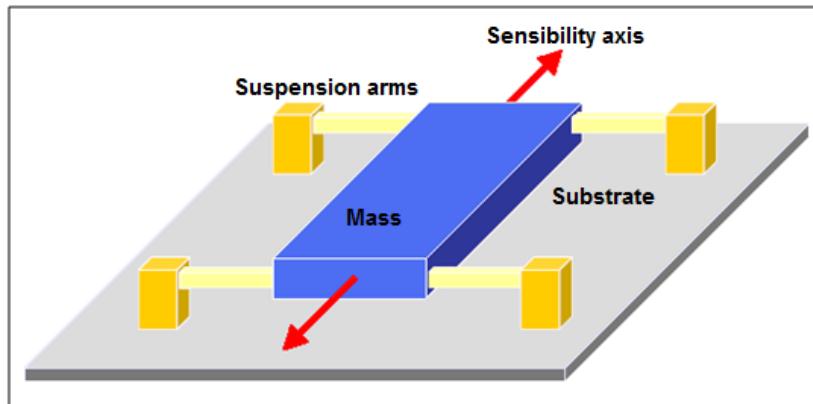


Figure 2.5: Diagram of the generic structure of MEMS accelerometers.

According to the mass displacement sensing method, accelerometers can be classified as resistive, capacitive, and piezoelectric. There exist other less popular sensing methods such as the ones based on optic fiber, servoaccelerometers and vibrating elements that will not be explained here.

Resistive accelerometers sense the mass displacement through a brush which moves along a potentiometer. The system may use magnetic, gas or viscous damping in order

to minimize the the acoustic noise generated by the contact of the brush with the potentiometer. Potentiometer-based resistive accelerometers typically have a frequency range from 20 to 60 Hz , depending on the stiffness of the spring, and develop a high level output signal. Their frequency response is lower than most of the accelerometers, usually varying from 15 to 30 Hz . A diagram of their architecture can be seen in Figure 2.6.

Capacitive accelerometers sense the mass displacement through variation of the capacity between two electrodes that are fixed to the internal surface and the moving mass respectively. Figure 2.7 shows a diagram of their architecture.

Piezoelectric accelerometers are normally used in applications where vibrations need to be measured, and in shock detection devices (such as cars' airbags). The piezoelectric crystals (usually quartz or ceramic) generate an electric charge when the accelerated mass applies a force on them. The quartz layers are precharged so there is a positive or negative variation when the applied force produces a change in the electric charge. Even though the sensitivity of the piezoelectric accelerometers is rather low compared to other types, they present the largest dynamic range of operation, being able to measure accelerations up to 1000 g , as well as a frequency response higher than 20 kHz . Figure 2.8 shows a diagram of their architecture.

2.1.2 Gyroscopes

A gyroscope is a device which is essentially formed by a rotation symmetry object that spins around its symmetry axis. They are mainly used for measuring and maintaining the orientation of an object. When used in IMUs, gyroscopes are used to measure angular velocity. Gyroscopes used for this purpose are based on the Coriolis Effect.

The Coriolis Effect originates the relative acceleration that is suffered by and object, which is moving inside a non inertial rotating reference system, when there is a variation of its distance with respect to the spin axis. The Coriolis Force causes an acceleration or desacceleration of the object moving along the radius with respect to the rotating system, that causes a variation in its trajectory. The Coriolis acceleration can be computed applying the following expression:

$$\vec{a}_c = -2\vec{\omega} \times \vec{v} \quad (2.11)$$

where $\vec{\omega}$ is the angular velocity of the rotating reference system and \vec{v} is the velocity of the moving object with respect to the rotating reference system.

Due to their miniaturization, it is impossible to build a MEMS gyroscope based on the classic rotating disc structure. We need then, a structure based on the Coriolis effect which is feasible to be built in nanometric scale.

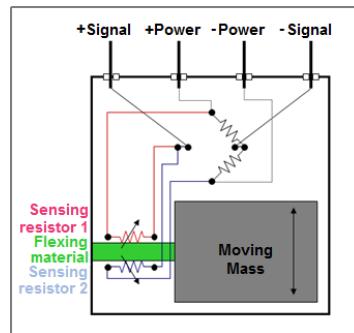


Figure 2.6: Resistive accelerometer architecture diagram. The flexion due to the movement of the mass causes variations in the sensitive resistors that are translated to variations in the output signal.

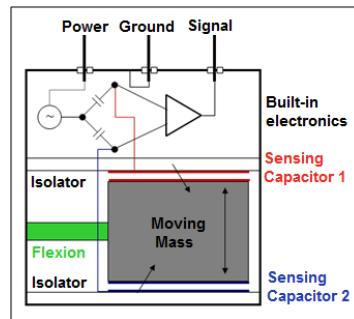


Figure 2.7: Capacitive accelerometer architecture diagram. The flexion due to the movement of the mass causes variations in the sensitive capacitors that are translated to variations in the output signal.

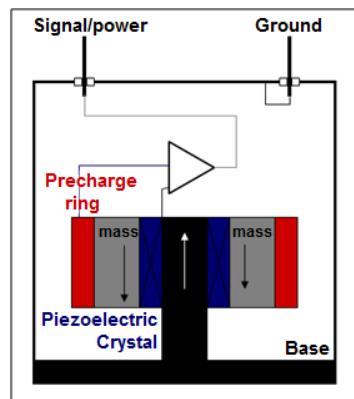


Figure 2.8: Piezoelectric accelerometer architecture diagram. The flexion of the crystals due to the movement of the mass causes variations in the electric charge that are translated to variations in the output signal.

Vibrating mass gyroscopes are the most usual employed MEMS gyroscopes. Their basic concept consists of a mass which is suspended by two arms that permit its oscillation. When an angular velocity is applied on the device, the mass, which is vibrating, will experiment a displacement caused by the Coriolis effect. If the displacement is known, then, after proper calibration, the angular velocity can be measured. The vibration of the mass is set so it moves with a velocity \vec{v} , allowing the apparition of Coriolis Effect. This vibration is usually produced by a piezoelectric crystal that expands and contracts under a variation of voltage. Figure 2.9 shows the basic diagram of the vibrating mass gyroscopes.

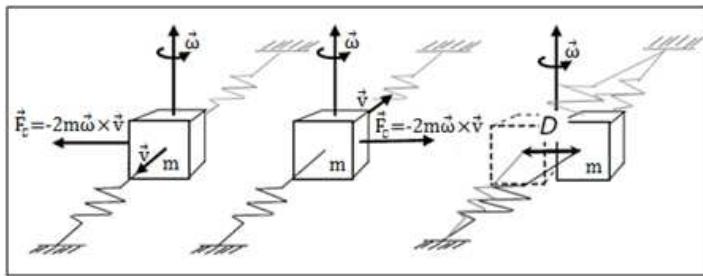


Figure 2.9: Basic structure of a vibrating mass gyroscope. The vibration of the mass causes it to move with a speed \vec{v} in each one of the oscillation senses (left and middle figures), which makes the Coriolis Force cause a displacement D (figure on the right).

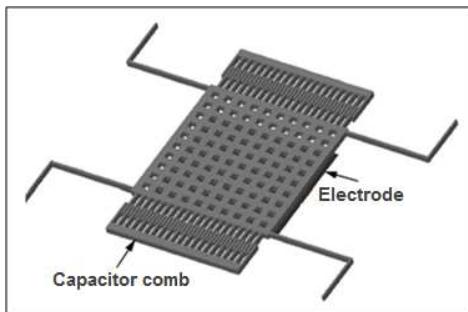


Figure 2.10: Close-up of the internal structure of a vibrating tuning fork gyroscope. A comb of capacitors measures the displacement caused by the Coriolis Effect.

A Coriolis vibratory gyro (CVG) is a gyroscope based on the coupling of a structural, driven, vibrating mode into at least one other structural mode (pickoff) via Coriolis acceleration, where the Coriolis force arises from the motion of the vibrating structure relative to the CVG case fixed frame that is rotating relative to inertial space. Figures 2.11 and 2.12 depict the various types of CVG that have been commercially constructed.

Tuning fork gyroscopes are the most common vibrating mass MEMS gyroscopes. As its name says, they are based on a tuning fork, which arms are made vibrate, that

VIBRATING PLATES

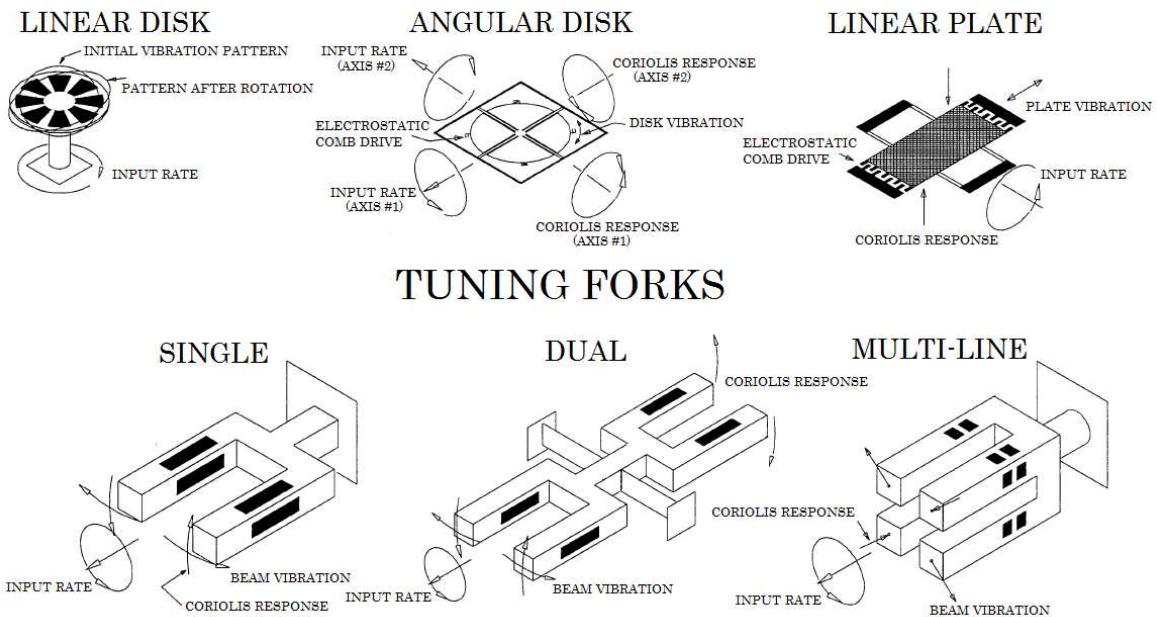
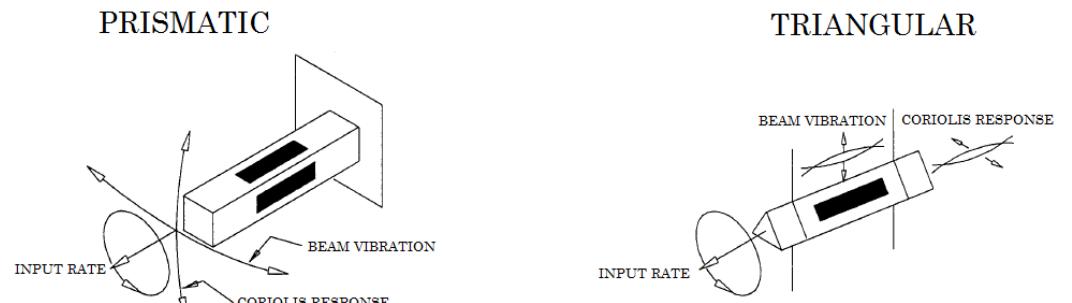


Figure 2.11: Vibrating Plate and Tuning Fork Coriolis Gyroscopes.

VIBRATING BEAMS



VIBRATING SHELLS

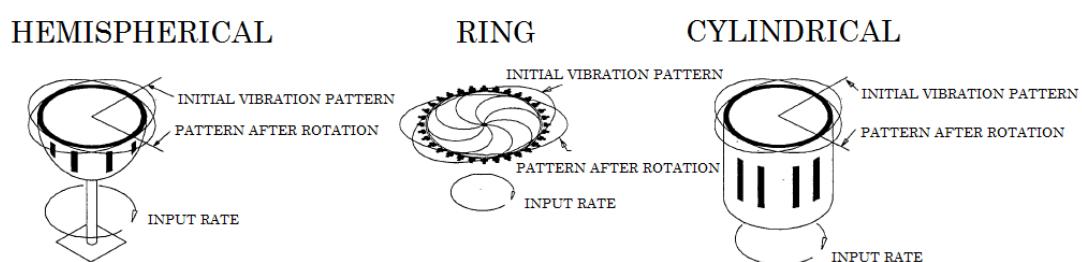


Figure 2.12: Vibrating Beam and Vibrating Shell Coriolis Gyroscopes.

spins with the same angular velocity applied to the sensor. An array of capacitors forming a comb is attached to the central part of the tuning fork in order to measure the displacement of the arms. As in the case of the capacitive accelerometers, the variation of the capacity is used to measure the displacement and, consequently, the angular velocity. Such configuration is depicted on Figure 2.10.

2.1.3 Magnetometers

Magnetic sensors can be classified according to whether they measure the total magnetic field or the vector components of the magnetic field. There are many ways to sense magnetic fields, most of them based on the intimate connection between magnetic and electric phenomena. Magnetic sensing techniques exploit a broad range of ideas and phenomena from the fields of physics and material science. Another important consideration is whether the magnetometer measures just the magnitude of the field or measures each of its vector components.

2.1.3.1 MEMS Based Magnetometers

With the development of microelectromechanical systems (MEMS), the idea of using movement to sense magnetic fields is being reexamined. Fabricating these devices has turned out to be challenging. This is especially true if the fabrication process requires using different technologies that are not naturally compatible. For example, the use of HF (Hydrogen Fluoride), often required to perform the release step needed to fabricate the MEMS structure, can damage other parts of the sensor.

Most of these sensors use the Lorentz force. An example of this is a magnetometer based on detecting the motion of a miniature bar magnet. The hard magnetic material used was deposited by electrodeposition. The choice of materials for the hard magnet was limited by the need to use HF in the release step. The bar magnetic responds to the field without drawing any power.

MEMS technology can improve magnetic sensors by minimizing the effect of noise. The concept for a device that can accomplish this, the MEMS flux concentrator [15], [16], is shown in figure 2.13. In the device, the flux concentrators composed of soft magnetic material, are placed on MEMS flaps. The flux concentrators enhance the field. Decreasing the separation between the flaps increases the enhancement. The two MEMS flaps are forced to oscillate by applying an AC voltage to the electrostatic comb drives. By tuning the frequency, one can excite the normal mode in which the distance between the flaps oscillates. The resonant frequency for the MEMS structure is designed to be about 10 $kH\ddot{z}$. The oscillation of the MEMS flaps modulates the field at the position of the sensor and, hence, shifts the operating frequency of the sensor above the frequency where noise dominates. Depending on the type of magnetic sensor used, this shift in operating frequency should increase the sensitivity of magnetometers

by one to three orders of magnitude.

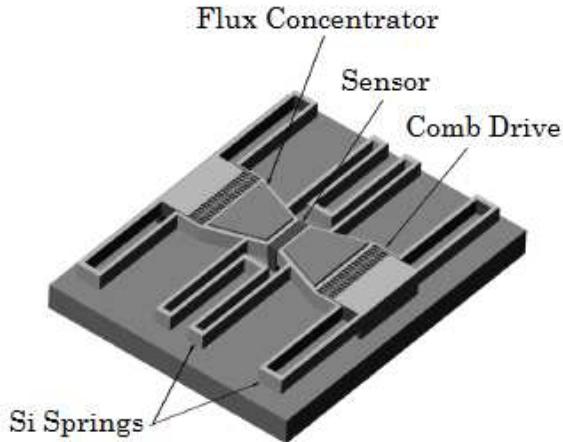


Figure 2.13: Concept of the MEMS flux concentrator. Note that there is a space between the substrate and the flux concentrators on the MEMS flaps.

2.2 Examples of IMUs

2.2.1 Wagyro

WAGYRO (Wireless Accelerometer and GYROscope) is the first prototype IMU developed in the Department of Computer Architecture and Technology of the University of Granada, Spain. It was originally designed to be used in Telefonica's R&D Rehabitic project. This project, currently in test at the Sant Joan de Deu Hospital in Mallorca, Spain, and the Esperança Hospital in Barcelona, Spain, allows patients to perform their rehabilitation routines at home following the instructions of a real time application that analyzes human limbs motion data. Initially, in the first phase, the IMUs are placed on thighs and shanks to obtain the knee joint angle, acceleration and angular velocity of the lower limbs. WAGYRO includes an Analog Devices ADXL330 triaxial accelerometer with a range of $\pm 3g$ and an InvenSense IDG-300 biaxial gyroscope with a theoretical range of $\pm 500^\circ/s$. Data are sampled at a rate of $50Hz$ and sent using Digi's XBee module based on *IEEE802.15.4*. Analog signals are converted with a 10 bit ADC, so raw data will be bounded to a $(0, 1023)$ range. Figure 2.14 shows WAGYRO's internal and external appearance.



Figure 2.14: Internal (left) and external (right) appearance of WAGYRO. Size is $40 \times 80 \times 20\text{mm}$ and weight is 58g .

2.2.2 WagyroMag

WagyroMag (Wireless Accelerometer, GYROscope and MAGnetometer) is the new prototype of IMU developed in the Department of Computer Architecture and Technology. It improves the previous version by adding new sensors and features. More specifically, now we have a triaxial gyroscope, a triaxial magnetometer, a temperature sensor and the data can be stored into a microSD card or be transmitted in real time again by the XBee chip. There is also a PIC24FJ64GB004 microcontroller that provides local process capability, a buzzer, and a 1 Mb serial F-RAM memory. It operates at 3.3 V and uses a battery, which can be charged via USB or an external 5 V power supply.

2.2.2.1 Detailed components list

- *PIC24FJ64GB004*: The *PIC24FJ64GB004* is a 44 Pin, 16 Bit, Flash Microcontroller with USB On-The-Go (OTG) and nanoWatt XLP Technology [31].

The *PIC24FJ64GB004* Features are:

- 44-Pins (total Pin Count)
- 64 kBytes of Program Memory
- 8 kByte SRAM
- 25 remappable Pins
- 5 Timers Modules (16 Bit)

- 2 UART w/IrDA Interfaces
- 2 SPI Interfaces
- 2 I2C Interfaces
- 10 Bit A/D converter, 13 Channels
- 1 RTCC Module
- 1 USB OTG (On-The-Go)

The CPU includes a modified Harvard Architecture with up to 16 MIPS @ 32 MHz, an 8 MHz internal oscillator - PLL up to 96 MHz with multiple divide options, and an C Compiler optimized instruction set architecture with only 76 base instructions and flexible addressing modes. Additional features are an 17-times-17 Bit Single-Cycle Hardware fractional/integer multiplier an 32-by-16 Bit hardware divider and a 16-times-16 Bit working register array [31].

Different power management modes are supported, for example, the Deep-Sleep-Mode allows to nearly shut down the Microcontroller whereas it still has the ability to wake up on external Trigger, WDT or RTTC alarm. The regular Sleep-Mode shuts down the peripherals and the core for power saving but offers a fast wake-up.

The operating Voltage can vary from 2.0 V to 3.6 V, but the digital input pins are all 5 V tolerant. Microchip warrants a minimum of 10.000 erase/write cycle endurance and 20–years of data retention. In–Circuit Serial Programming (ICSP), In–Circuit Debugging (ICD) via 2 PINS and JTAG Boundary Scan are supported.

- *ADXL335*: The ADXL335 is a small, low-power, 3-axis ± 3 g accelerometer. Its operating voltage can vary from 1.8 V to 3.6 V consuming $350 \mu\text{A}$ (typical). It is 10.000 g shock resistant and temperature stable ($\pm 0.01\%/\text{ }^{\circ}\text{C}$). The ADXL335 emits an analog voltage for each axis which goes from 0 V to 3 V representing ± 3 g.
- *LPR550AL*: The LPR550AL is a Micro-Electro-Mechanical System (MEMS) gyroscope with dual axis pitch and roll functionality produced by ST Microelectronics. Basically it is a combination out of one actuator and one accelerometer for each axis. The operating voltage can vary from 2.7 V to 3.6 V consuming 6.8 mA. It is used to measure angular velocity in x – *axis* and y – *axis*.
- *LY550ALH*: The LY550ALH is a high performance low-power single-axis micro-machined gyroscope able to measure angular rate along yaw axis. The LY550ALH has a full scale of $\pm 500^\circ/\text{s}$ and is capable of detecting rates with a -3dB bandwidth up to 140Hz . It is used to measure angular velocity in z – *axis*.

- *FM24V10*: The FM24V10 is an 1 Mb Serial 3 V F-RAM Memory IC with a fast two-wire serial interface (I2C). The maximum bus frequency is 3.4 MHz. The operating voltage can vary from 2.0 V to 3.6 V and the current depends on the bus clock frequency. $175 \mu\text{A}$ @ SCL = 100 kHz, $400 \mu\text{A}$ @ SCL = 1 MHz, $1000 \mu\text{A}$ @ SCL = 3.4 MHz.
- *HMC5843*: The HMC5843 is a 3-axis digital compass IC produced by Honeywell. It includes an I2C module for communication with a Microcontroller, which supports standard (100 kHz bus frequency) and fast mode (400 kHz bus frequency). The operating voltage can vary from 2.5 V to 3.3 V. In measurement mode the device consumes 0.9 mA and in sleep mode (single supply) 110 μA . The measurement range is ± 4 Gauss.
- *MCP9700A-TT*: The MCP9700A-TT is a tiny analog temperature sensor with a temperature range from -40°C to $+125^\circ\text{C}$. The accuracy is stated with a maximum of $\pm 2^\circ\text{C}$ (0°C to $+70^\circ\text{C}$). This IC is optimized for analog-to-digital converters with an output of $10m\text{V}/^\circ\text{C}$. Its operating voltage can vary from 2.3V to 5.5V consuming 6 μA (typical).
- *Micro-SD card*: A general micro-SD card is used for data storage. Any micro-SD card up to 4GB can be employed.

Figure 2.15 shows the appearance of Wagyromag out of its protection case. This new prototype allows monitoring sessions lasting more than 24 hours (in storage mode), allowing, therefore, a wide range of possible medical applications.



Figure 2.15: Internal appearance of Wagyromag.

2.2.3 MemSense's Bluetooth IMU

The Bluetooth IMU manufactured by MemSense [2] offers measurements of 3D acceleration, 3D angular rate of rotation and 3D magnetic field. The range of transmission can be affected by the environment, but typical range capabilities are on the order of 30 m. There exist different IMU versions attending to the measurement range of its sensors. Ours has a range of $\pm 600^\circ/s$, $\pm 5g$ and $\pm 1.25\text{Gauss}$. The maximum bandwidth is 50 Hz and the maximum sampling rate is 150 Hz. It has a lower autonomy than the Wagyro family prototypes as it uses a bluetooth connection instead of a 802.15.4 based connection which has a much lower power consumption. This is a drawback as it does not allow long monitoring sessions. A 9 V battery allows up to 4 hours of run time. Figure 2.16 shows the external appearance of MemSense's Bluetooth IMU. Dimensions are 5.423 cm \times 4.2799 cm \times 1.109 cm without the battery. Adding a battery will considerably increase the thickness. Figure 2.17 depicts the coordinate system of the IMU. It follows the right hand rule convention.



Figure 2.16: External appearance of MemSense's Bluetooth IMU.

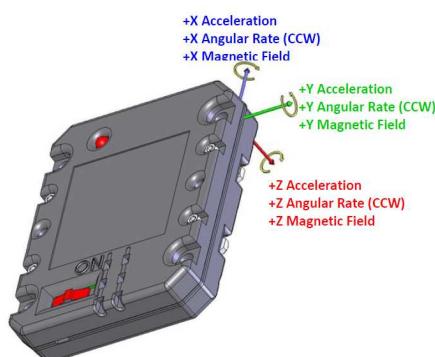


Figure 2.17: MemSense's Bluetooth IMU coordinate system.

2.3 Focus of research

It is still possible to continue researching to develop new prototypes of IMUs. Forthcoming IMUs should have higher local processing capability by adding small processors or small DSPs (such as the dsPIC hybrid), in order to perform locally the calibration, preprocessing and processing algorithms. They should also be lighter and smaller to facilitate their attachment to the body of the subject under test. A very important field of research is to decrease power consumption. Recent studies [20], [35] have been carried out in order to find systems to harvest energy from human body motion. One way to scavenge energy from human activity is to use vibrating beam piezoelectric harvesters. These generators [3] contain piezoelectric layers that are attached to a mass at the tip. When the system moves, the mass starts to oscillate and the layers start to expand and compress which derives in the generation of power.

Nowadays, it is not possible yet to completely remove batteries as the piezoelectric harvesters do not produce enough power for the complete system. Some published works [45] show that future harvesters attached to the body might be enough to extend battery life, or even completely power the IMU or other sensor motes in a wireless personal area sensor network.

Research could be also focused on developing low consumption transmission algorithms for the sensor network. Such algorithms may decrease the power consumption of the transducers, which represent the highest percentage of system power consumption.

CHAPTER 3

Datalogging Software

The next step in the project is gathering the data coming from the IMU and storing it for further processing. Data is received using an XBee receptor when using Wagyro or Wagyromag, and using a bluetooth adaptor when using Memsense's Bluetooth IMU. Data can also be gathered and stored through and microSD card if we are using Wagyromag in off-line monitoring sessions.

Two different software tools have been developed to display data in real time and store them in text files. The first one, WagyLog, has been implemented using National Instrument's Labview 2009 and can be used either for Wagyro, Bluetooth IMU or Wagyromag. The second tool, VisUMU, has been implemented using Java and can be used to read data from the Bluetooth IMU.

Both applications have the final objective of generating a file with the read samples, saving them in columns (one for each axis of the sensors), after the time stamp column, which is the first one.

Once the data files are stored, we can proceed with the rest of the steps to achieve an accurate human body position monitoring, that is, the calibration and signal processing procedures. Such procedures will be developed in the following chapters. Right now we will focus in gathering and storing the data which is the main objective of this chapter.

Let's study more in depth the developed applications.

3.1 VisIMU

VisIMU is a software application developed in collaboration with the University of Applied Sciences of Munster, Germany. It is conceived as a general framework to read and store data received from different IMUs. By specifying the format frame and the number and kind of sensors, new devices can be added to the list of compatibility. The application is part of an ongoing project that aims to develop a general tool with the following features:

- Read & store incoming sensor data from all kind of IMUs.
- Automatic calibration of the received data.
- Graphical visualization of the received data through 2D plots.
- IMU position computation (carried out using different algorithms).
- Graphical visualization of the computed position on a 3D scenario.
- Interaction with a web platform designed to store and visualize data remotely.

Figure 3.1 shows the user interface of the application while reading data being sent by the sensors.

Go to C to read the user manual and further information.

3.2 WagyLog

WagyLog is an application to read and store data from IMUs. It can read data from Wagyro, the Bluetooth IMU and Wagyromag. It is developed using Labview 2009, so in order to run the application we need Labview 2009 installed in our system.

LabVIEW is a graphical programming tool conceived to design process control routines in a sensor network or industrial scenario. It permits fast prototyping and testing of control applications. It uses G language to build the Virtual Instruments (VI), i.e. the applications.

The developed VIs are divided in two parts, a frontal panel that shows the graphical user interface and block diagram that contains the graphical programming including flux diagrams and different functional blocks. In order to build a complete VI we need to build both parts.

LabVIEW is an useful tool to rapidly integrate developed algorithms into a working application as it includes many built-in module libraries including hundreds of mathematical, data manipulation and visualization and signal processing functions in form of blocks. When using WagyLog the data will be also stored in a text file having the same format than the file generated by VisIMU. Similarly, we will process the information in MATLAB after the data files have been stored.

Figure 3.2 shows the frontal panel of WagyLog. There is a plotter for each axis of the sensors and a part to configure the connection parameters.

3.3 Focus of research

Future research on data gathering applications will be focused on improving and

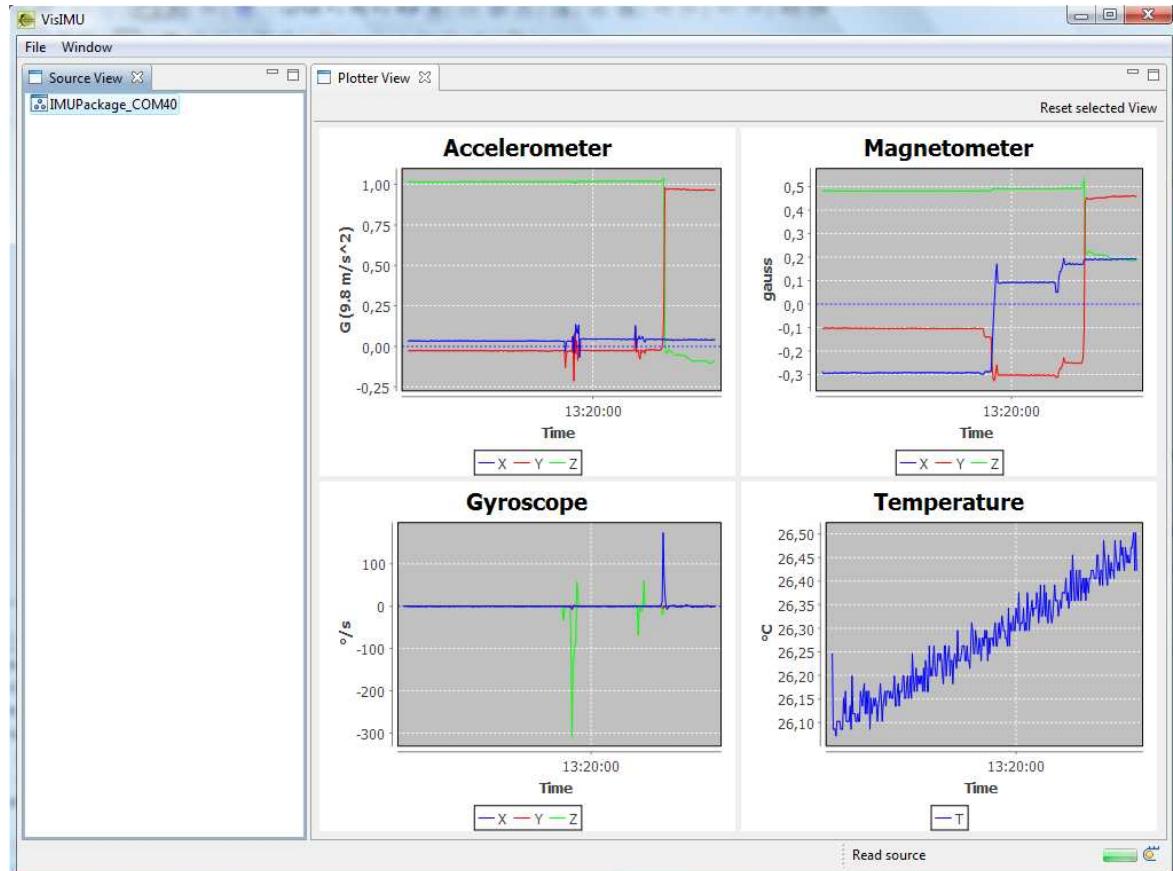


Figure 3.1: VisIMU's user interface



Figure 3.2: WagyLog's Frontal Panel

increasing the functionalities of VisIMU. A wizzard to generate new readers for any kind of IMUs will be implemented. The user will only need to specify the number of sensors, the number of axes of each sensor and the format of the frame sent by the IMU in order to be able to extract the data.

Different calibration algorithms (pre-use and in-use) will also be included in the application in order to see their effect in real time. Moreover, all the already developed position computation algorithms will be added so a 3D avatar can show the movement and position of the body in real time.

The general objective is, in summary, building an application that performs all the steps that are needed to implement a complete human body position monitoring system so there is no need to use further data analysis software as MATLAB.

CHAPTER 4

Calibration Procedures

When working with sensors, calibration is one of the most important, delicate and meticulous processes that needs to be carried out. Prior to the calibration process, the information at the sensors' output will be a signal composed of integer numbers, or real numbers (if the information is not processed by an Analog/Digital Converter) bounded into a range which is determined by the precision of the sensors and converters. These numeric values lack of physical value, so it is strictly necessary to convert them into a scale that can be measured in physical units. Moreover, the calibration will help to reduce the undesired components and errors of the output caused by different kinds of effects that will be explained in this section.

4.1 Introduction, actual trends and classification

Depending on the sensor being calibrated, there is a wide range of calibration procedures that are based on different principles. First of all, we can classify them attending to the moment where the procedure is carried out. Most algorithms are applied before the use of the IMU, but, occasionally, we can apply automatic algorithms that recalibrate the sensor while it is being used. Such automatic systems are recommended for lower quality sensors that present a time varying behavior and are also very sensitive

to temperature changes.

On the other hand, we can classify the calibration procedures attending to the employed instrumentation. For military, industrial and aviation applications the sensors need to be very accurate and so has the calibration equipment. Accurate calibration equipment, usually composed of computer controlled multi axis rate tables including a thermal chamber (see figure 4.1) is really expensive.



Figure 4.1: The Ideal Aerosmith Model 2003HP three-axis rate table with integral LN2 Thermal Chamber.

The companies using military, industrial or aviation sensors can afford buying such instrumentation as their cost represents a very small percentage of the whole budget. However, for human body monitoring applications the employed instrumentation has to be very cheap and simple as the cost of a rate table is at least up to 250 times [44] the cost of Wagyo.

Most calibration works are based on the work developed by Franco Ferraris [21] in 1995. Ferraris proposed a procedure to calibrate triaxial gyroscopes and accelerometers without using any kind of instrumentation. From there, many works have proposed variations of the original algorithm . To calibrate magnetometers most authors use the ellipsoid fitting procedure [37], [26] and [23].

This chapter aims to present different kinds of algorithms for each sensor. Some of

them have already been tested and the rest will be tested in forthcoming research, and are presented here to let the reader know their theoretical basis, and as a presentation of the state of the art.

4.2 Error analysis

The main error sources of the accelerometers, gyroscopes are scale factor errors (caused by their non linearity), offset (static and dynamic), misalignment of the sensor triad with respect to the IMU case and non orthogonality of the sensor triad. Dynamic offset is highly present in gyroscopes due to their nature. Another important error effect in MEMS gyroscopes is Angle Random Walk (ARW), which is later explained.

Magnetometers also have scale factor and offset errors but their sources are different, however, their effects are the same on output of the sensors.

4.2.1 Accelerometers

When using MEMS accelerometers the main error sources are scale factor errors and offsets (also known as biases). The offset is highly static so once it has been calculated it can be easily corrected. Misalignment can be fixed by properly assembling the circuit board inside the IMU case. Orthogonality errors depend on the quality of the sensors. The orthogonality of the sensor axes depends on the manufacture process. The output of the accelerometer is also disrupted by random noise. As accelerometers were the first MEMS sensors to be employed in industry, their manufacture technology is very advanced and new sensors are less distorted by error sources, that is, the scale factors are highly linear, the offsets are low and very static, the sensors are almost completely orthogonal and there is low presence of random noise.

4.2.2 Gyroscopes

When using MEMS gyroscopes, the main error sources are also the scale factor errors and offsets. The offsets are dynamic, i.e. their value increases with time, so they need to be recalculated during use. Misalignment and orthogonality errors are also present. Dynamic offset can lead to complete erroneous measurements, therefore it is the most harmful error source. The ARW can also be very harmful if the quality of the sensors is low. ARW is caused by the integration of the random noise present at the sensor output. Such integration leads to a random angle wander that distorts completely the gyroscope measurements. It will be increased the longer we integrate, providing, therefore, a fundamental limitation to any angle measurement that relies solely on integration of rate. Depending on the quality of the sensor, the dynamic offset will be

predominant over the ARW or viceversa. To study the behavior of the sensor we only need to leave the IMU static and record data during a period of time. This data is later integrated and plotted. If the plotted angle shows a random behavior, then the ARW is predominant. On the other hand, if the angle shows a linear behavior, then the dynamic bias is predominant. Figures 4.2 and 4.3 show the presence of random noise and dynamic bias on Wagyo's gyroscope. Figure 4.4 shows the integrated data gathered while the IMU is static. As we can see, the variation of the angle is almost linear, so we can conclude that the presence of ARW is almost negligible. This is due to the white nature of the noise, which has zero mean. The temperature effects lead to a different slope of the linear behavior of the angle in every experiment execution.

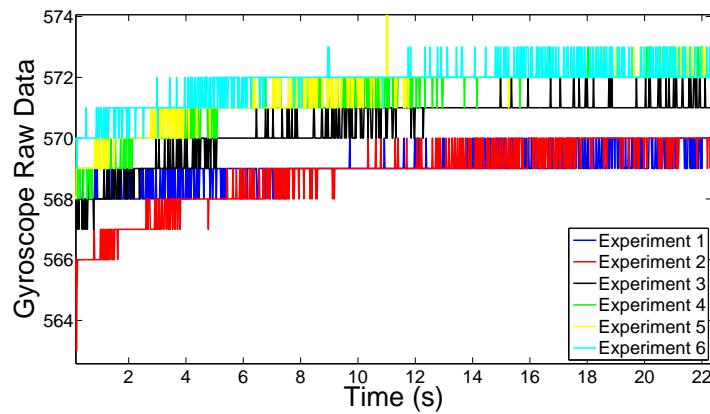


Figure 4.2: Registered raw gyroscope data in static position. Presence of noise and dynamic bias is very evident.

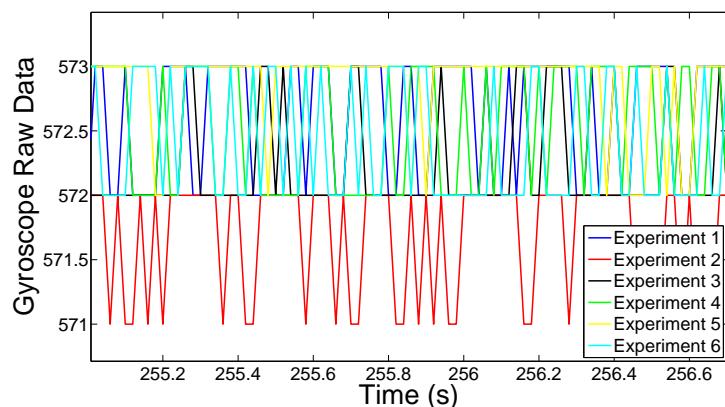


Figure 4.3: Registered raw gyroscope data in static position (zoomed). Presence of noise is very evident.

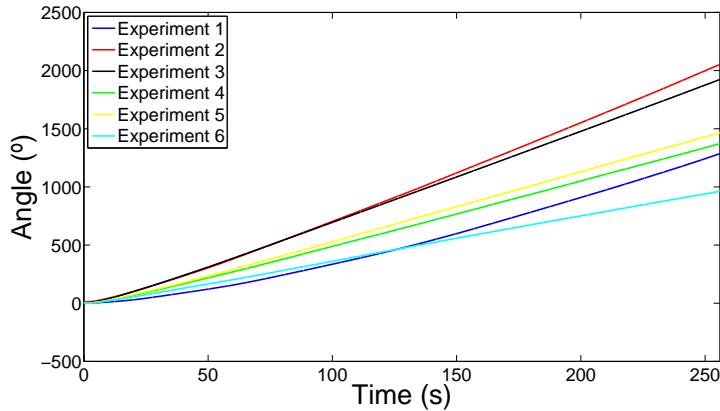


Figure 4.4: Integration of registered raw gyroscope data in static position. Notice the linear behavior. No random wander is observed.

4.2.3 Magnetometers

The magnetometer output is typically corrupted due to five error sources, namely: scale factor errors, misalignment errors, null shift errors, and finally, hard iron and soft iron errors. At the physical level, these error sources can be broken into two categories, those caused by the sensors and those inherent in the measuring of magnetic fields. Scale factor errors and null shift errors are a trait of each individual sensing element, and physical misalignment errors are caused by manufacturing tolerances in the construction of sensor sets. Hard iron and soft iron errors on the other hand are side effects of measuring a magnetic field (caused, respectively, by permanent magnetization of ferrous metals induced usually by stress during manufacture, and variable magnetic permeability which causes induced magnetic fields to appear).

The null shift of a sensor, also known as a sensor's dc offset or zero bias, is a constant offset that shifts the output of each sensor.

The sensitivity of a sensor serves to scale the output, and as no two sensors will have the exact same sensitivity it must be determined for each axis independently.

When the sensors in a set are aligned properly, a field aligned with only one axis is observed by only one sensor. However, if there is a misalignment, multiple sensors will observe a field that is on only one axis.

The hard iron errors are constant, unwanted magnetic fields observed by a magnetometer. A magnetometer will pick up any magnetic field, including those not belonging to the Earth. Their source must be attached to the same body frame as the magnetometer, otherwise the fields would vary depending on the position of the IMU.

Soft irons are materials that emit their own field in response to exposure to an external field. The effect of hard and soft iron materials is explained in the following section in more depth. The magnetometer accuracy is also affected by

1. *Temperature effects:* There are two characteristics of temperature to consider—the offset drift with temperature and the sensitivity tempco. The sensitivity tempco will appear as a change in output gain of the sensor over temperature. (See figure 4.5). Magnetoresistive sensors generally have sensitivity tempcos that are well correlated, or matched. The matching tempcos imply that the output change over temperature of the X axis will track the change in output of the Y axis. This effect will cancel itself since it is the ratio of Y over X that is used in the heading calculation [Azimuth= $\text{arcTan}(h_Y/h_X)$].

2. *Nearby ferrous materials:* The amount of disturbance depends on the material content of the platform and connectors as well as ferrous objects moving near the compass. When a ferrous object is placed in a uniform magnetic field it will create disturbances as shown in figure 4.6. The net result is a characteristic distortion, or anomaly, to the earth's magnetic field that is unique to the shape of the object. With no disturbances, when a two-axis magnetic sensor is rotated in the horizontal plane, the output plot of h_x vs. h_y will form a circle centered at the (0,0) origin (see figure 4.7). The effect of a magnetic disturbance on the heading will be to distort the circle shown in figure 4.7. Hard iron effects add a constant magnitude field component along each axes of the sensor output. This appears as a shift in the origin of the circle equal to the hard iron disturbance in the h_x and h_y axis as depicted in figure 4.8.
The amount of distortion from the soft iron depends on the compass orientation. Soft iron influence on the field values measured by X and Y sensors are shown in figure 4.9.

3. *Compass tilt errors:* The heading calculation will be affected by pitch and roll calculation errors. At the equator, tilt errors are less critical since the earth's field is strictly in the horizontal plane. This provides larger h_x and h_y readings and little Z component correction. On the other hand, near the magnetic poles, tilt errors are extremely important since there is less h_x and h_y field and more Z component.
The magnetic sensor offset drift with temperature is not correlated and may in fact drift in opposite directions.

4. *Variation of the earth's field:* It is well known that the earth's magnetic poles and its axis of rotation are not at the same geographical location. They are about 11.5° rotation from each other. This creates a difference between the true north, or grid north, and the magnetic north, or direction a magnetic compass will point. This error is only important for navigation systems, not for body movement monitoring applications.

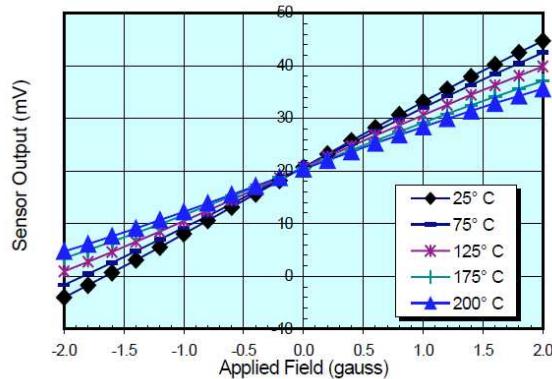
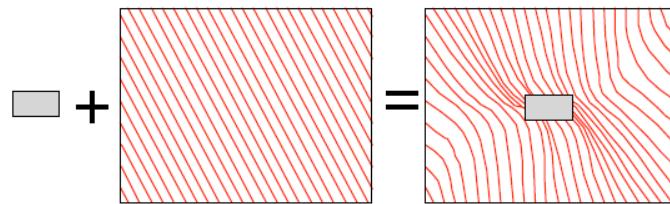


Figure 4.5: Magnetic sensor output temperature variation has a pivot point at zero applied field.



Ferrous Object + Uniform Magnetic Field = Field Disturbance

Figure 4.6: Ferrous object disturbance in uniform field.

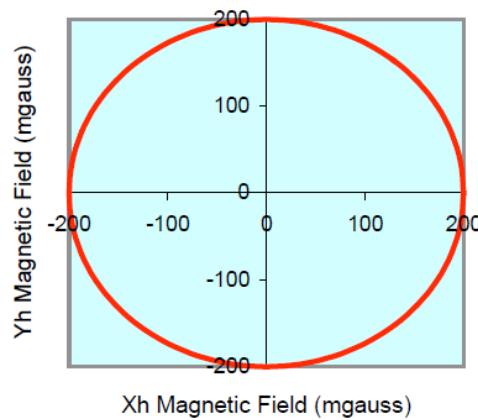


Figure 4.7: Magnetic sensor outputs (h_x, h_y) rotated horizontally in the earth's field with no disturbances.

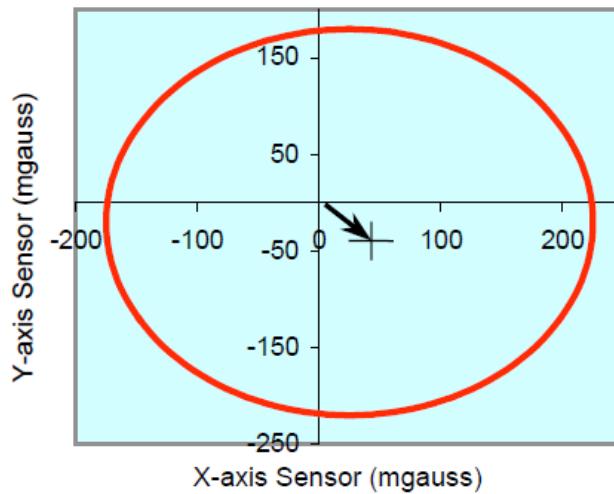


Figure 4.8: Hard iron offsets when rotated horizontally in the earth's field.

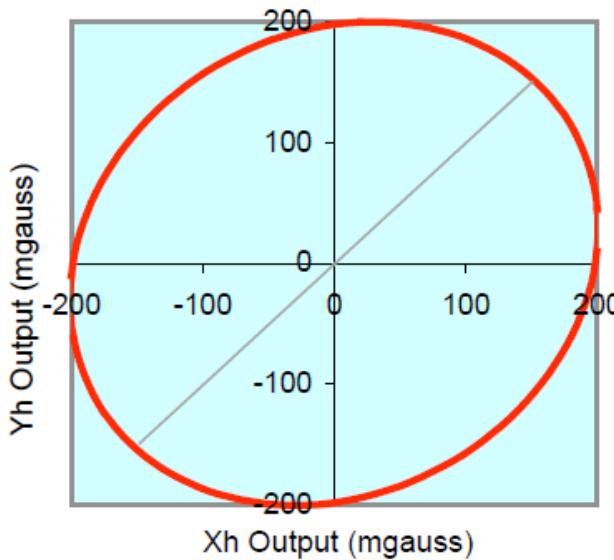


Figure 4.9: Soft iron distortion when rotated horizontally in the earth's field.

4.3 Sensor modeling

In this section we will develop a mathematical model to represent the output of the sensors and include the aforementioned errors. The model is general and can be used for the three sensors by simplifying it or completing it depending on the employed sensor. The general triaxial sensor model is written in vectorial form, where index k

represents the type of the sensor (g, a or m; gyro, accelerometer or magnetometer, respectively). Measured quantities, outputs and biases of a sensor triad are incorporated in the vectors: \vec{u}_k , \vec{y}_k and \vec{b}_k , respectively; sensitivities are arranged in the matrix S_k . Another matrix is introduced: the orthogonalization matrix T_k . Other effects like compass's cross-axis sensitivity, the gyro acceleration sensitivity and the nonlinearities of the sensors are neglected. Then, the sensor model can be represented as:

$$\vec{y}_k = S_k T_k \vec{u}_k + \vec{b}_k \quad (4.1)$$

$$\vec{u}_k = \begin{pmatrix} u_{kx} \\ u_{ky} \\ u_{kz} \end{pmatrix}, \quad \vec{y}_k = \begin{pmatrix} y_{kx} \\ y_{ky} \\ y_{kz} \end{pmatrix}, \quad \vec{b}_k = \begin{pmatrix} b_{kx} \\ b_{ky} \\ b_{kz} \end{pmatrix} \quad (4.2)$$

$$S_k = \begin{pmatrix} s_{kx} & 0 & 0 \\ 0 & s_{ky} & 0 \\ 0 & 0 & s_{kz} \end{pmatrix} \quad (4.3)$$

$$T_k = \begin{pmatrix} 1 & 0 & 0 \\ \cos \alpha_k & 1 & 0 \\ \cos \beta_k & \cos \gamma_k & 1 \end{pmatrix} \quad (4.4)$$

Orthogonalization matrix T_k transforms the vector expressed in the orthogonal sensor reference frame k_o into the vector expressed in the non-orthogonal sensor reference frame k (figure 4.10).

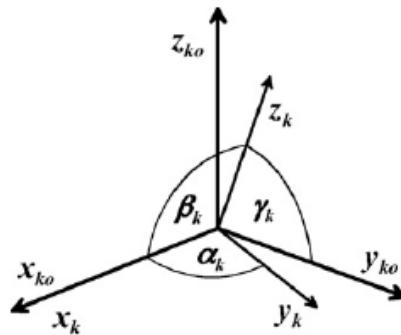


Figure 4.10: Orthogonalization of the sensor frame k .

Therefore, 9 parameters must be determined during the calibration and alignment process for each sensor triad. These parameters are divided into two groups: the group of mechanical parameters and the group of electrical parameters. Mechanical parameters (orthogonalization parameters) are independent of temperature and time during normal operation (which assumes no excessive shocks and stresses) and need to be determined only once during the final production phase. Electrical parameters, on

the other hand, must be re-established more frequently or even at every start up (e.g. gyro bias). Finally, the estimate $\hat{\vec{u}}_k$ for the observed physical quantity \vec{u}_k is:

$$\hat{\vec{u}}_k = T_k^{-1} S_k^{-1} (\vec{y}_k - \vec{b}_k) \quad (4.5)$$

4.4 Calibration before use

Before using the IMU we need to calculate the parameters from the mathematical model. These parameters are computed differently depending on the kind of sensor.

4.4.1 Accelerometers

The parameters \vec{p} are established by the minimization of an objective function $O(\vec{p})$. The objective function is defined as the mean square error between the reference value u_{ref} and the corresponding data vector $u_n(\vec{p})$:

$$O(\vec{p}) = \frac{1}{N} \sum_{n=1}^N (u_{ref} - u_n(\vec{p}))^2 \quad (4.6)$$

where N is the number of measured values in the data vector. The gravity norm estimate ($u_n(\vec{p}) = \|\hat{\vec{a}}_n\|$) by the accelerometer triad and the Earth's magnetic field norm estimate ($u_n(\vec{p}) = \|\hat{\vec{m}}_n\|$) from the electronic compass are compared to the normalized local values ($u_{ref} = 1$ in both cases).

The parameters are grouped in the calibration parameter vector $\vec{p}_{k,cal}$:

$$\vec{p}_{k,cal} = [s_{kx} \quad s_{ky} \quad s_{kz} \quad b_{kx} \quad b_{ky} \quad b_{kz} \quad \alpha_k \quad \beta_k \quad \gamma_k]^T \quad (4.7)$$

The acquisition of the data needed for the optimization is carried out by placing the IMU into at least nine different orientations where several data points should be acquired at each orientation. The precise knowledge of the orientation is not necessary, however it is important that the IMU is in standstill during the data acquisition to minimize the noise in the sensor outputs. The parameter estimation is performed using the constrained Newton optimization method. The initial values of the parameters and the constraints are set according to the typical values quoted in the sensor's datasheets. For applications where we only want to compute the position of the IMU relative to the body frame, or other IMUs, we can simplify the method to calibrate the accelerometers and use the six-positions method. In such method, we will only calculate the scale factors and the offsets, being the sensor model:

$$\vec{y}_k = S_k \vec{u}_k + \vec{b}_k \quad (4.8)$$

Where again, S_k is the vector containing the scale factors and \vec{b}_k is the vector containing the biases (offsets). Therefore, we will have two unknowns for each equation that can be solved obtaining at least two data points for each axis.

Accelerometers are sensitive to gravity acceleration, so when each one of its sensitivity axes is placed parallel to gravity in both positive and negative senses, the measured accelerations will be 1g and -1g respectively. Therefore, two pairs of points are obtained for each axis just by placing the IMU on a fix surface leaning on the two perpendicular sides to the gravity axis, i.e. a 180° rotation is done along the three accelerometer axes as explained in [21]. Figure 4.11 shows a diagram with the position of the IMU for each axis.

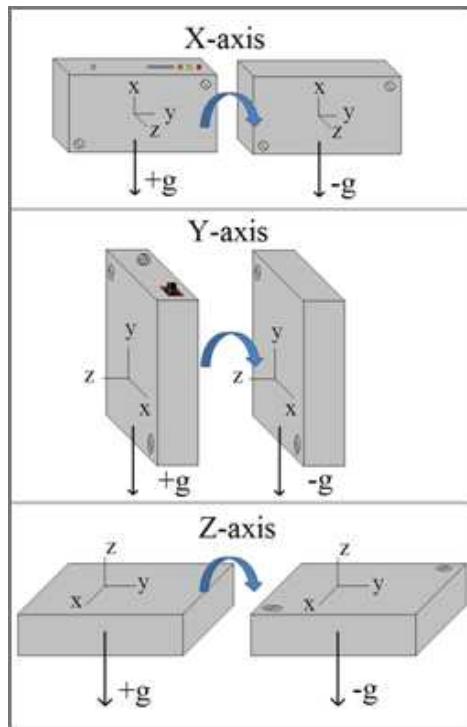


Figure 4.11: IMU positioning to transform raw data into acceleration units (g) for axes X , Y and Z . Figure axes are related to the Earth frame.

4.4.2 Gyroscopes

To obtain the calibration parameters almost all works in literature use the method presented in [21] to calibrate the triaxial gyroscope. In order to fit the complete triaxial sensor model the original method has to be modified so it incorporates the orthogonalization matrix as well. The bias vector \vec{b}_g is estimated as the mean value of the sensor data during the measurement period while the MIMU is kept in standstill. To determine the periods where the IMU is static we can use two different methods:

4.4.2.1 Threshold-based static detection

A Simple Moving Average (SMA) is performed on a 2 seconds window of gyroscope data. If the resulting average value does not exceed a preestablished threshold then the IMU is considered to be still and the SMA is set as the new bias.

$$u_{SMA} = \frac{1}{M-N} \sum_{i=M}^{M-N} u_{g\alpha}(i) \quad (4.9)$$

where M is the starting point of the moving average, N is the block size of the moving average and $u_{g\alpha}(i)$ is the gyroscope output signal. A study of the variation of the dynamic bias with time was carried out in order to determine the stabilized bias value, which was set as the preestablished threshold.

4.4.2.2 Detection based on the acceleration magnitude

When the IMU is static only the acceleration of the gravity will be sensed. Hence, the magnitude of the acceleration will be close to the unity. If the magnitude remains close to this value in a period of 2 seconds the IMU is considered to be static. Then, the buffered gyroscope data during the last 2 seconds is averaged and the new bias value is set.

$$\|\vec{u}_a\| = \sqrt{\vec{u}_{ax}^2 + \vec{u}_{ay}^2 + \vec{u}_{az}^2} = 1 \pm \delta \quad (4.10)$$

where $\delta=0.01$ is the maximum amplitude of the recorded noise at the accelerometer output.

The other six parameters are determined by performing the remaining three measurements. Each measurement is accomplished during the rotation about the individual sensitivity axis with known constant angular rate (actually known rotations after integrating). This way there is no need for complex instrumentation to apply a known constant angular rate. This is later explained). The applied angular rates are arranged on the diagonal of the matrix W_g and the bias corrected angular rate estimates ($\vec{y}_g - \vec{b}_g = S_g T_g M_g \vec{u}_g$) from the gyro triad are arranged in the matrix v_g , where the element $r_{g,ij}$ represents the i th gyro's output when the rotation about the j th axis is accomplished.

$$W_g = \begin{pmatrix} \omega_x & 0 & 0 \\ 0 & \omega_y & 0 \\ 0 & 0 & \omega_z \end{pmatrix} \quad (4.11)$$

$$v_g = \begin{pmatrix} r_{g,xx} & r_{g,xy} & r_{g,xz} \\ r_{g,yx} & r_{g,yy} & r_{g,yz} \\ r_{g,zx} & r_{g,zy} & r_{g,zz} \end{pmatrix} \quad (4.12)$$

The subsequent procedure can be resumed on the integrated quantities, meaning that the angular rates are substituted by the angles and no knowledge about the angular velocity is needed. However the angle of rotation needs to be known. With this assumption the angular rate matrix W_g is transformed by integration into the angle matrix A_g and the matrix v_g with bias corrected angular rate estimates is transformed into the angle estimate matrix Y_g .

$$Y_g = S_g T_g M_g A_g \quad (4.13)$$

The calibration procedure is as follows. The MIMU is placed on the flat surface and a full revolution about the surface normal axis is made. Then two successive rotations about the remaining axes are completed. The angles of rotation are written in the matrix A_g and the angle estimates in the matrix Y_g . The steps to calculate matrices S_g , T_g and M_g are as follows:

$$Y_g A_g^{-1} = S_g T_g M_g \quad (4.14)$$

$$(Y_g A_g^{-1})(Y_g A_g^{-1})^T = (S_g T_g M_g)(S_g T_g M_g)^T \quad (4.15)$$

then, the alignment matrix M_g is abridged, because of its orthonormality:

$$(Y_g A_g^{-1})(Y_g A_g^{-1})^T = (S_g T_g)(S_g T_g)^T \quad (4.16)$$

$$S_g T_g = \text{chol}[(Y_g A_g^{-1})(Y_g A_g^{-1})^T]^T \quad (4.17)$$

$$[T_g, S_g] = LU(S_g T_g) \quad (4.18)$$

$$M_g = T_g^{-1} S_g^{-1} Y_g A_g^1 \quad (4.19)$$

As well as with accelerometers, we can neglect orthogonalization errors for applications where less accuracy is needed. A novel calibration algorithm that calculates the scale factors and the biases of each of the gyroscopes axis has been developed.

In the gyroscope case, we can not obtain two pairs of points to calculate the calibration line by using a reference like we did for the accelerometer. Many authors [8], [55], [56] use rate tables with angular velocity controlled by a computer or an encoder. The price of this kind of equipment is many times higher than the cost of most of the MEMS IMUs (price of the 1291BR rate table from Ideal Aerosmith used in [8] is around 250 times more expensive than our IMU). We propose a solution employing rather simple and available equipment such as a bicycle wheel, which will be used as a rotating device.

By placing the IMU in the wheel's axis of rotation we can subject it to angular rates, but the problem of counting the wheel turns to pair actual angular rates to the gyro output is still present. Most IMUs include gyroscopes and accelerometers so that in some situations accelerometer data can improve gyroscope data and viceversa. We will use the accelerometer data to determine when a wheel turn is completed. By tilting the spin axis of the wheel and making it spin, a sinusoidal behavior of the acceleration signal is observed. This is caused by the influence of gravity on the

centripetal acceleration, which will make the output of the accelerometer increase when the centripetal acceleration component in the X axis has the same sense than the gravity component along the X axis, and decrease when they have opposed senses. The sinusoidal signal will complete a period when a wheel turn is completed. This behavior is better understood when observing Figure 4.12.

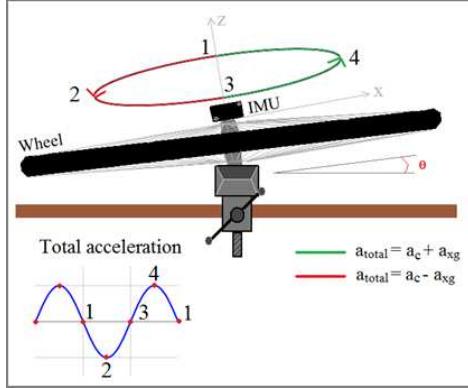


Figure 4.12: Diagram of the experimental setup to count wheel turns using acceleration data.

Figure 4.12 shows that the total measured acceleration \vec{a}_{total} is composed of the centripetal acceleration \vec{a}_c together with the gravity acceleration component along the X axis \vec{a}_{gx} .

$$\vec{a}_c = -\vec{\omega}r^2 \quad (4.20)$$

$$\vec{a}_{gx} = \vec{g}\sin(\theta) \quad (4.21)$$

$$\omega = 2\pi f = \frac{2\pi}{T} \quad (4.22)$$

In equation (4.20) r has a small value because the IMU is placed on the spin axis to gather proper data from the gyroscope; still this fact is not significant since we will be able to observe the sinusoidal signal.

The data acquisition program stores gyroscope and accelerometer data along with a time stamp column which will let us detect the wheel turns in the acceleration signal and, therefore, determine the period of each turn. A vector of periods is built and by applying (4.20) a vector with the angular velocity at instant 2 (Figure 4.12) of each wheel turn is found. This vector is then paired with the gyroscope raw data, so pairs of raw data-angular speed calculated with acceleration data are obtained and used to

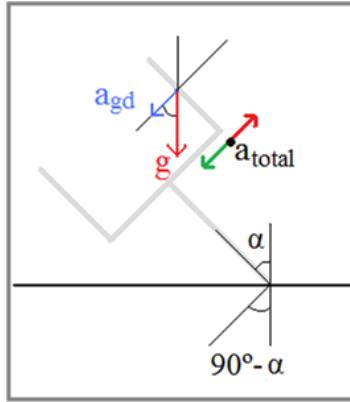


Figure 4.13: Acceleration effects on a tuning fork gyroscope.

build the calibration line. The latter is done by calculating the trend line using least squares estimation.

The friction will reduce the angular velocity of the wheel in a progressive way until it stops; this way the whole sensing range of the gyroscope is comprised, from the highest rate to static state. The wheel has to be spun clockwise and anti-clockwise in order to obtain values in the positive and negative range. The tilt angle of the spin axis has to be chosen with care because the gyroscope is sensitive to gravity acceleration due to its structure and working principle. Micromachined Silicon Vibration Gyroscopes are based on the Coriolis force to sense angular velocity.

Figure 4.13 shows the behavior of the tuning fork when a tilt is applied to its rotating axis. The acceleration resulting from the force applied to the fork to generate the vibration is distorted by the influx of the component of the gravity in the displacement axis. The total acceleration will be greater when the vibration acceleration has the same sense than the gravity component and, on the other hand, it will be smaller when they have opposite senses. The same behavior is present in all vibration based gyroscopes.

$$\vec{a}_{total} = \vec{a}_{vib} + \vec{a}_{gd} \quad (4.23)$$

$$\vec{a}_{gd} = \vec{g} \cos(90^\circ - \alpha) \quad (4.24)$$

Hence, the gyroscope output presents a sinusoidal nature which is more pronounced the higher the tilt angle gets, having its peak when $\alpha = 90^\circ$. The smaller the tilt angle, the least gravity acceleration effects are present, however, when the tilt angle comes close to zero, the sinusoidal acceleration signal gets very noisy to a point where it is impossible to differentiate the periods. A compromise between the two situations is

needed. After several tests a tilt angle of $\alpha = 7^\circ$ showed to be the best option, as the periods of the acceleration signal were observable and the effects of the gravity on the gyroscope output were almost negligible. This procedure reveals the calibration line to be used to transform raw data into angular velocity in $^\circ/s$.

4.4.3 Magnetometers

We can redefine the mathematical model in order to use it for the magnetometer. Like this we identify in a more accurate way the magnetometer specific errors that are not present in the accelerometer and the gyroscope. A comprehensive mathematical model for the output error of a strapdown magnetometer can be written as

$$\mathbf{h}_{me}^w = [C_m C_{sf} C_{si} (\mathbf{h}^b + \mathbf{b}^b + \mathbf{w}^b)] \quad (4.25)$$

Where the subscript *me* means measured, $\mathbf{b}^b = [b_x^b \ b_y^b \ b_z^b]$ and $\mathbf{w}^b = [w_x^b \ w_y^b \ w_z^b]$ represent hard iron biases and wide-band noise, respectively, and are additional errors that corrupt the true measurements \mathbf{h}^b . The variables \mathbf{C}_{si} , \mathbf{C}_{sf} and \mathbf{C}_m are 3×3 matrices that account for soft iron, scale factor, and misalignment errors, respectively.

$$\mathbf{C}_{si} = \begin{pmatrix} \alpha_{xx} & \alpha_{xy} & \alpha_{xz} \\ \alpha_{yx} & \alpha_{yy} & \alpha_{yz} \\ \alpha_{zx} & \alpha_{zy} & \alpha_{zz} \end{pmatrix} \quad (4.26)$$

The α_{ij} terms represents the effective soft iron coefficients and are the constants of proportionality between the magnetic field applied to a soft iron and the resulting induced magnetic field. From a notation point of view, α_{ij} , for example, represents the effective coefficient relating the field generated in the x-direction in response to an applied field in the y-direction.

$$\mathbf{C}_{sf} = \begin{pmatrix} 1 + s_{fx} & 0 & 0 \\ 0 & 1 + s_{fy} & 0 \\ 0 & 0 & 1 + s_{fz} \end{pmatrix} \quad (4.27)$$

The scale factor errors s_{fx} , s_{fy} and s_{fz} represent the uncertainty in knowledge of the constant of proportionality relating magnetometer input to output. In an ideal installation, the magnetometer triad will be mounted imperfect alignment with the body axis of the IMU case. In actual practice, perfect alignment cannot always be achieved. The matrix C_m accounts for this misalignment.

$$\mathbf{C}_m = \begin{pmatrix} 1 & -\varepsilon_z & \varepsilon_y \\ \varepsilon_z & 1 & -\varepsilon_x \\ -\varepsilon_y & \varepsilon_x & 1 \end{pmatrix} \quad (4.28)$$

The three independent parameters defining the matrix (ε_x , ε_y and ε_z) represent small rotations about the body axes of the vehicle that will bring the platform axes into perfect alignment with the body axes. Thus it is constant and only needs to be estimated

once.

After redefining the mathematical model we can apply the calibration algorithm described next.

The fundamental idea behind calibration in the magnetic field domain is that the locus of error-free measurements from a triad of perpendicular magnetometers is a sphere having its center located at the origin. The radius of the sphere is equal to the magnitude of the Earth's magnetic field vector. The effect of hard iron errors is a displacement of the origin from $(0, 0, 0)$ to (b_x, b_y, b_z) . Scale factor and soft iron errors will modify the error-free sphere locus into an ellipsoid. The effect of the wide-band noise is to roughen the smooth surface of the measurement locus. Figure reffig:SphereErrors shows graphically the effects of the errors on the ideal sphere.

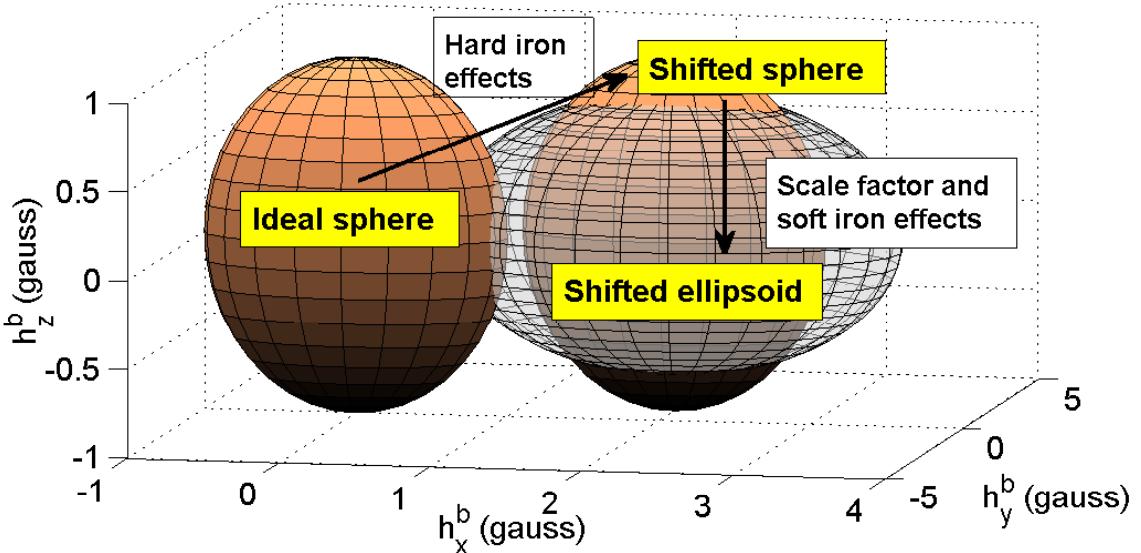


Figure 4.14: Effects of magnetometer errors on the ideal field sphere.

This algorithm is only valid if α_{ij} for $i = j$ is close to zero, as we assume it to be zero. Therefore, the calibration algorithm is the problem of determining the parameters of an ellipsoid that best fit the data collected from a magnetometer triad. Mathematically the ellipsoid can be expressed as follows:

$$\|h\|^2 = h^2 = \left(\frac{h_{me,x}^b - b_x}{\gamma_x} \right)^2 + \left(\frac{h_{me,y}^b - b_y}{\gamma_y} \right)^2 + \left(\frac{h_{me,z}^b - b_z}{\gamma_z} \right)^2 \quad (4.29)$$

where

$$\gamma_x = (1 + s_{fx})(1 + \alpha_{xx}) \quad (4.30)$$

$$\gamma_x = (1 + s_{fy})(1 + \alpha_{yy}) \quad (4.31)$$

$$\gamma_x = (1 + s_{fx})(1 + \alpha_{zz}) \quad (4.32)$$

The parameters to be estimated are the hard iron biases denoted by b_x, b_y and b_z and the combined effect of scale factor errors and the α_{ii} soft iron terms denoted by α_x, α_y and α_z . The known inputs to the calibration algorithm are the measured magnetometer outputs, $\alpha_{me,x}^b, \alpha_{me,y}^b$ and $\alpha_{me,z}^b$, and the magnitude of Earth's magnetic field vector $\|h\|$ in the geographic area where the calibration is being performed. The local value of H_m in Granada is obtained from [5] knowing that the latitude of the lab is $37^\circ 10' 47'' N$, the longitude is $3^\circ 36' 30'' O$ and the elevation with respect to the WGS84 ellipsoid is $0.76 km$. The calculator returns the value of B_x, B_y and B_z in Teslas. Knowing that $1 Gauss = 1.10^{-4} Tesla$ the value of H_m in the lab in Granada, Spain is $0.432 Gauss$. We will use a batch least-squares estimator to fit an ellipsoid to the measured magnetic field data. In order to gather the data we will move the IMU randomly and live it static for a short period of time in each location.

Different approaches can be used to derive the estimator equations for fitting data to an ellipsoid. The approach used in this algorithm is to linearize 4.29. This results in a set of equations that can be easily implemented in a Kalman filter. The states of the estimator are the perturbations of the ellipsoid parameters. Given an initial guess of the unknown parameters, the estimated perturbations are sequentially added to the initial guess and the procedure is repeated until we reach convergence.

To linearize 4.29 we use express the perturbation of $\|h\|$, δh given in [34]

$$-\delta h = \left(\frac{h_{me,x}^b - b_x}{h\gamma_x^2} \right) \delta b_x + \left(\frac{h_{me,x}^b - b_x}{\sqrt{h\gamma_x^3}} \right)^2 \delta \gamma_x + \left(\frac{h_{me,y}^b - b_y}{h\gamma_y^2} \right) \delta b_y + \\ \left(\frac{h_{me,y}^b - b_y}{\sqrt{h\gamma_y^3}} \right)^2 \delta \gamma_y + \left(\frac{h_{me,z}^b - b_z}{h\gamma_z^2} \right) \delta b_z + \left(\frac{h_{me,z}^b - b_z}{\sqrt{h\gamma_z^3}} \right)^2 \delta \gamma_z \quad (4.33)$$

$$\delta h = \xi_x \delta b_x + \eta_x \delta \gamma_x + \xi_y \delta b_y + \eta_y \delta \gamma_y + \xi_z \delta b_z + \eta_z \delta \gamma_z \quad (4.34)$$

where $h = \|h\|$. Thus, given field measurements from k time steps (4.34) can be written as

$$\begin{bmatrix} \delta h_1 \\ \delta h_2 \\ \vdots \\ \delta h_n \end{bmatrix} = \begin{bmatrix} \xi_{x_1} & \eta_{x_1} & \xi_{y_1} & \eta_{y_1} & \xi_{z_1} & \eta_{z_1} \\ \xi_{x_2} & \eta_{x_2} & \xi_{y_2} & \eta_{y_2} & \xi_{z_2} & \eta_{z_2} \\ \xi_{x_3} & \eta_{x_3} & \xi_{y_3} & \eta_{y_3} & \xi_{z_3} & \eta_{z_3} \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ \xi_{x_{k-1}} & \eta_{x_{k-1}} & \xi_{y_{k-1}} & \eta_{y_{k-1}} & \xi_{z_{k-1}} & \eta_{z_{k-1}} \\ \xi_{x_k} & \eta_{x_k} & \xi_{y_k} & \eta_{y_k} & \xi_{z_k} & \eta_{z_k} \end{bmatrix} \begin{bmatrix} \delta b_x \\ \delta \gamma_x \\ \delta b_y \\ \delta \gamma_y \\ \delta b_z \\ \delta \gamma_z \end{bmatrix} \quad (4.35)$$

Equation 4.35 is in the form $\delta\mathbf{h} = H\delta\mathbf{x}$ where $\delta\mathbf{x}$ is the vector of unknowns given by

$$\delta\mathbf{x} = [b_x \quad \gamma_x \quad b_y \quad \gamma_y \quad b_z \quad \gamma_z]^T \quad (4.36)$$

The vector $\delta\mathbf{x}$ is the difference between the known magnetic field vector magnitude and its magnitude as computed from the magnetometer outputs. That is, $\delta h_x = h_k - h_{me,k}$ where h_k is, as said before, 0.432 Gauss and $h_{me,k}$ is computed as

$$h_{me,k} = \sqrt{(h_{me,x}^b)^2 + (h_{me,y}^b)^2 + (h_{me,z}^b)^2} \quad (4.37)$$

An estimate of the calibration parameters $\hat{b}_x, \hat{b}_y, \hat{b}_z, \hat{\gamma}_x, \hat{\gamma}_y$ and $\hat{\gamma}_z$ is obtained by using the following iterative algorithm:

1. Select an initial guess for $\hat{b}_x, \hat{b}_y, \hat{b}_z, \hat{\gamma}_x, \hat{\gamma}_y$ and $\hat{\gamma}_z$. The initial guess for γ_x, γ_y and γ_z must be nonzero.
2. Using the values of $\hat{b}_x, \hat{b}_y, \hat{b}_z, \hat{\gamma}_x, \hat{\gamma}_y$ and $\hat{\gamma}_z$ build equation 4.35.
3. Obtain a least-squares estimate for $\delta\mathbf{x}$, denoted by $\delta\hat{\mathbf{x}}$, as follows

$$\delta\hat{\mathbf{x}} = (H^T H)^{-1} H^T \delta\mathbf{h} \quad (4.38)$$

4. Use the estimate for $\delta\hat{\mathbf{x}}$ and update the unknown parameters as follows:

$$\begin{aligned} b_x(+) &= b_x(-) + \delta\hat{\mathbf{x}}(1) \\ \gamma_x(+) &= \gamma_x(-) + \delta\hat{\mathbf{x}}(2) \\ b_y(+) &= b_y(-) + \delta\hat{\mathbf{x}}(3) \\ \gamma_y(+) &= \gamma_y(-) + \delta\hat{\mathbf{x}}(4) \\ b_z(+) &= b_z(-) + \delta\hat{\mathbf{x}}(5) \\ \gamma_z(+) &= \gamma_z(-) + \delta\hat{\mathbf{x}}(6) \end{aligned} \quad (4.39)$$

5. Compute the covariance matrix \mathbf{P} (which is a measure of the quality of the calibration) by using

$$\mathbf{P} = \sigma_w^2 (H^T H)^{-1} \quad (4.40)$$

where σ_w =standard deviation of the magnetometer wide-band noise.

6. Return to step 2 and repeat until convergence is achieved, which is when the estimate of $b_x, b_y, b_z, \gamma_x, \gamma_y$ and γ_z do not change from one iteration to the next. The estimated calibration parameters can now be used to compute the corrected field measurements h_x^b, h_y^b and h_z^b from the measured magnetometer readings $h_{me,x}^b, h_{me,y}^b$ and $h_{me,z}^b$ using the following relations:

$$h_x^b = \frac{h_{me,x}^b - \hat{b}_x}{\hat{\gamma}_x} \quad (4.41)$$

$$h_y^b = \frac{h_{me,y}^b - \hat{b}_y}{\hat{\gamma}_y} \quad (4.42)$$

$$h_z^b = \frac{h_{me,z}^b - \hat{b}_z}{\hat{\gamma}_z} \quad (4.43)$$

The stability of the least-squares solution is sensitive to the following factors:

- Closeness of initial guess of hard iron biases and γ factors to their actual value.
- Magnitude of wide-band noise on magnetometer outputs.
- Shape and size of magnetometer measurement locus, i.e. the most surface covered by the measurements, the better will be the estimation.

In order to warranty convergence, we can apply an algorithm to determine the value of initial conditions.

The algorithm to establish initial conditions use a nonlinear, two-step estimator that breaks the parameter identification problem into two steps. In the first step, a first-step state vector is formed, the elements of which are algebraic combinations of the elements of the second-step state vector. The elements of the second-step state vector are hard iron biases and those of the scale factor are soft iron γ terms. Following estimation of the first-step states, elements of the second-step state vector are extracted through algebraic manipulation. Derivation of the equations for the nonlinear two-step estimator begin by expanding 4.29.

$$\begin{aligned} h^2 &= \frac{(h_{x,me}^b)^2 - 2(h_{x,me}^b)^2 + (b_x)^2}{\gamma_x^2} + \\ &\frac{(h_{y,me}^b)^2 - 2(h_{y,me}^b)^2 + (b_y)^2}{\gamma_y^2} + \frac{(h_{z,me}^b)^2 - 2(h_{z,me}^b)^2 + (b_z)^2}{\gamma_z^2} \end{aligned} \quad (4.44)$$

Given k measurements, we can construct k separate equations. Equation 4.44 can be expressed in matrix form following the $\mathbf{z} = H\mathbf{x} + \mathbf{v}$ structure as follows

$$-\left[\begin{array}{c} h_{x_1,me}^2 \\ h_{x_2,me}^2 \\ h_{x_3,me}^2 \\ \vdots \\ h_{x_{k-1},me}^2 \\ h_{x_k,me}^2 \end{array} \right] = [H_{11} H_{12}] \left[\begin{array}{c} b_x \\ \mu_1 b_y \\ \mu_2 b_z \\ \mu_1 \\ \mu_2 \\ \mu_3 \end{array} \right] + \mathbf{v} \quad (4.45)$$

The vector \mathbf{v} represents the measurement noise, and the measurement matrix H consists of two $k \times 3$ submatrices, \mathbf{H}_{11} and \mathbf{H}_{12} , which are defined as

$$\mathbf{H}_{11} = \begin{bmatrix} -2h_{x_1,me}^b & -2h_{y_1,me}^b & -2h_{z_1,me}^b \\ -2h_{x_2,me}^b & -2h_{y_2,me}^b & -2h_{z_2,me}^b \\ -2h_{x_3,me}^b & -2h_{y_3,me}^b & -2h_{z_3,me}^b \\ \vdots & \vdots & \vdots \\ -2h_{x_{k-1},me}^b & -2h_{y_{k-1},me}^b & -2h_{z_{k-1},me}^b \\ -2h_{x_k,me}^b & -2h_{y_k,me}^b & -2h_{z_k,me}^b \end{bmatrix} \quad (4.46)$$

$$\mathbf{H}_{12} = \begin{bmatrix} (h_{y_1,me}^b)^2 & (h_{z_1,me}^b)^2 & 1 \\ (h_{y_2,me}^b)^2 & (h_{z_2,me}^b)^2 & 1 \\ (h_{y_3,me}^b)^2 & (h_{z_3,me}^b)^2 & 1 \\ \vdots & \vdots & \vdots \\ (h_{y_{k-1},me}^b)^2 & (h_{z_{k-1},me}^b)^2 & 1 \\ (h_{y_k,me}^b)^2 & (h_{z_k,me}^b)^2 & 1 \end{bmatrix} \quad (4.47)$$

$\mathbf{x} = [b_x \quad \mu_1 b_y \quad \mu_2 b_z \quad \mu_1 \quad \mu_2 \quad \mu_3]^T$, where

$$\mu_1 = \frac{\gamma_x^2}{\gamma_y^2} \quad (4.48)$$

$$\mu_2 = \frac{\gamma_x^2}{\gamma_z^2} \quad (4.49)$$

$$\mu_3 = b_x^2 + \mu_1 b_y^2 + \mu_2 b_z^2 - \mu_4 \mu_4 = h^2 \gamma_x^2 \quad (4.50)$$

represents the first-step state vector.

An estimate for \mathbf{x} , denoted as $\hat{\mathbf{x}}$ is obtained by

$$\hat{\mathbf{x}} = (H^T H)^{-1} H^T \mathbf{y} \quad (4.51)$$

where $\mathbf{y} = [h_{x_1,me}^2 \quad h_{x_2,me}^2 \quad h_{x_3,me}^2 \quad \dots \quad h_{x_{k-1},me}^2 \quad h_{x_k,me}^2]^T$. Finally, we can obtain the initialization parameters by extracting them from $\hat{\mathbf{x}}$

$$\hat{b}_x = \hat{\mathbf{x}}(1) \quad (4.52)$$

$$\hat{b}_y = \frac{\hat{\mathbf{x}}(2)}{\hat{\mathbf{x}}(4)} \quad (4.53)$$

$$\hat{b}_z = \frac{\hat{\mathbf{x}}(3)}{\hat{\mathbf{x}}(5)} \quad (4.54)$$

$$\hat{\gamma}_x = \sqrt{\frac{\mu_4}{h^2}} \quad (4.55)$$

$$\hat{\gamma}_y = \sqrt{\frac{\mu_4}{\mu_1 h^2}} \quad (4.56)$$

$$\hat{\gamma}_z = \sqrt{\frac{\mu_4}{\mu_2 h^2}} \quad (4.57)$$

4.5 Results

4.5.1 Accelerometer

As human body monitoring applications do not need extremely high levels of accuracy when calculating relative joint angles we have only carried out the 6 positions calibration algorithm.

Table 4.1 shows the raw measured values, inside the range [0-1023], for positive and negative orientation for each sensitivity axis along the gravity axis, as well as the values of orthogonal axes. Output expressions for each axis are shown in (4.58), (4.59) and (4.60).

Acceleration components in Table 4.1 are named $u_{\alpha\alpha\beta}$, where α is the non sensitive axis and β is the sensitivity axis when the IMU is oriented along the gravity axis.

Acceleration Component	Positive axis orientation	Negative axis orientation
u_{axx}	618	415
u_{axy}	516	518
u_{axz}	520	517
u_{ayx}	518	515
u_{ayy}	620	414
u_{ayz}	516	519
u_{azz}	513	518
u_{azy}	520	515
u_{azz}	617	420

Table 4.1: Measured raw accelerometer data

With these values we can define 6 equations and obtain the following scale factors and offsets

$$u_{ax} = 9.852 \cdot 10^{-3} n_{ax} - 5.088 \quad (4.58)$$

$$u_{ay} = 9.708 \cdot 10^{-3} n_{ay} - 5.019 \quad (4.59)$$

$$u_{az} = 0.010 n_{az} - 5.263 \quad (4.60)$$

where n_{ax} , n_{ay} and n_{az} are the outputs of the x , y and z accelerometer axes respectively.

4.5.2 Gyroscope

By spinning the wheel with a tilt angle of $\alpha = 7^\circ$ clockwise and anti-clockwise for both IMU's gyroscope sensitivity axis the calibration lines are obtained. Figures 4.15, 4.16, 4.17 and 4.18 show the process. Figure 4.15 shows the acceleration signal sensed in the Z axis when the wheel is spinning clockwise. The progressive decrease of the frequency reveals the decrease of the wheel's angular velocity.

Time difference between consecutive peaks is computed to obtain a vector including the period of each wheel turn. We can observe at a glance in Figure 4.16, how the periods augment due to the effect of the friction. By applying (4.22) on the periods vector, a vector with the angular velocity of each turn is calculated (Figure 4.17). Finally the trend line of the calculated angular velocity is paired with the stored gyroscope data and put together with the trend line obtained following the same procedure for the data recorded during the negative spin. Figure 4.18 shows the calculated calibration line for the X sensitivity axis. Experimental values show a very clear linear behavior so they are accurately fit by the trend line estimation.

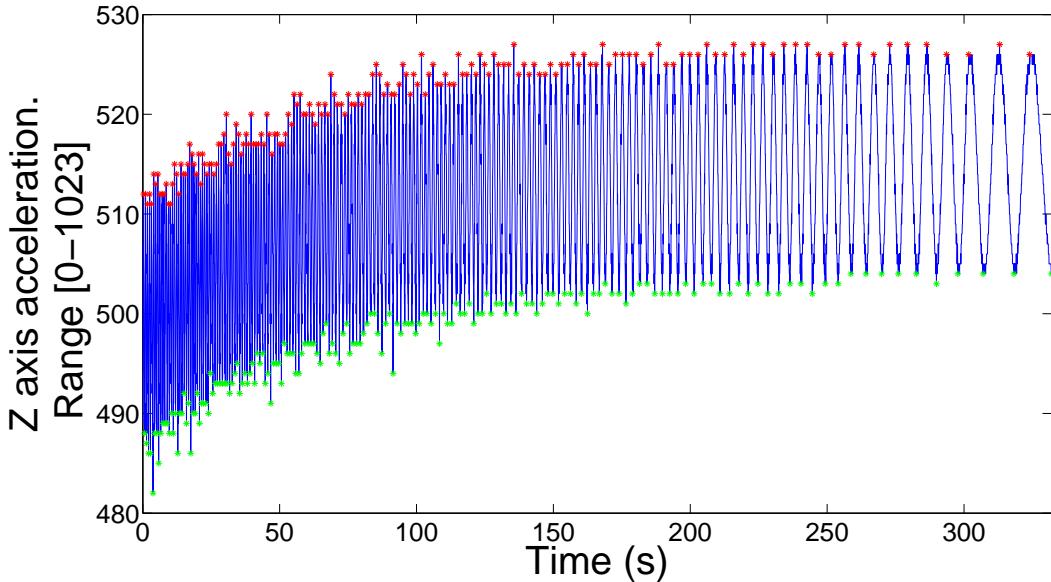


Figure 4.15: Peak detection of the accelerometer signal recorded during the positive spin. Turn periods increase with time due to friction.

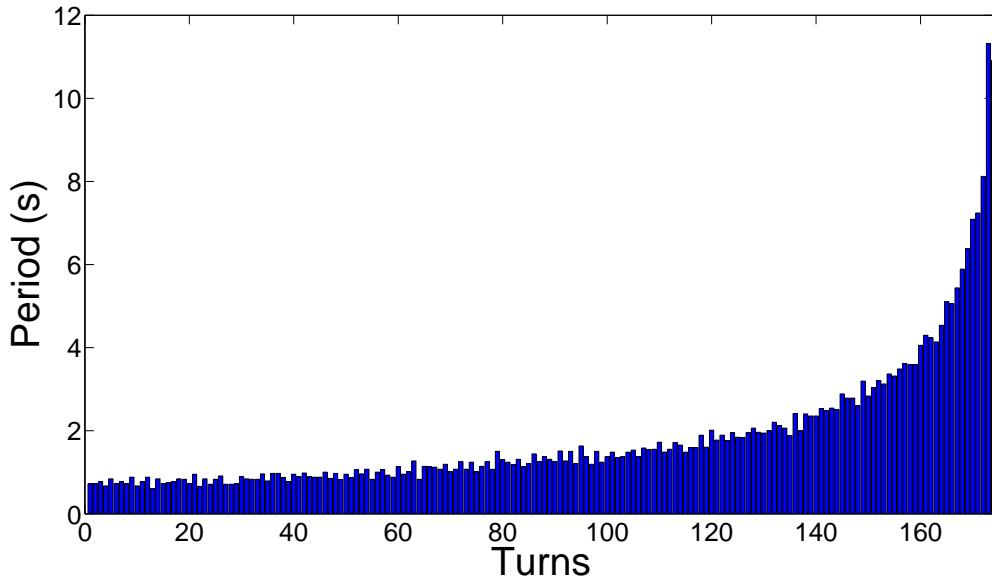


Figure 4.16: Wheel's turn periods obtained from the positive spin using the accelerometer data.

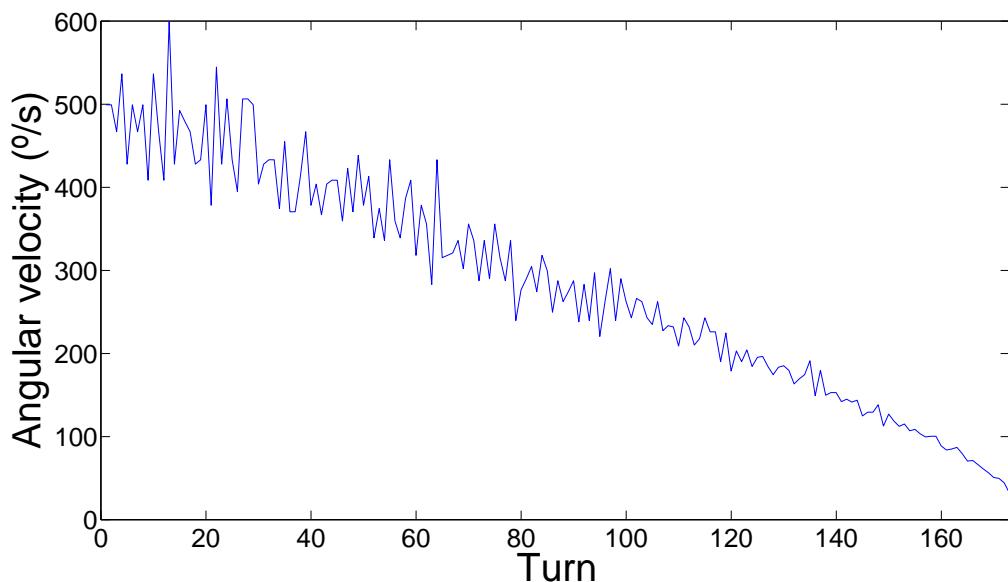


Figure 4.17: Angular velocity obtained with accelerometer data (positive spin).

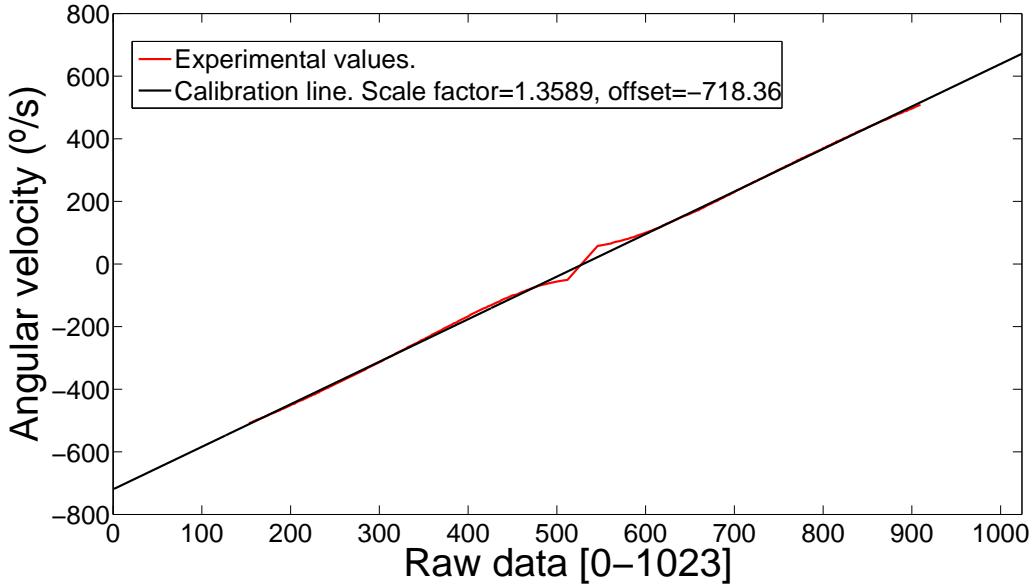


Figure 4.18: Calibration line calculated for the X sensitivity axis.

Expressions (4.61) and (4.62) show the calculated calibration lines.

$$u_{gx} = 1.3589n_{gx} - 718.34 \quad (4.61)$$

$$u_{gy} = 1.3658n_{gy} - 788.06 \quad (4.62)$$

Where u_{gx} and u_{gy} are the angular velocities in $^{\circ}/s$ for the X and Y axes respectively and n_{gx} and n_{gy} are the gyroscope raw data outputs for X and Y axes respectively.

The dynamic bias compensation is carried out by one of the two procedures previously explained. Figure 4.19 shows a study of the dynamic bias variation letting the IMU in a static position for 20 minutes. The dynamic bias observation reveals a variation of $10.92 ^{\circ}/s$ for the Y axis and a variation of $2.71 ^{\circ}/s$ for the X axis in 20 minutes. Such a variation can lead to catastrophic results when, for example, the data are integrated to obtain the angle. It is clearly demonstrated that a compensation algorithm is needed.

Figures 4.20 and 4.21 show the results of both moving average and acceleration magnitude methods. The two methods were able to recalculate the bias during the static periods of the IMU and, thus, dynamic bias effects are compensated during the whole operation time.

Once the calibration process is finished, the IMU is ready to be used, and its measured data can be used in a wide range of applications. The next chapter explains how to measure the position by the computation of the inclination angle of the IMU.

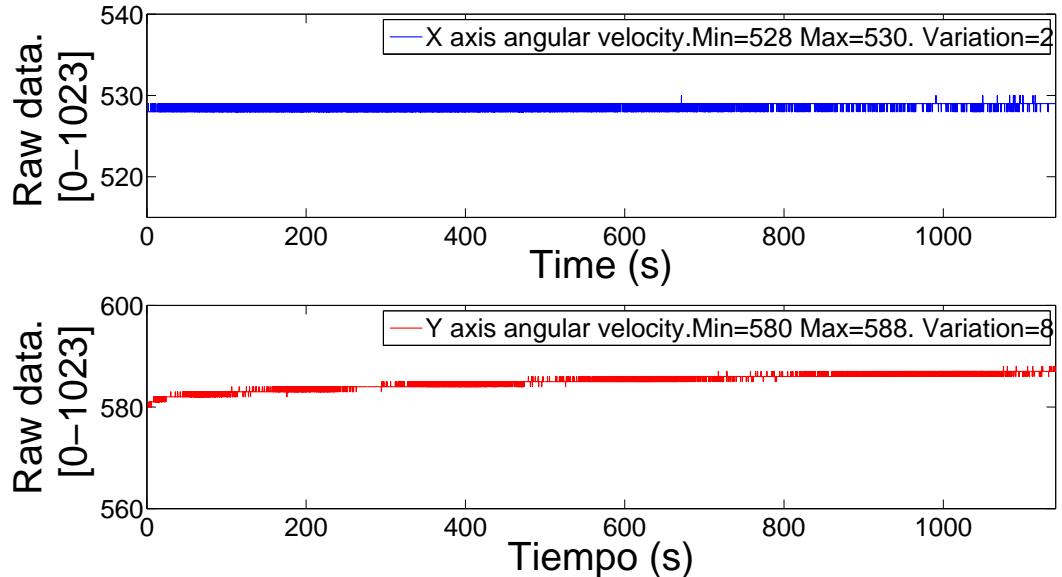


Figure 4.19: Study of dynamic bias variation during static position.

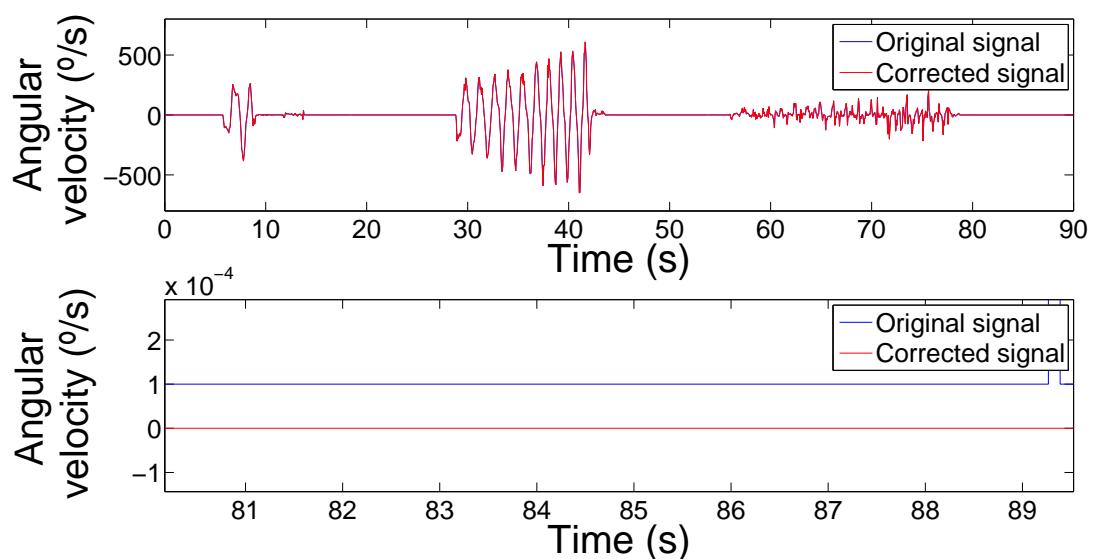


Figure 4.20: Dynamic bias correction using the moving average method.

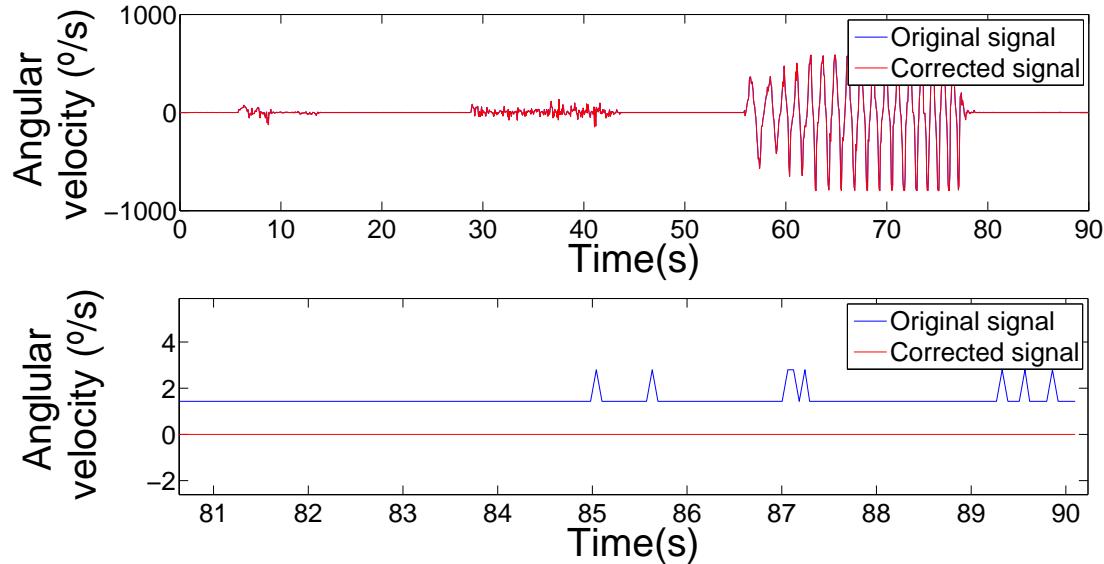


Figure 4.21: Dynamic bias correction using the acceleration magnitude method.

4.6 Focus of research

Research focused on calibration procedures is still a hot topic since we can still find more efficient, simple and accurate procedures. When using algorithms based on ellipsoid fitting we can continue trying different ellipsoid fitting tools to achieve better accuracy. We also need to find a general and complete mathematical model that is valid for all the sensors. Future research will be focused on testing different existing procedures and carrying out a comparative study to determine which one has a better tradeoff between complexity and accuracy.

CHAPTER 5

Position Monitoring

After calibrating the sensors, the measurements will be much more accurate and so will be the computation of the position of the body part where the IMU is attached. In order to calculate the tilt angle of the IMU, we will be using the information given by the acceleration, the angular velocity and the Earth's magnetic field, improving the accuracy by applying a sensor fusion approach based on adaptive filtering. Throughout this section, we will study the different methods to obtain the inclination of the IMU, and consequently, the inclination of the body part where the IMU is placed.

A novel sensor fusion approach is presented. This approach is based on using different variations of the Least Mean Squares and Recursive Least Squares filters.

5.1 Coordinate system

The Navigation Angle System is usually chosen to determine the spatial position of the IMU. This system was developed to be applied on air navigation systems and is composed of three angles known as pitch, roll and yaw, which are also known as the Tait-Bryan angles in mathematics.

- The pitch angle defines the inclination of the axis going from the tail to the

nose of the plane, i.e. it is a rotation around the axis that goes from wingtip to wingtip. It is usually denoted as θ .

- The roll angle defines the inclination of the axis going from wingtip to wingtip of the plane, i.e. it is a rotation around the axis that goes from the tail to the nose of the plane. It is usually denoted as ϕ .
- The yaw angle defines a rotation around the axis that goes from the top to the bottom of the plane. It is usually denoted as ψ .

Figure 5.1 shows a diagram of the Navigation Angle reference system.

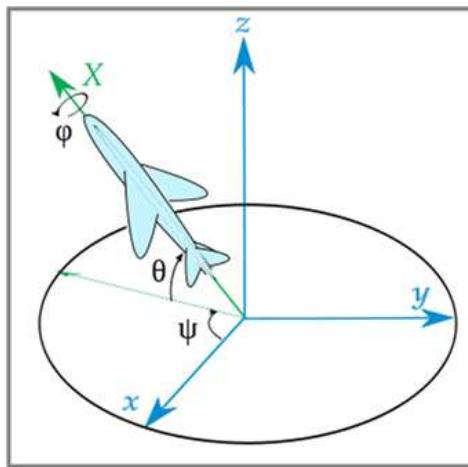


Figure 5.1: Navigation Angle positioning system. θ =pitch, ϕ =roll, ψ =yaw.

There exist four main methods to calculate the navigation angles. The first one is only based in the accelerometer data, the second one is only based in the gyroscope data, the third one uses both accelerometer and magnetometer data and the last one uses a sensor fusion strategy which uses the data from all the sensors and merges it using adaptive filtering. Each of this methods has different pros and cons that are detailed next.

5.2 Angle computation methods

5.2.1 Computation using accelerometers

As it was commented in section 2.1.1, accelerometers are sensitive to gravity acceleration. In a static state, or under a constant velocity, i.e. when there is no acceleration due to changing movements, the measured accelerations at the accelerometers axis are

the decomposition of the gravity acceleration components. Hence, we can obtain the tilt angle in each one of the axis just by applying basic trigonometric relations.

$$\theta = \arctan \left(\frac{u_{ay}}{u_{az}} \right) \quad (5.1)$$

$$\phi = \arctan \left(\frac{u_{ax}}{u_{az}} \right) \quad (5.2)$$

Figures 5.2 and 5.3 depict the decomposition of the gravity acceleration components to calculate the angles by applying equations (5.1) and (5.2).

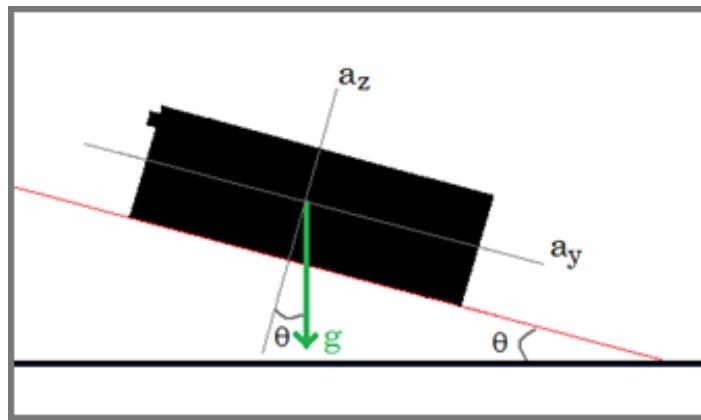


Figure 5.2: Gravity components decomposition to obtain the pitch angle.

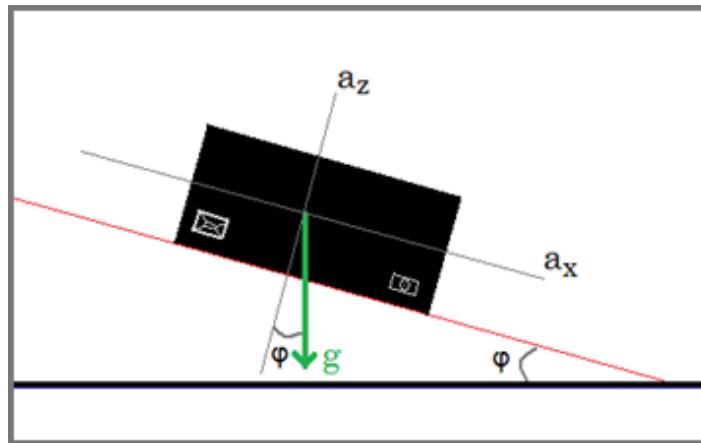


Figure 5.3: Gravity components decomposition to obtain the roll angle.

5.2.1.1 Pros and cons

Since this method relies solely in the gravity static acceleration to determine the tilt angle, when the individual carrying the IMU is performing activity entailing high dy-

namic accelerations, the measured acceleration by the accelerometer will be distorted and, thus, the obtained tilt angle will be totally erroneous. This method only allows the computation of the pitch and roll and does not permit to calculate the yaw angle as it is an angle contained in a plane parallel to the earth's surface.

5.2.2 Computation using gyroscopes

When calculating the tilt angle using the angular velocity, we need to integrate the output signal of the gyroscope. The integration of the angular velocity over a period of time will permit us to obtain the angle displacement during that period of time.

There exist many different numerical integration methods. We will use the numerical integration using the trapezoidal rule which calculates the area under the function performing a trapezoidal approximation as follows:

$$\alpha = \sum_{i=1}^N \left(T u_g(n-1) + \frac{1}{2} T(u_g(n) - u_g(n-1)) \right) \quad (5.3)$$

Where T is the time period length and $u_g(n)$ is the output of the gyro at instant n .

5.2.2.1 Pros and cons

This method offers a smoother angle signal since it does not have the electronic random noise present in the accelerometer output signal. Moreover, the signal is not affected by the dynamic acceleration, so, even under intense movements, the position can be tracked. Unluckily, as explained in section 4.2, the gyroscope output suffers from dynamic bias. This dynamic bias, when integrated, causes an angle wander which increases linearly with time. Figure 5.4 shows the effects of the dynamic bias angle wander on the gyroscope's output signal, when performing rotations of 180° around the sensitivity axis.

The angle wander caused by the dynamic bias will depend on the quality of the employed gyroscope. Recent MEMS gyroscopes present low values of dynamic bias, then, the effects of the integration will be less dramatic. However, the determination of the position by relying solely on the gyroscope output is totally unfeasible.

We need, then, an approach that softens the dynamic acceleration effects of the accelerometer and reduces the effects of the dynamic bias angle wander of the gyroscope. This approach can be achieved by employing a sensor fusion strategy based on adaptive filtering.

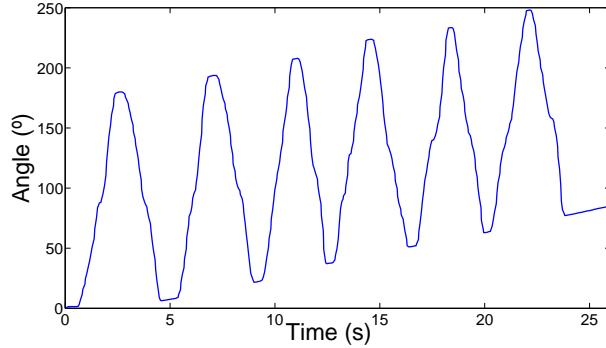


Figure 5.4: Effects of dynamic bias on the integrated gyroscope output signal.

5.2.3 Computation using accelerometers and magnetometers

We can use a magnetometer, also known as a digital compass, to calculate the heading, which is the displacement of a magnetometer sensor to the earth's horizontal magnetic field component. The angle is measured clockwise from 0° to 360° from the true North direction. If the sensor is horizontal to the Earth frame, i.e. roll and pitch angles are null, the heading can be calculated with the two earth's horizontal magnetic field components (X_h and Y_h) using the following equation:

$$\text{heading} = \arctan\left(\frac{Y_h}{X_h}\right) \quad (5.4)$$

Since this situation ($\theta=0$ and $\phi=0$) is very unusual, all three magnetic field components have to be used. First, the measured fields have to be projected to the horizontal plane [11] using the following equations:

$$X_h = h_X * \cos \phi + h_Y * \sin \theta * \sin \phi - h_Z * \cos \theta * \sin \phi \quad (5.5)$$

$$Y_h = h_Y * \cos \theta + h_Z * \sin \theta \quad (5.6)$$

Where h_X , h_Y and h_Z are the magnetic field values measured at the X , Y and Z axes respectively. Once the horizontal components are calculated, equation 5.4 can be applied to determine the heading, but due to the limits of the arctan the following corrections have to be applied depending on the sign of X_h and Y_h component:

Table 5.1: Corrections for Heading calculation with magnetometer

$(X_h < 0)$	$=$	$180^\circ - \arctan\left(\frac{Y_h}{X_h}\right)$
$(X_h > 0, Y_h < 0)$	$=$	$-\arctan\left(\frac{Y_h}{X_h}\right)$
$(X_h > 0, Y_h > 0)$	$=$	$360^\circ - \arctan\left(\frac{Y_h}{X_h}\right)$
$(X_h = 0, Y_h < 0)$	$=$	90°
$(X_h = 0, Y_h > 0)$	$=$	270°

5.2.3.1 Pros and cons

In an ambience with scarce or negligible magnetic perturbations, such as the countryside, this method is really accurate since the magnetometer readings will be very close to the earth's magnetic field. The magnetometer is very robust against acceleration and has no dynamic bias, therefore, it is able to follow the position with high accuracy. However, the countryside is not a very likely scenario where we want to monitor human body motion. Most of the applications of human body monitoring will take place in urban areas, where magnetic disturbances are present. Such disturbances make the magnetometer to measure other magnetic fields apart from the earth's magnetic field, which leads to erroneous positioning calculation. If the sensor is always going to be used in the same scenario, e.g. the same hospital room, we can compensate the disturbances (if they are not time varying) with proper calibration and the magnetometer readings will be accurate enough. But, if we want a robust system that can be used in many places, recalibration of the sensor is really cumbersome and it becomes an impossible task if there is no instrumentation or qualified personnel to do so. Therefore, we need to apply sensor fusion strategies to obtain a robust system under all working circumstances.

5.2.4 Sensor fusion

Since the signal at the output of the accelerometer is not integrated, the random noise will not cause an angle wander as in the gyroscope case. Given that the angle is computed by sensing the gravity acceleration, we will not be able to obtain a correct measurement when there are dynamic accelerations distorting the sensing of the gravity acceleration. This fact prevents us from using only the accelerometer to measure the angle accurately as short bursts of dynamic accelerations are present even in the less intense activities.

We are forced to use a system that is robust under occasionally presence of dynamic accelerations. The gyroscope signal would be ideal if there was no bias drift since it does not have high amplitude noise components and is invulnerable to dynamic accelerations.

If we manage to remove the bias in the gyroscope signal, we will have a good approximation of the position of the subject's limbs. In order to remove the bias we can just rely on the only available reference which is the accelerometer angle signal. Since this signal is occasionally erroneous we need a system that is able to decrease the impact of those erroneous observations by filtering them.

5.2.4.1 Definition

Sensor fusion strategies aim to use the information offered by different sensors to offer an accurate measurement. The fusion is based on the premise that some sensors may apport useful information under conditions where other sensors are not working properly. Therefore, by applying such a fusion, sensors complement each other and offer an acceptable measurement in every moment.

In terms of signal processing, we need a tool that filters the electronic noise present in the accelerometer signal and removes the dynamic bias present in the gyroscope signal. Most authors use an adaptive filtering solution based on the Kalman filtering [33], [64] to clean out the signal from electronic noise and dynamic bias. However, there exist other adaptive filters that might be a good alternative to Kalman filtering as they behave better under high noise and high dynamic bias conditions as we will see in later in this work.

The following subsections present the filtering algorithms that will be tested in the comparative study. A brief theoretical description is included so their main characteristics are highlighted and presented to those readers not familiar with adaptive filtering.

5.2.4.2 Kalman filter

The Kalman filter [33], [64] is a set of mathematical equations that provides a means to estimate the state of a process minimizing the mean of the squared error. The Kalman filter addresses the general problem of trying to estimate the state $x \in \Re^n$ of a discrete-time controlled process that is governed by the linear stochastic difference equation

$$x_k = \Phi_{k-1}x_{k-1} + w_{k-1} \quad (5.7)$$

with a measurement $z \in \Re^m$

$$z_k = H_k z_k + v_k \quad (5.8)$$

where H and Φ are the measurement and state transition matrices respectively which in practice can be considered as invariable. The random variables w_k and v_k represent

the process and measurement noise respectively and are assumed to be independent of each other, white, and with normal probability distributions

$$p(w) \sim N(0, Q) \quad (5.9)$$

$$p(v) \sim N(0, R) \quad (5.10)$$

where Q and R are the process noise covariance and measurement noise covariance matrices respectively and can also be considered to be invariant in practice.

When deriving the equations for the Kalman filter, the goal is to find an equation that computes an a posteriori state estimate \hat{x}_k as a linear combination of an a priori estimate \hat{x}_k^- and a weighted difference between an actual measurement z_k and a measurement prediction $H\hat{x}_k^-$

$$\hat{x}_k = \hat{x}_k^- + K(z_k - H\hat{x}_k^-). \quad (5.11)$$

The difference in (5.11) is called the measurement innovation, or the residual. The residual reflects the discordance between the predicted measurement $H\hat{x}_k^-$ and the actual measurement z_k . The matrix K in (5.11) is chosen to be the gain that will minimize the a posteriori error covariance. Another way of thinking about the weighting by K is that as the measurement error covariance approaches zero, the actual measurement z_k is "trusted" more and more, while the predicted measurement $H\hat{x}_k^-$ is trusted less and less. On the other hand, as the a priori estimate error covariance approaches zero the actual measurement is trusted less and less, while the predicted measurement is trusted more and more.

The Kalman filter estimates a process by using a form of feedback control: the filter estimates the process state at some time and then obtains feedback in the form of noisy measurements. As such, the equations for the Kalman filter fall into two groups: time update equations and measurement update equations. The time update equations can also be thought of as predictor equations, while the measurement update equations can be thought of as corrector equations. If we assume Φ , H and Q to be constant, the time update equations are as follows

$$\hat{x}_k^- = \Phi\hat{x}_{k-1}^- \quad (5.12)$$

$$P_k^- = \Phi P_{k-1}^- \Phi^T + Q \quad (5.13)$$

while the measurement update equations are

$$K_k = P_k^- H^T (H P_k^- H^T + R)^{-1} \quad (5.14)$$

$$\hat{x}_k = \hat{x}_k^- + K_k(z_k - H\hat{x}_k^-) \quad (5.15)$$

$$P_k = (I - K_k H) P_k^- \quad (5.16)$$

where P_k^- and P_k are the a priori and a posteriori estimate error covariance respectively and K is a factor known as the Kalman Gain.

Consequently in order to build a computational algorithm the following steps must be implemented:

- Known parameters.
 1. Φ : State transition matrix.
 2. H : Measurement matrix.
 3. Q : Process noise covariance matrix.
 4. R : Measurement noise covariance matrix.
- Computations.
 1. Compute P_k^- by substituting P_{k-1} , Φ and Q in (5.13).
 2. Compute K_k by substituting P_k^- , H and R in (5.14).
 3. Compute P_k by substituting K_k and P_k^- in (5.16).
 4. Compute successive values of \hat{x}_k (which is the output of the filter) recursively using (5.12), (5.15) and the computed values of K_k . Start with the given initial estimates \hat{x}_0 and P_0 .

Figure 5.5 shows a diagram of the model underlying the Kalman filter. Based on the values provided by the user and the observed process we are able to estimate the hidden values that characterize the process of the dynamic system x_k .

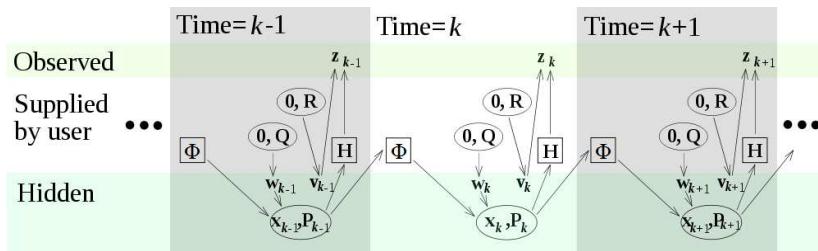


Figure 5.5: Kalman filter algorithm computational diagram.

5.2.4.3 Kalman filter sensor fusion approach

After describing the general working principles of the Kalman filter we must adapt the filter and identify its parameters in order to apply it in a sensor fusion application. First of all we need to identify the process we want to estimate. In this case we want the gyroscope's almost linear angle wander to be estimated so it can be removed.

By removing it we will obtain an accurate unbiased angle signal. Then, we need an observation process that will act as a reference to help determine the hidden process. The best option is to choose the accelerometer angle signal since it can provide reliable angle references as long as it is not distorted by high rates of dynamic acceleration. Therefore, the angle accelerometer signal and the angle gyroscope signal are the two inputs signals of the Kalman filter sensor fusion approach.

Depending on the application scenario, we can tell the filter which signal to trust more by modifying the values of the filter parameters. When the measurement error covariance approaches zero, the acceleration signal is trusted more and more, while the predicted measurement is trusted less and less. On the other hand, as the a priori estimate error covariance approaches zero the acceleration signal is trusted less and less, while the predicted measurement using the gyroscope signal is trusted more and more. This behaviour allows us to tune the filter attending to the activity that is being monitored. When monitoring intense activities the accelerometer signal will be given a lower degree of trust so dynamic accelerations can be filtered. Alternatively, when non-intense movements are monitored, we give the accelerometer angle signal a higher degree of trust. Figure 5.6 depicts the diagram of the Kalman filter sensor fusion approach to remove dynamic bias and filter random noise and dynamic acceleration.

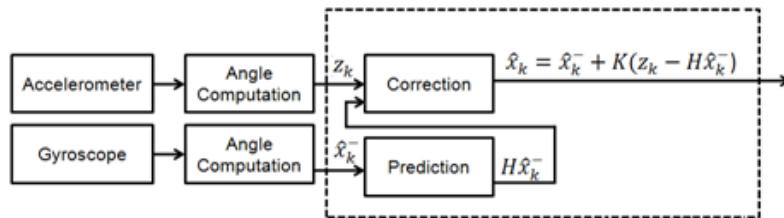


Figure 5.6: Diagram of the Kalman filter sensor fusion approach.

The presented Kalman model is one of the most used sensor fusion approaches in other works. Our objective is to develop a new sensor fusion approach based on LMS and RLS filtering algorithms.

5.2.4.4 Least Mean Squares filters

The Least Mean Squares Filters (LMS) are linear adaptive filtering algorithms, which, in general consists of two basic processes:

1. A filtering process, which involves:
 - Computing the output of a linear filter in response to an input signal.

- Generating an estimation error by comparing this output with a desired response
2. An adaptive process, which involves the automatic adjustment of the parameters of the filter in accordance with the estimation error.

The combination of these two processes working together constitutes a feedback loop. A significative feature of the LMS algorithms family is its simplicity since it does not require measurements of the pertinent correlation functions, nor does it require matrix inversions.

1. Standard LMS.

The Standard LMS algorithm [28] is the simplest and first algorithm to be developed of the LMS algorithms family. During the filtering process, the desired response $\mathbf{d}(n)$ is supplied for processing alongside the tap-input vector $\mathbf{u}(n)$. Given this input, the transversal filter produces an output $\hat{\mathbf{d}}(n)$ used as an estimate of the desired response $\mathbf{d}(n)$. Accordingly, we may define an estimation error $\mathbf{e}(n)$ as the difference between the desired response and the actual filter output.

We assume that the desired response $\mathbf{d}(n)$ is linearly related to the input vector $\mathbf{u}(n)$ by a multiple regression model whose parameter vector is unknown, hence the need for adaptive filtering.

In the adaptive weight-control mechanism, a scalar version of the inner product of the estimation error $\mathbf{e}(n)$ and the tap input $\mathbf{u}(n - k)$ is computed for $k = 0, 1, \dots, M - 1$ where M is the number of coefficients of the filter. The result to the previous product defines the correction $\delta\hat{\mathbf{w}}_k(n)$ applied to the tap weight $\hat{\mathbf{w}}_k(n)$ at the iteration $n + 1$. The scaling factor used in this computation is denoted by a positive quantity μ known as the step-size parameter of the filter.

The feedback loop acting around the tap weights behaves like a low-pass filter, whose average time constant is inversely proportional to the step-size parameter μ . Hence by assigning a small value to μ , the adaptive process is made to progress slowly, and the effects of gradient noise on the tap weights are largely filtered out. The standard LMS algorithm is, then, computed as follows

- (a) Parameters:
 - M : number of taps, that is, the length of the filter.
 - μ : step-size parameter.
- (b) Initialization:
 - Set $\hat{\mathbf{w}}_0 = \mathbf{0}$ unless prior knowledge of the tap-weight vector $\hat{\mathbf{w}}_k(n)$ is available. In that case use it to select an appropriate value for $\hat{\mathbf{w}}_0$.

(c) Given data:

- $\mathbf{u}(n)$: $M - by - 1$ tap-input vector at time n . $\mathbf{u}(n) = [u(n), u(n - 1), \dots, u(n - M + 1)]^T$.
- $\mathbf{d}(n)$: desired response at time n .

(d) Data to be computed:

- $\widehat{\mathbf{w}}_{n+1}$: estimate of tap-weight vector at time $n + 1$.

(e) Computation: For $n = 0, 1, 2, \dots$, compute

$$e(n) = d(n) - \widehat{\mathbf{w}}_n^H \mathbf{u}(n). \quad (5.17)$$

$$\widehat{\mathbf{w}}_{n+1} = \widehat{\mathbf{w}}_n + \mu \mathbf{u}(n) e^*(n). \quad (5.18)$$

2. Normalized LMS (N-LMS).

Normalized LMS [65] normalizes by $\|\mathbf{u}(n)\|^2$ equation (5.18) altering the magnitude, but not the direction, of the estimated gradient vector. This leads to a faster convergence. The update equation is, then,

$$\widehat{\mathbf{w}}_{n+1} = \widehat{\mathbf{w}}_n + \mu \frac{\mathbf{u}^*(n) e(n)}{\|\mathbf{u}(n)\|^2} \quad (5.19)$$

where μ is the step size, $e(n)$ is the error at instant n and $\mathbf{u}(n)$ is the input signal of the filter.

3. Momentum Normalized Least Mean Squares filter (MN-LMS).

The Momentum Normalized Least Mean Squares [14] filter tries to normalize the weighted average square error function and leads to a different weight vector update equation as follows

$$\widehat{\mathbf{w}}_{n+1} = \widehat{\mathbf{w}}_n + \frac{\mu \mathbf{u}^*(n) e(n)}{\mu \|\mathbf{u}(n)\|^2 + 1} \quad (5.20)$$

where μ is the step size, $e(n)$ is the error at instant n and $\mathbf{u}(n)$ is the input signal of the filter.

5.2.4.5 Recursive Least Squares filters

1. Standard RLS.

The Recursive least squares (RLS) adaptive filter [28] is an algorithm which recursively finds the filter coefficients that minimize a weighted linear least squares cost function relating to the input signals. This is in contrast to the LMS filter that aims to reduce the mean square error.

Its rate of convergence is typically an order of magnitude faster than that of

the simple LMS filter due to the fact that the RLS filter whitens the input data by using the inverse correlation matrix of the data, assumed to be of zero mean. However, it is achieved at the expense of an increase in computational complexity, specifically the LMS filter requires $2N + 1$ operations per filter update, whereas the RLS filter requires $2.5N^2 + 4N$, being N the number of filter coefficients. Its standard formulation is computed as follows

$$\hat{\mathbf{w}}_n = \hat{\mathbf{w}}_{n-1} + \alpha(n)\mathbf{g}(n) \quad (5.21)$$

where

$$\mathbf{g}(n) = \frac{\mathbf{z}(n)}{\lambda + \mathbf{u}^T(n)\mathbf{z}(n)} \quad (5.22)$$

$$\alpha(n) = d(n) - \hat{\mathbf{w}}_{n-1}^T\mathbf{u}(n) \quad (5.23)$$

$$\mathbf{z}(n) = \mathbf{P}(n-1)\mathbf{u}^*(n) \quad (5.24)$$

The covariance matrix \mathbf{P}_n is updated by applying

$$\mathbf{P}(n) = \lambda^{-1}[\mathbf{P}(n-1) - \mathbf{g}(n)\mathbf{z}^H(n)] \quad (5.25)$$

where $0 < \lambda \leq 1$ is known as the forgetting factor which gives exponentially less weight to older error samples and is intended to ensure that data in the distant past are forgotten so to afford the possibility of following the statistical variations of the observable data when the filter operates in a nonstationary environment. To build a computational RLS algorithm, the next steps must be followed:

(a) Initialization:

- $\hat{\mathbf{w}}_0 = \mathbf{0}$
- $\mathbf{P}(0) = \delta^{-1}\mathbf{I}$ where δ is called the regularization parameter and is a small or large positive constant for high and low SNR respectively.

(b) Computation: For each instant of time $n = 1, 2, \dots$, compute

- Compute $\mathbf{z}(n)$ using (5.24).
- Compute $\mathbf{g}(n)$ using (5.22).
- Compute $\alpha(n)$ using (5.23).
- Update the filter weight $\hat{\mathbf{w}}_n$ vector using (5.21).
- Update the covariance matrix $\mathbf{P}(n)$ using (5.25).

2. Householder Recursive Least Squares Filter (H-RLS).

The Householder RLS [48] algorithm is a variation of the standard RLS filter that is based on the Householder transformation [29]. Its scheme avoids, almost entirely, costly square roots and divisions (present in other numerically well-behaved adaptive LS schemes) and provides directly the estimates of the unknown

system coefficients. Furthermore, it offers enhanced parallelism, which leads to efficient implementations. The computing process is as follows,

$$V = G\mathbf{u}(n) \quad (5.26)$$

$$U = G^T V \quad (5.27)$$

$$\gamma = \lambda + V'V \quad (5.28)$$

where the operator $(\cdot)'$ represents the complex conjugate transposition of the matrix (\cdot) .

$$\zeta = \frac{1}{\gamma + \sqrt{\lambda + \gamma}} \quad (5.29)$$

Now the filter gain can be computed,

$$K = \frac{1}{\gamma U}. \quad (5.30)$$

Next, the square root covariance matrix G is updated,

$$G(n) = \frac{1}{\sqrt{\lambda}} G(n-1) - \zeta VU' \quad (5.31)$$

And finally the weight vector is updated,

$$\hat{\mathbf{w}}_{n+1} = \hat{\mathbf{w}}_n + e(n)^* K' \quad (5.32)$$

where again λ is the forgetting factor, $e(n)$ is the error at instant n and $\mathbf{u}(n)$ is the input of the filter.

3. QR Decomposition Least Squares Filter (QRD-RLS).

The widespread acceptance of RLS filters has been impeded by a sometimes unacceptable numerical performance in limited precision environments. This degradation of performance is especially noticeable for the family of techniques collectively known as fast RLS filters. These fast algorithms are typically characterized by an absence of matrix multiplications and require $O(N)$ operations per sampling interval, where N is the number of coefficients in the filter. Finite precision implementations of these fast RLS filters have sometimes been observed to be numerically unstable. A possible solution comes from applying stable orthogonal linear algebraic transformations to the original RLS problem. These linear algebraic transformations are used to triangularize the multiplying matrix in a set of simultaneous linear equations.

One popular method for producing such a triangular structure is the QR decomposition [24], [25]. QR decomposition transforms the original RLS problem, which uses data covariance values, into a problem that uses only transformed

data values. This, in turn, causes the numerical dynamic range of the transformed computational problem to be reduced to one-half the dynamic range of the original RLS problem [6], which has immediate benefit concerning stability considerations. The filter weight equation is calculated by applying

$$\hat{\mathbf{w}}_n = \hat{\mathbf{w}}_{n-1} + \frac{\mathbf{c}(n)}{b(n)} e(n) \quad (5.33)$$

where

$$\mathbf{c}(n) = \frac{R^{-1}(n-1)\mathbf{a}(n)}{b(n)\sqrt{\lambda}} \quad (5.34)$$

$$\mathbf{a}(n) = \frac{R^{-T}(n-1)\mathbf{u}(n)}{\sqrt{\lambda}} \quad (5.35)$$

$$b(n) = \sqrt{1 + \mathbf{a}(n)^T \mathbf{a}(n)} \quad (5.36)$$

5.2.4.6 LMS/RLS Filter Sensor Fusion Approach

After the theoretical introduction to the LMS and RLS filters that will be tested, we can proceed to develop a sensor fusion approach using such filtering algorithms. As said in the theoretical introduction, in LMS/RLS algorithms we can assume the desired response $\mathbf{d}(n)$ is linearly related to the input vector $\mathbf{u}(n)$. By letting $\mathbf{d}(n)$ be the accelerometer angle signal α_{an} and $\mathbf{u}(n)$ be the gyroscope angle signal α_{gn} , we are satisfying that condition since α_{an} and α_{gn} are related linearly. LMS and RLS filters are widely used to cancel undesired components in signals, such as noise and echo. Therefore, we will use the gyroscope signal as the input to be filtered and the accelerometer signal as the desired signal. This configuration will remove the gyroscope's dynamic bias while it will filter the accelerometer's noise, since resultant LMS and RLS filters will have low-pass nature in almost all configurations. The accuracy and convergence time will vary enormously depending on the filter parameters and the input signals nature as is later shown in section 5.2.4.8. Figure 5.7 shows the diagram of the developed LMS/RLS sensor fusion approach.

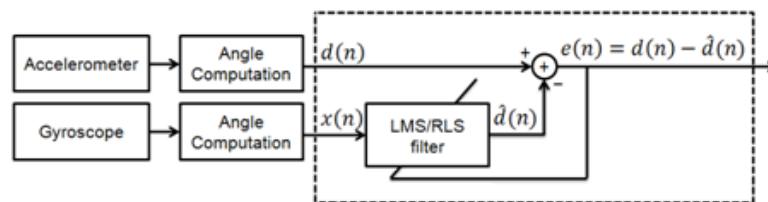


Figure 5.7: Diagram of the LMS/RLS filter sensor fusion approach.

5.2.4.7 Experiments

After presenting a brief theoretical introduction to the filters that will be employed in the comparative study, we proceed now to describe the performed experiments with the objective to determine how the different algorithms behave under distinct measurement conditions.

1. Application of adaptive filtering on synthesized signals

Adaptive filters present different behavior depending on the parameter configuration and the nature of the activity being monitored. In order to perform a comparative study between different adaptive filters to determine which one presents best performance, it is imperative to synthesize gyroscope and accelerometer angle signals. As the desired unbiased gyroscope signal is unknown, we will synthesize it so the Mean Squared Error (MSE) can be performed together with the approximation of the unbiased gyroscope signal we obtain from the output of the filter. Two different experiments are carried out, the first one applies adaptive filters on low intensity synthesized signals, i.e. signals containing low or no dynamic acceleration and, on the other hand, the second one applies adaptive filters on high intensity synthesized signals, i.e. signals containing occasional bursts of dynamic acceleration.

Three different signals are synthesized in both experiments: an accelerometer angle signal, an unbiased gyroscope signal and a biased gyroscope signal. The unbiased gyroscope signal is synthesized by building a sinusoidal signal having an amplitude of $\pm 180^\circ$ with a sampling frequency of 50 Hz and a period of 2 seconds. The accelerometer signal is synthesized differently for each experiment. For the low intensity activity experiment, it is built by adding random noise having a maximum amplitude of 10° to the unbiased gyroscope signal with the objective of simulating electronic noise. For the high intensity activity experiment, it is built by adding two different random noises to the unbiased gyroscope signal: the first random noise is also applied to simulate electronic noise and has a maximum amplitude of 10° and the second random noise is applied to simulate bursts of dynamic acceleration and has a maximum amplitude of 150° . The duration of the bursts is limited to 1 second. Finally, the biased gyroscope signal is built for both experiments by adding a linear bias, having the same slope than real measured signals, to the unbiased gyroscope signal. All signals are 8 minutes long. Figures 5.8 and 5.9 show the accelerometer and biased gyroscope signals for the low intensity and high activity monitoring respectively.

$$MSE = (\alpha_g - \hat{\alpha}_g)^2 \quad (5.37)$$

The biased gyroscope angle signal and the accelerometer angle signal are used as inputs of the adaptive filters. The final MSE is calculated by averaging the MSE, calculated using (5.37), of 50 executions for each of the filter algorithms,

where α_g is the unbiased gyro signal. New signals are synthesized for each one of the executions so they are slightly different as the accelerometer noise has a random nature. Figures 5.10-5.13 show the calculated MSE for the different adaptive filtering employed algorithms for both low intensity and high intensity synthesized signals.

The MSE signals are divided in 3 equal parts for which the average MSE is calculated. These average MSE values will give us information about the filter performance at the beginning, the middle and the end of the time period. Tables 5.2 and 5.3 show these values for the low and high intensity experiments respectively.

2. Application of adaptive filtering on real signals

After running theoretical simulations to obtain the MSE of each of the adaptive filters we can now apply them to real signals gathered from the accelerometer and gyroscope included inside the IMU. Two different low and high intensity exercises are monitored, the first one corresponds to a low intensity knee rehabilitation exercise where the IMU is placed on the shank of the patient. The patient moves up and down the shank covering a range of approximately 120° during about 2 minutes. Figure 5.14 shows the recorded signals for such an exercise. The second exercise corresponds to series of high intensity vertical jumps performed by a subject, with the IMU attached to the hip, during about 30 seconds.

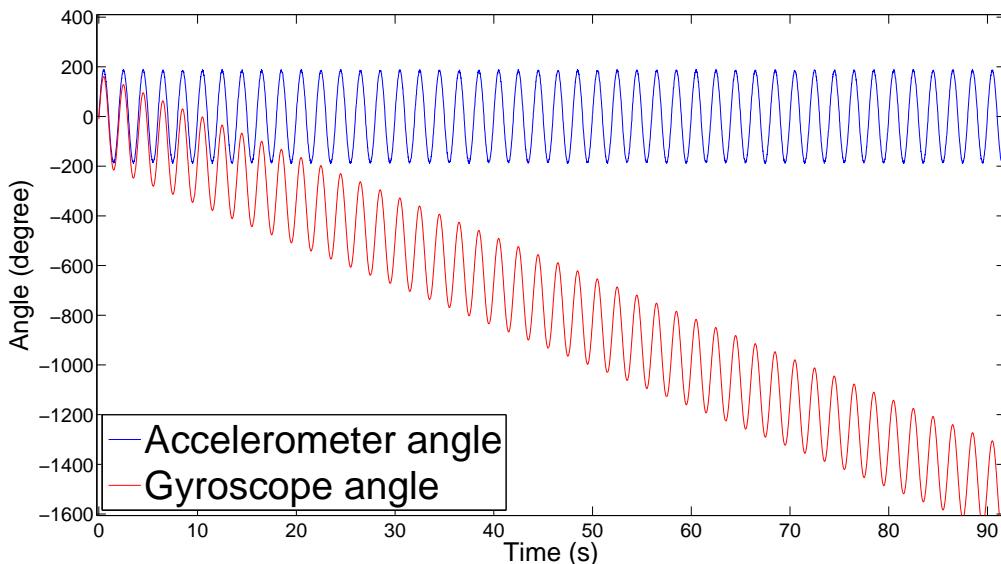


Figure 5.8: Synthesized gyroscope and accelerometer angle signals for the low intensity activity experiment.

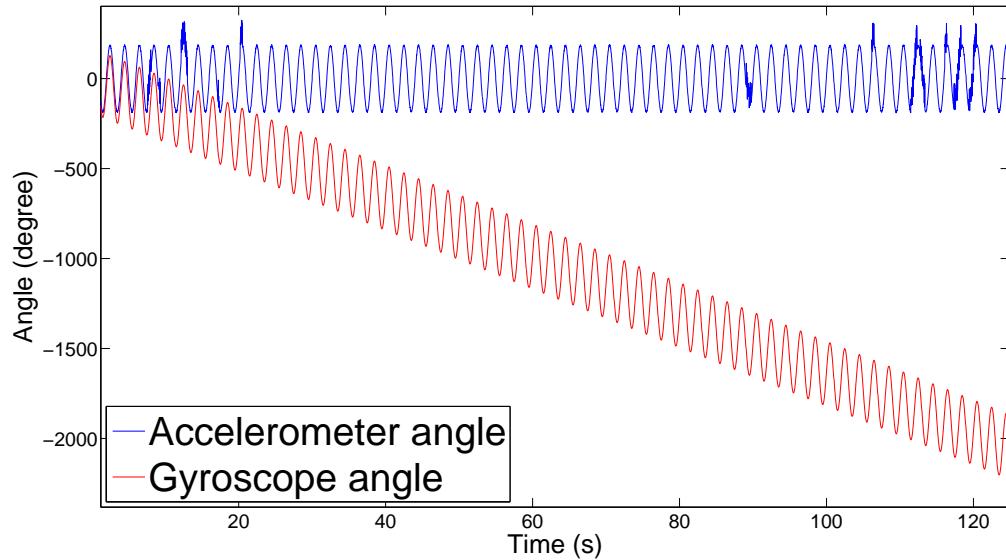


Figure 5.9: Synthesized gyroscope and accelerometer angle signals for the high intensity activity experiment.

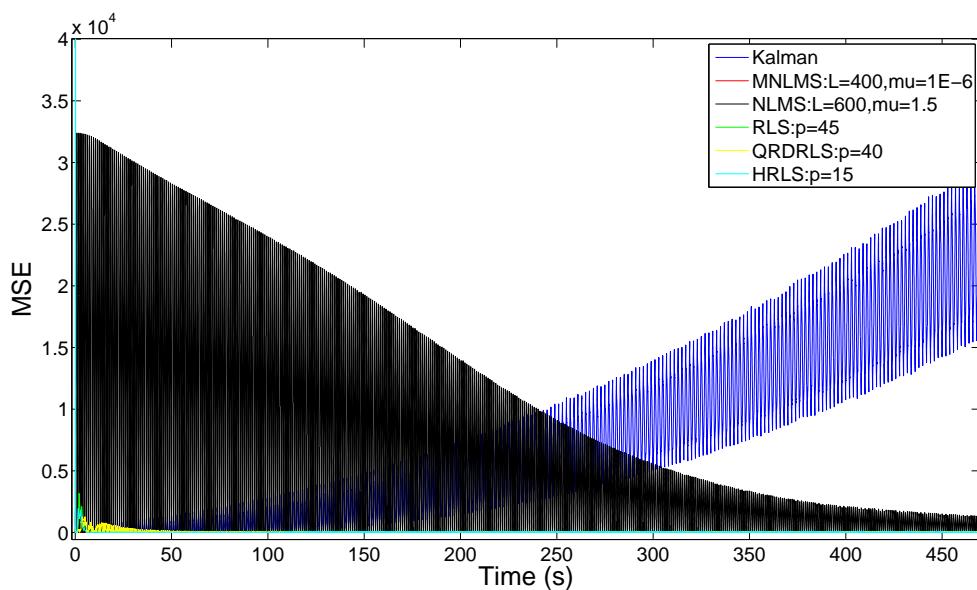


Figure 5.10: Average MSE calculated for 50 executions of the adaptive filtering algorithms on low intensity synthesized signals.

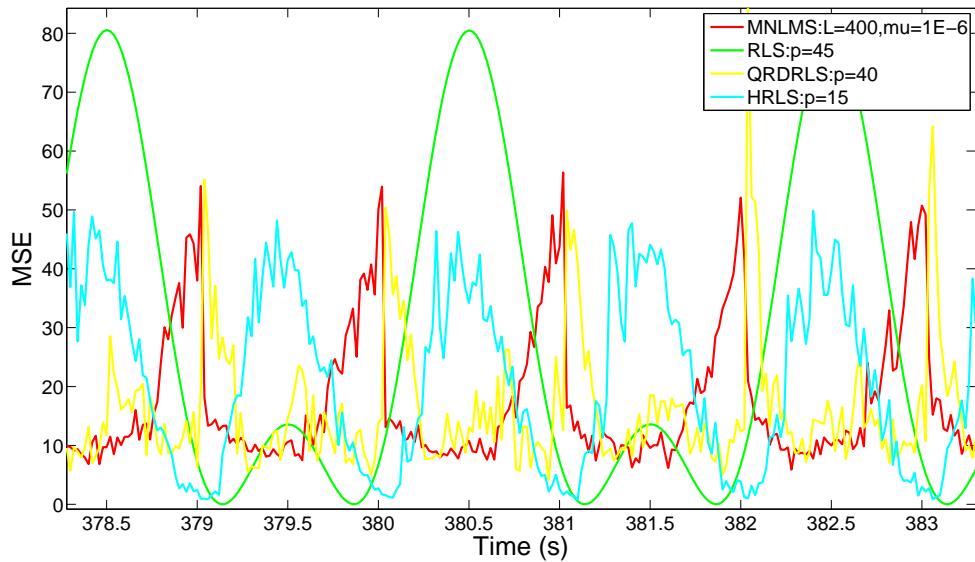


Figure 5.11: Zoom of the Average MSE calculated for 50 executions of the adaptive filtering algorithms on low intensity synthesized signals.

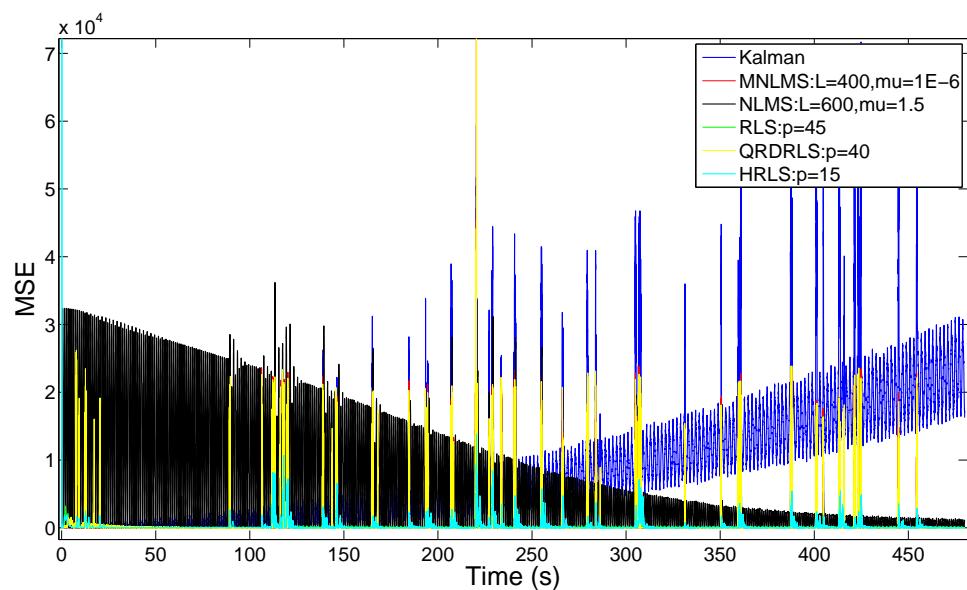


Figure 5.12: Average MSE calculated for 50 executions of the adaptive filtering algorithms on high intensity synthesized signals.

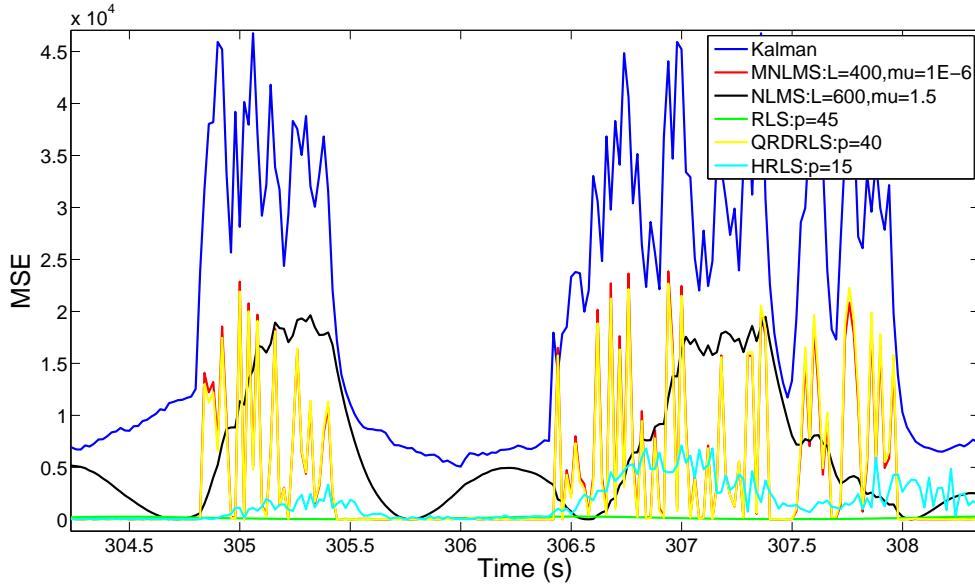


Figure 5.13: Zoom of the Average MSE calculated for 50 executions of the adaptive filtering algorithms on high intensity synthesized signals.

Table 5.2: Average MSE values for each one of the 3 periods in which the low intensity synthesized signal is divided. S_z : Observation signal noise covariance, Var_a : Process noise covariance related to the acceleration signal, Var_g : Process noise covariance related to the gyroscope signal.

Filter	Parameters	Mean MSE (dB)		
		Period 1	Period 2	Period 3
Kalman	$S_a=0.003$			
	$Var_a=0.03$		59.64	75.62
	$Var_g=0.005$			84.33
NLMS	Filter Size=600, Step Size=1.5	82.17	74.49	61.48
MNLMS	Filter Size=400, Step Size= 1×10^6	35.37	26.07	24.9
RLS	Filter Size=10	44.24	23.03	24.48
HRLS	Filter Size=10	47.39	26.42	26.35
QRDRRLS	Filter Size=40	36.81	23.33	24.23

Table 5.3: Average MSE values for each one of the 3 periods in which the high intensity synthesized signal is divided. S_z : Observation signal noise covariance, Var_a : Process noise covariance related to the acceleration signal, Var_g : Process noise covariance related to the gyroscope signal.

Filter	Parameters	Mean MSE (dB)		
		Period 1	Period 2	Period 3
Kalman	Sa=0.003			
	Var_a =0.03	63.15	77.30	84.89
	Var_g =0.005			
NLMS	Filter Size=600, Step Size=1.5	82.18	75.07	63.03
MNLMS	Filter Size=400, Step Size= 1×10^6	53.13	75.07	51.02
RLS	Filter Size=45	42.34	41.97	40.16
HRLS	Filter Size=15	51.58	48.30	43.09
QRDRLS	Filter Size=40	53.46	55.32	50.85

Figure 5.15 shows the recorded signals during the exercise. We can observe at a glance that the peaks produced by the dynamic acceleration lead to erroneous angle measurements in the accelerometer angle signal.

The acceleration and gyroscope angle signals are used again as the adaptive filters' inputs and the output is calculated for a wide range of values of different filter parameters. Figures 5.16-5.18 show the output for the best configurations of each one of the tested filters in both exercises.

5.2.4.8 Results discussion

1. Results of theoretical simulations

Based on the theoretical simulation results we can observe several differences on the performance of the filters between the two experiments. In the case of the low intensity synthesized signals, the lowest MSE is obtained for the QRD-RLS filter having a filter size of 40 coefficients. MN-LMS also behaves properly, however, its length is 10 times higher than the QRD-RLS algorithm. This might not be a problem for applications where the sampling frequency is high or no real-time applications, but as in our application, when the sampling frequency is rather low (50 sps), a long filter may cause an initial delay that could be intolerable. On the other hand, QRD-RLS has a higher computational load than MN-LMS which could also lead to its unsuitability for real time applications. The convergence time is essential as it determines the time until the output of the filter

can be considered to represent accurate measurements. In terms of convergence, QRD-RLS is slightly faster than MNLMS mainly due to the faster convergence nature of the RLS algorithms family.

RLS also presents an acceptable accuracy, so it may also be considered as a good option since it is the most efficient of the tested filters at filtering random noise. We can also observe the convergence speed boosting effect of the normalization term in the MN-LMS weight update equation (5.20) which makes it to be much faster than standard LMS and even reaching rates of convergence close to QRD-RLS.

The Kalman filter algorithm shows a slight divergence since its weighted feedback error correction misbehaves when the difference between the estimation and the actual value is large. Therefore, it is not a good solution when the dynamic bias value is rather large.

Several parameter values were tried to determine if the divergence was due to an improper parameter selection, but all configurations led to a progressive divergence as the angle wander increased. In conclusion, all filters outperformed the Kalman filter approach for the conditions of our experiment.

In the case of high intensity synthesized signals the results are somewhat different as QRD-RLS shows poor performance at filtering the dynamic acceleration peaks, which makes it a bad option for situations where the subject is moving intensely. Among all the tested filters, RLS has thrown the best results since it almost completely filters all the peaks and its MSE values are close to those from the low intensity experiment.

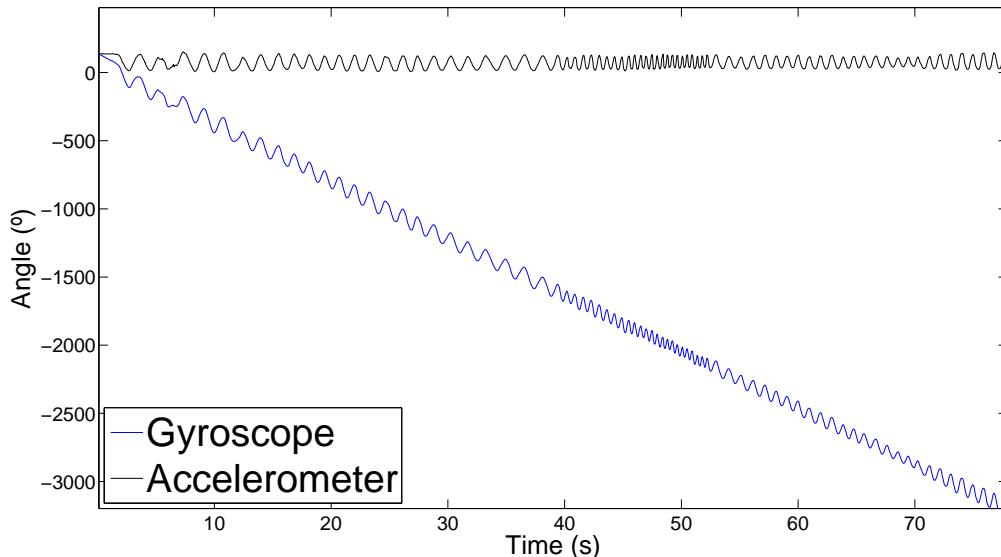


Figure 5.14: Accelerometer and gyroscope angle signals gathered during a knee rehabilitation exercise.

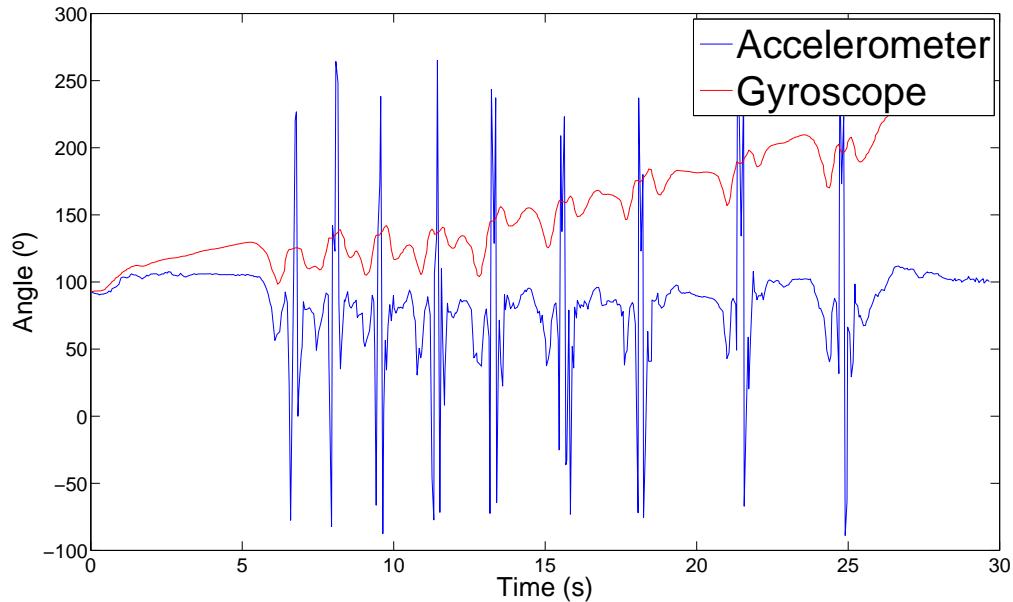


Figure 5.15: Accelerometer and gyroscope angle signals gathered during a series of vertical jumps.

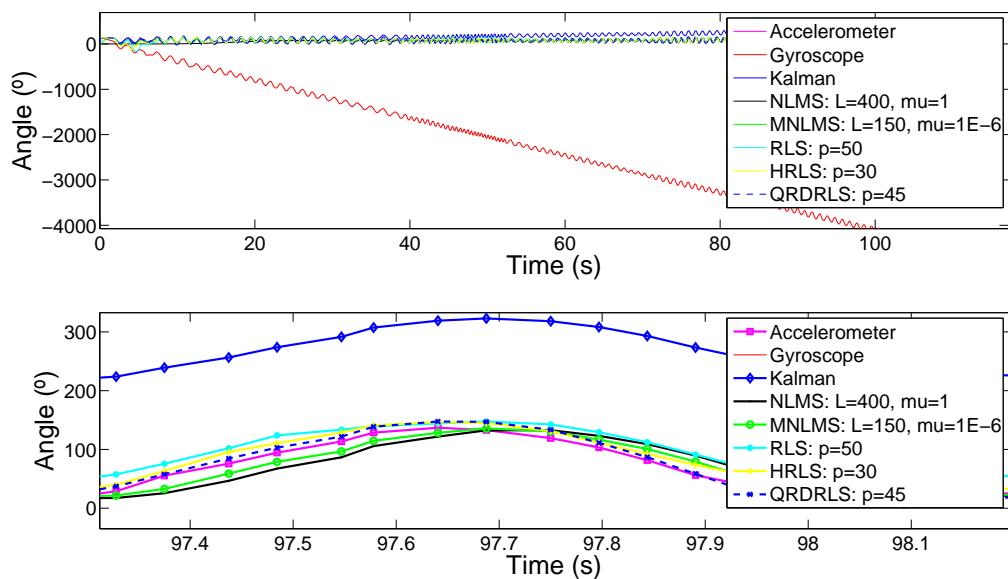


Figure 5.16: Output of the adaptive filters for real signals gathered during a knee rehabilitation exercise. Complete signals (a), zoomed signals (b).

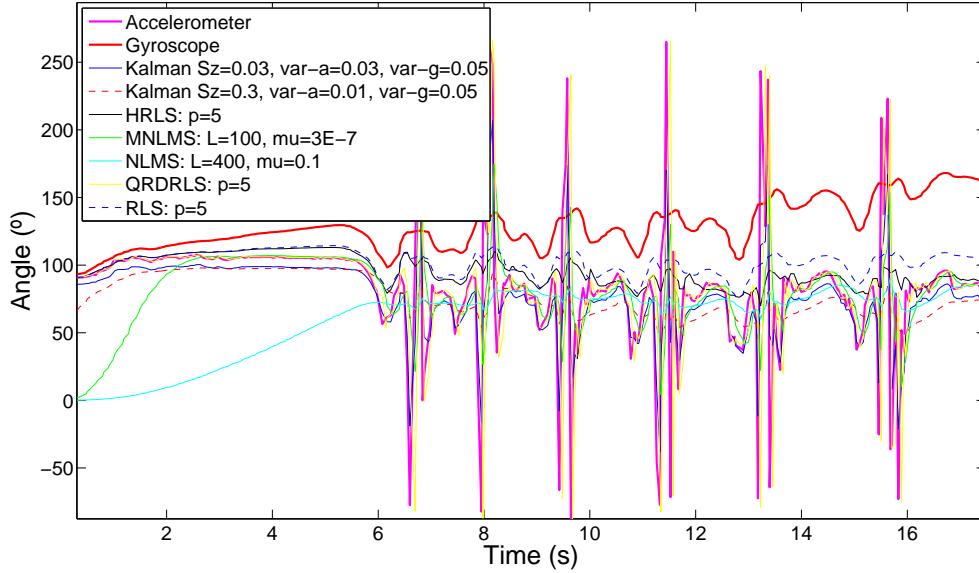


Figure 5.17: Output of the adaptive filters for real signals gathered during a series of vertical jumps. Complete signals.

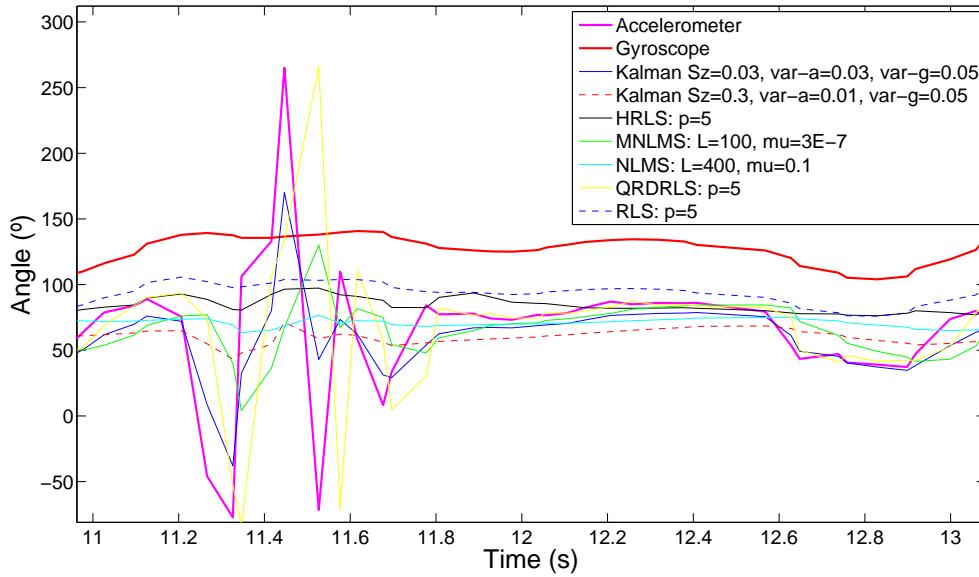


Figure 5.18: Output of the adaptive filters for real signals gathered during a series of vertical jumps. Zoomed signals.

The LMS variations also present difficulties to properly filter the peaks and, as expected, converge slower. These facts make them a bad choice under the new

experiment conditions.

Again, Kalman filter slightly and slowly diverges and it does worse than RLS at filtering the peaks. Summarizing, RLS has shown to be the most robust algorithm under all the tested conditions for synthesized signals.

2. Results of application on real signals

As we can observe in Figure 5.16, H-RLS (filter size=30), RLS (filter size=50) and QRD-RLS (filter size=45) offer the best performance for low intensity exercises as they completely remove the bias and filter quite efficiently the random noise present in the accelerometer signal. Kalman filter does good at filtering the noise but does not converge to the proper solution as it presents an increasing positive bias. LMS filters also converge to the desired signal but slower than RLS filters and are also less effective filtering noise. When applied to the high intensity exercise signals, RLS filter shows the best performance since it is able to remove the dynamic bias and filter the dynamic acceleration peaks as depicted in Figure 5.18. We can also observe that as seen in the theoretical simulations QRD-RLS is not a good solution as it does not filter the peaks at all.

All the results obtained under the application of the adaptive filters on real signals confirm the results shown by the theoretical simulations.

5.3 Focus of research

As for future work goals, we plan to test other algorithms based on Kalman filtering, such as the Extended Kalman Filter, The Kalman-Bucy Filter, the Unscented Kalman Filter, as well as different modern particle filters with the objective of finding the best filter to preprocess the Inertial Measurement Units output signals and remove the effects of random noise and dynamic bias. Magnetometer data will also be included in the sensor fusion approaches in order to obtain better accuracy. Therefore, a triple-input diagram will be needed.

CHAPTER 6

Applications of the System

During the past sections we have described a complete human body position monitoring system, we have shown the employed hardware and described new and existing calibration procedures, as well as the different signal preprocessing methods that need to be applied. After the complete system is developed, we can start applying it to different areas of health care in order to help prevention, diagnosis and rehabilitation of many diseases and injuries.

This section has the goal to present the basics of human body monitoring applied to telerehabilitation, Activities of Daily Life (ADL) monitoring, gait and posture monitoring, as well as other medical applications such as detection of Epileptic seizures, sleep disorders and Parkinson tremor, dyskinesia and bradykinesia episodes. This system can also be employed in sport science.

6.1 Telerehabilitation

Most surgical procedures in patients suffering from diverse trauma in the limbs entail rehabilitation programs with a duration oscillating between several months and a few years [41]. The rehabilitation programs usually start with a phase were the patient gets treatment from physiotherapists including massages, infrared heat, short wave

stimulation, etc. During this first phase the patient has to move unavoidably to the medical center. When this first treatment phase is completed the patient gets into a phase where a series of low intensity exercises have to be performed to strengthen the muscles and rehabilitate the affected zone [51], [39]. This kind of exercises do not require the continuous presence of supervising staff since they are usually simple to carry out and can be easily done by the patient once they have been told how to proceed. Due to the overcrowding of rehabilitation centers there is a long waiting list to start most of the treatments. When the patient is called out, he is usually scheduled during working hours.

As a consequence of the overcrowding, patients do not receive a complete treatment and are usually sent home prematurely where they perform the exercises under no supervision.

Telerehabilitation improves the efficiency and shortens the duration of the treatment since it allows the patient to remotely carry out the second phase without having to renounce to medical supervision.

The telerehabilitation system needs instrumentation to monitor the patient, algorithms to process the gathered data, a human-machine interface where the patient can observe in real time the execution of the exercises through a 3d avatar, and a patient-doctor communication platform that allows the latter to supervise the rehabilitation process, check the proper execution of the exercises , update the rehabilitation plan and schedule new sessions and exercises.

Wagyro has been used during the past year in Telefonica's R&D Rehabitic project. Rehabitic is a knee telerehabilitation ongoing project that uses Wagyro hardware and offers real time 3D animation of the movements a patient is performing. The patient can select between different exercises. The duration, intensity and frequency of occurrence of every exercise is advised and adjusted by the therapist, who has access to the recorded data through a web platform. The system is also able to ask questions to the patient, for example: How much pain do you feel today? This way the therapist gets a better understanding how the patient is feeling. Also the therapist is able to give feedback to the patient.

Wagyromag will be used in a second phase of the project where other parts of the body will be included, such as the joints of the arms or the back. The more complex kinematics of the back needed from a sensor with a higher degree of freedom which Wagyromag can provide.

6.2 Gait and Posture analysis

Gait and posture analysis are of utmost importance in prevention and diagnosis of osseus, muscular, degenerative and nervous injuries and diseases. Many studies have been carried out [52],[66] to try to determine when a patient is more likely to fall due to their faulty gait and posture patterns, as well as other studies that focus on detection of

limbs injuries based on gait. Most of these studies use human body position monitoring systems that are based on video or inertial sensors. Inertial sensors allow the study to be extended to different scenarios, as the patient can be monitored wherever he is as long as he is wearing the IMU. Video systems are limited to a closed scenario which is surrounded by cameras. Such video systems offer better accuracy but their price is many times higher than inertial sensor systems.

The use of neural networks and classification methods permits the system to detect movement patterns that can be identified with several diseases.

6.3 Analysis of activities of daily life

Nowadays, most of elderly people have different teleassistance and emergency systems installed at home. These systems are based on the wireless devices known as 'panic buttons'. This kind of buttons are usually worn by the patients in a necklace or in a bracelet, so when they are in an emergency situation, such as a dangerous fall, a heart attack or other kind of discomfort, they can push it and an operator from the emergency call center contacts them by telephone. If the person is not responding, medical help is sent immediately.

This kind of systems are useless when the person falls unconscious and is unable to push the button. As a consequence, inertial systems are starting to be embedded on panic buttons so falls and long inactivity periods can be detected. Likewise, it is very interesting for the relatives of elderly people to track and control their daily activity. Due to the reticence of elderly people to have cameras installed at home so their relatives can periodically observe their activity to check how are they doing, inertial systems can be employed to somehow monitor their basic ADL such as walking, sitting and sitting up, lying down, etc. This kind of systems are able to alert from abnormal activities during their routine, such as periods of inactivity, falls, sleep habits, etc. Researchers are also working to develop mobile teleassistance so care systems' range is extended, offering assistance while the person is out of home.

It is strictly necessary to use human body position monitoring systems in complete and reliable assistance systems. Figure 6.1 shows the diagram of a teleassistance system based on inertial sensors carried by the person in a necklace or in a bracelet.

6.4 Detection of seizures

6.4.1 Epilepsy

Other important issues are related with nocturnal seizures. Epilepsy patients usually keep a register of the suffered attacks, so they can hand it to their doctor during con-

trol and medication revision sessions. Occasionally, patients suffer seizures while they are asleep. These kind of attacks are known as nocturnal seizures, and they may go unnoticed to the patient, thus, not being annotated in the register. It may happen that a patient suffers many nocturnal seizures and few or no seizures during the day. As a consequence, during the control sessions with the doctor the patient would tell the doctor the medication is working properly, when it is not. To avoid such a situation, the patient may sleep with an attached IMU so nocturnal seizures can be detected and stored in the memory, allowing the doctor to notice that nocturnal attacks are happening. Hence, IMUs may be a useful tool to improve the monitoring and detection of epileptic crisis.

A pilot project will soon be started in collaboration with the Epilepsy Unit of the Virgen de las Nieves Rehabilitation and Traumatology Hospital. The goal is to monitor patients having nocturnal epileptic seizures in order to identify their frequency and duration. Some works have already been carried out aiming to detect nocturnal epileptic seizures [42], [12], [17].

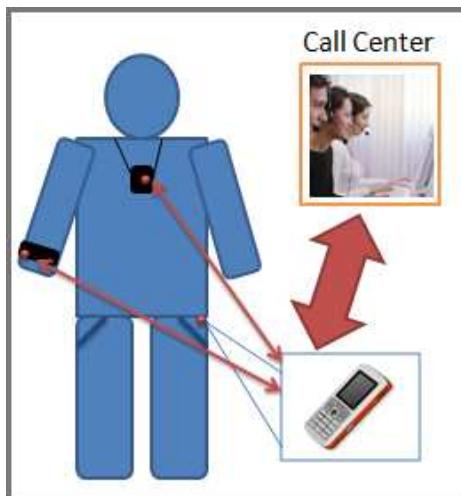


Figure 6.1: Diagram of a mobile teleassistance system based on inertial sensors.

6.4.2 Parkinson

Inertial systems can also be used to detect tremor [50], bradykinesia [61] and dyskinesia [46] episodes of Parkinson's disease patients. Bradykinesia is the term for defining slow execution of the movement, tremor is the term used to define repetitive periodic movements within a certain frequency range and dyskinesia is a movement disorder which consists of effects including diminished voluntary movements and the presence of involuntary movements. Since all these disorders are related with motor activities, they can be tracked and detected by an IMU.

6.5 Sleep disorders

Sleep disorders can also be monitored by inertial sensors. From time to time patients suffering from sleep disorders have to spend the night at sleep units located in hospitals so their heart rate, brain and motor activity are monitored. A low cost inertial sensor would allow patients to be monitored at home during periods between control sessions at the hospital. By monitoring them at home, sleep specialists can follow the evolution of the patient daily and offer a better medical treatment. There exist some commercial devices such as Fitbit [4] that measure and monitor activity during sleep periods and measure the quality of the rest.

6.6 Sport science

Another field of operation for our system could be sports science. A golf swing, a football or a tennis racket could be tracked and visualized on a computer. With this technology, it is possible to optimize the performance of the athlete.

As described in [27] such systems can also be used for judging in a sports competition for example a snowboarding competition, which gives the judges the opportunity to focus on skills they normally are not able to focus on. Nowadays the judges are busy focusing on the heights, turns or the amount of loops, which could be taken over by an wireless measurement unit. So it might be possible to develop an automatic scoring system that uses the data recorded by the wireless measurement unit. Such systems could also be used for training since they should show the same values in a competition than in the training.

Due to the limitation of the accelerometer to ± 3 g, it might not be possible to measure every kind of movement. A complete analysis of the application of inertial sensors in sport science can be found in [43]. It includes measurements of the acceleration of a golf player's hip, wrist and the golf swing. The measurements show much higher values than ± 3 g. Therefore some hardware changes might be necessary to cover these applications.

Commercial applications are also possible; inertial sensors could be used to monitor the movement of the body during an exercise session and a system could act as a personal trainer. Such system could be installed in gyms or at home.

6.7 Focus of research

There exist an enormous field of applications of human body monitoring systems. In order to find possibilities of application is always good to talk to experts of different fields, such as doctors. Engineers lack of knowledge to find applications of human body

monitoring in health applications, so such interaction with medicine experts is more than recommended.

CHAPTER 7

Conclusions

Throughout this work, the different parts of a complete human body position and movement monitoring system based on inertial sensors have been presented. Chapter 1 presents a general state of the art of human body monitoring systems. Chapter 2 presented the general structure of inertial units, the physical principles and structure of MEMS accelerometers, gyroscopes and magnetometers. It also presents the IMUs that have been used and are going to be used in this ongoing research. More specifically there is an emphasis on the developed prototypes. This work presents Wagyromag, a new version of Wagyro (presented in previous work) which offers the old features and new ones such as local processing capabilities, 3D magnetic field measuring, local storage in a microSD card, temperature measuring and a buzzer.

Chapter 3 presents the two developed software tools to store and represent the measurements in real time, highlighting Visimu, a Java based application that is still being developed to offer a general framework to gather, represent and store data from any kind of IMU, and also to interact with a web platform that visualizes the data in real time.

Chapter 4 presents different calibration algorithms. Some of them have already been tested showing good performance, and other new algorithms, such as ellipsoid fitting procedures, will be tested on the forthcoming research.

Chapter 5 contains most of the research carried out in the past months. Throughout this chapter different adaptive filtering sensor fusion techniques have been presented

with the objective of obtaining a more accurate system to measure the angle of a subject's limbs. The sensor fusion concept has been explained and two different approaches have been presented: the first one based on Kalman Filtering, and the second one based on LMS and RLS adaptive filtering variations, more specifically N-LMS, MN-LMS, standard RLS, H-RLS and QRD-RLS. A comparative study has been carried out to determine which filters offer best accuracy and performance. To that effect, accelerometer and gyroscope signals derived from low and high intensity exercises have been synthesized to calculate the MSE using the ideal angle signal as the desired signal. It has been shown that filters behave differently depending on their operation parameters and the nature of the input signals. For low intensity input signals, QRD-RLS filter offered the best results as it was able to properly remove the dynamic bias present in the gyroscope angle signal in a shorter period of time. However, for high intensity input signals, QRD-RLS showed poor efficiency as it was unable to filter the dynamic acceleration peaks present on the accelerometer angle signal. In this case, standard RLS filter obtained the best results as the trade-off between dynamic bias removal and dynamic acceleration peaks filtering was more balanced than in the rest of the tested filters. In any case, all filters outperform Kalman filtering approach as it does not behave properly when the error between the estimated and the actual value is rather large, i.e. when the dynamic bias slope is quite large.

Overall, standard RLS has revealed itself to be the most robust algorithm for obtaining accurate measurements under both low and high intensity movements conditions.

In order to corroborate the results obtained in the theoretical simulations carried out on synthesized signals, all filters have been applied on real signals gathered from a low intensity knee rehabilitation exercise and a high intensity exercise composed of a series of vertical jumps. For both exercises we obtained the same results as in the theoretical experiments.

Chapter 6 presents different possible applications of the developed system. Some of them are already on test and are giving good results, such as the telerehabilitation platform, and other will be soon started, such as monitoring epilepsy patients to detect nocturnal seizures. In general, this system offers a wide range of features that can be used and applied in many different research or commercial fields, such as posture and gait monitoring, sports science, military science, inertial navigation, analysis of activities of daily life, fall detections, etc.

Monitoring human movement is still a hot topic that will generate research opportunities in the following years. By monitoring human body movement we are approaching more and more the technology to humans and making easier the generation of human-machine interfaces that integrate us and our actions in technologic systems that lead to a more prosperous society.

APPENDIX A

Wagyro: Schematics and Layouts

This Appendix shows the fabrication document of Wagyro. It includes the schematics and the layouts.

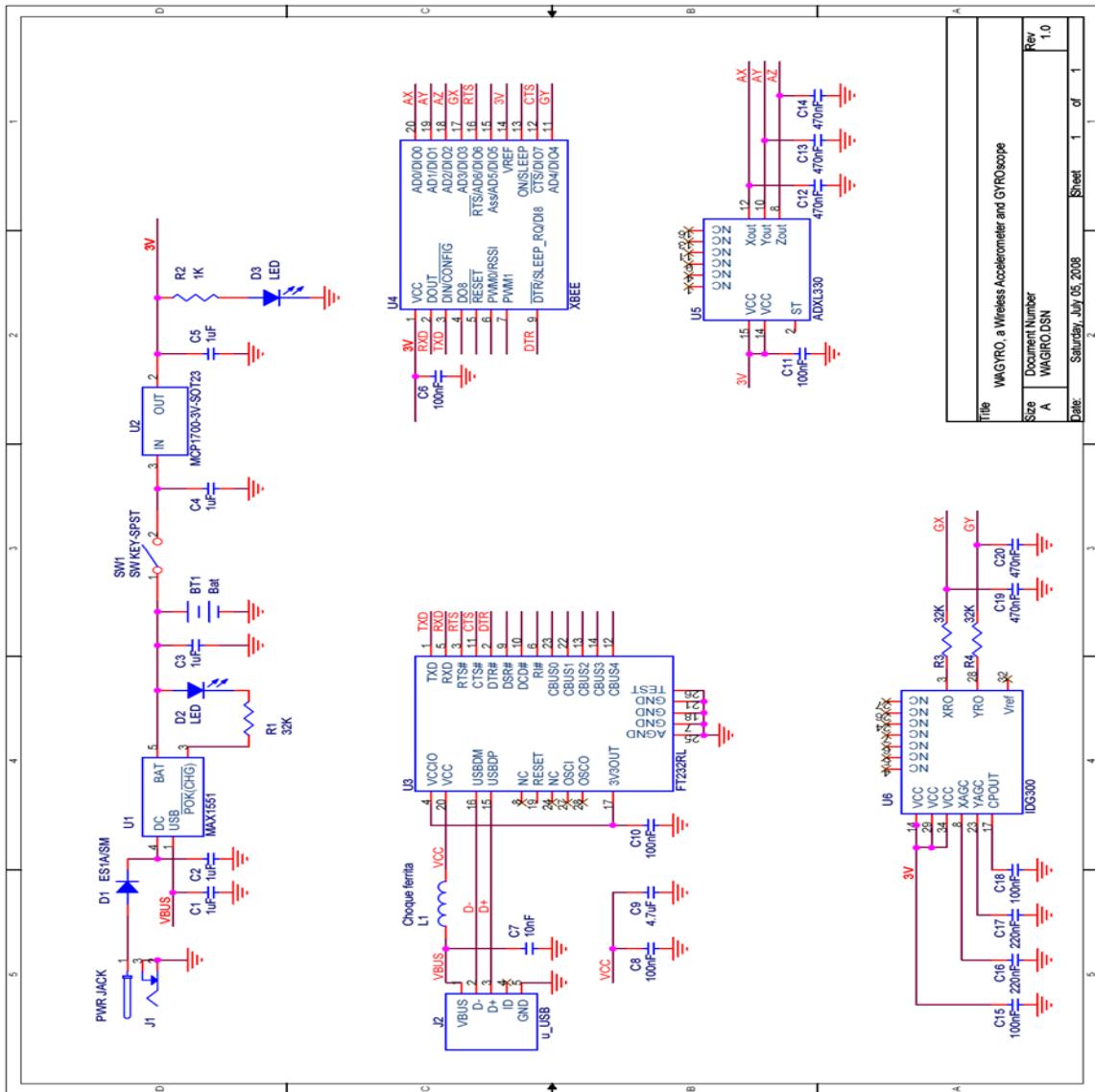


Figure A.1: Schematics of Wagyro.

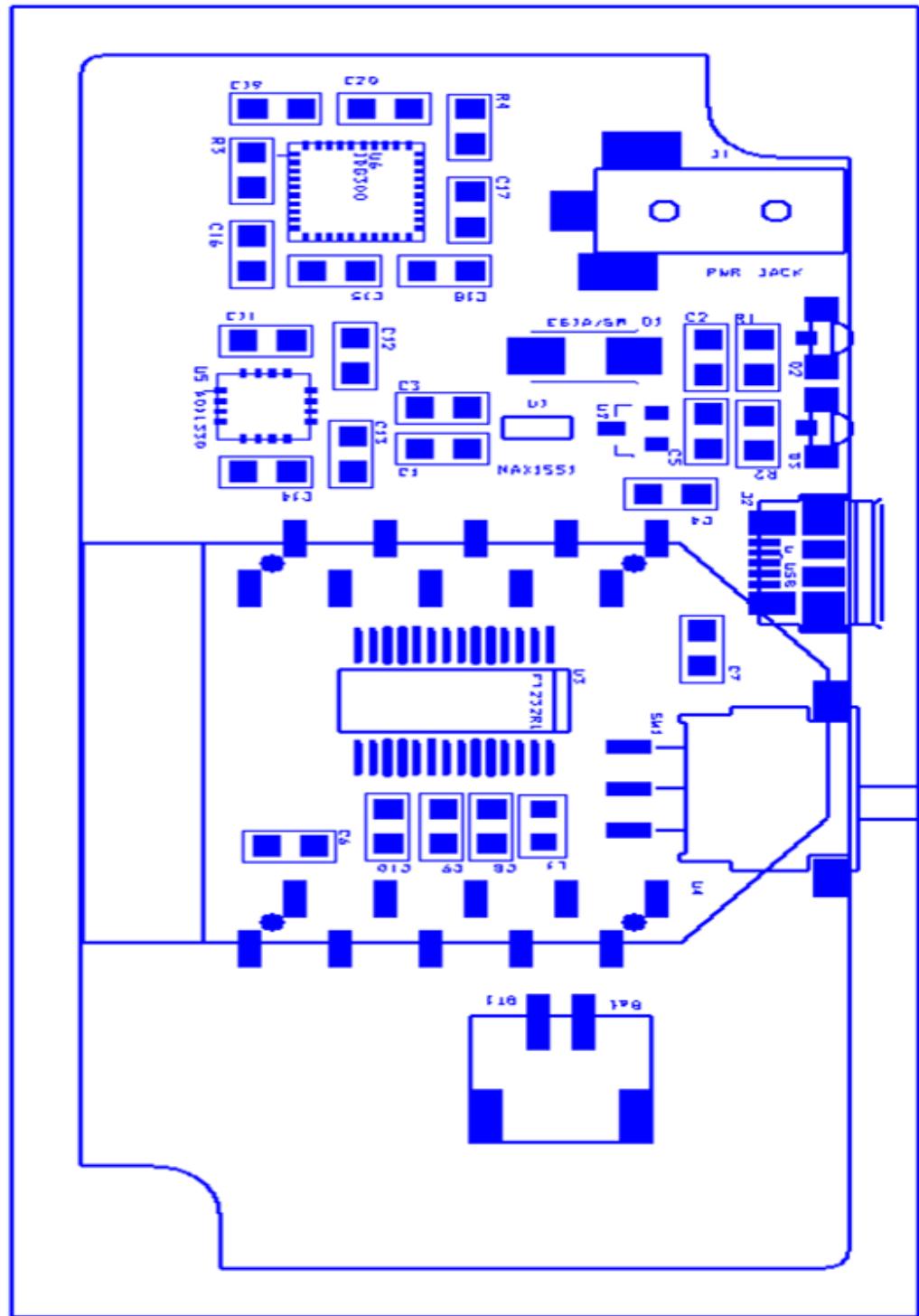


Figure A.2: Components layout.

APPENDIX B

Wagyromag: Schematics and Layouts

This Appendix shows the fabrication document of WagyroMag. It includes the schematics and the layouts.

B.1 Schematics

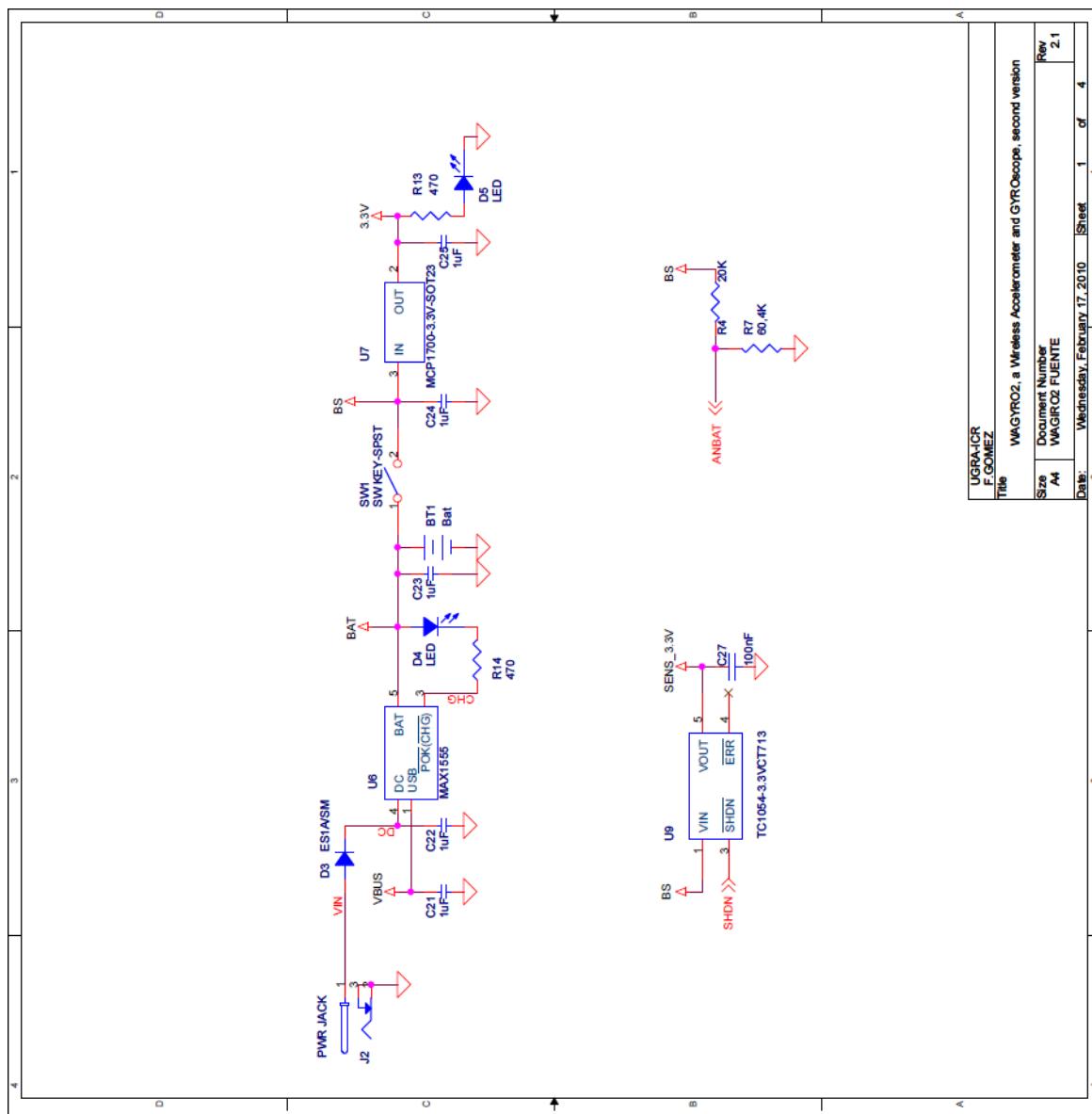


Figure B.1: Schematics of Wagyromag (1/4).

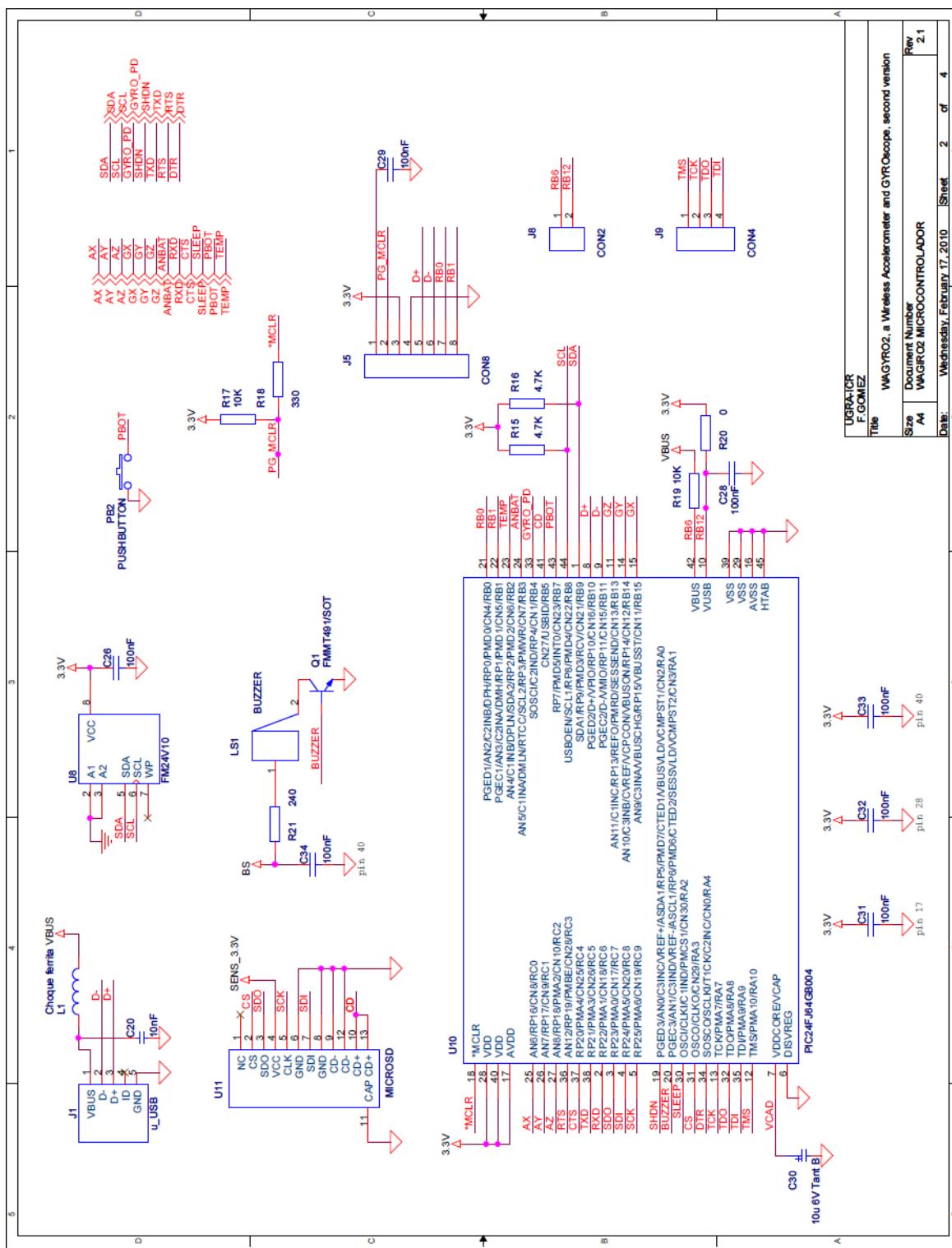


Figure B.2: Schematics of Wagyromag (2/4).

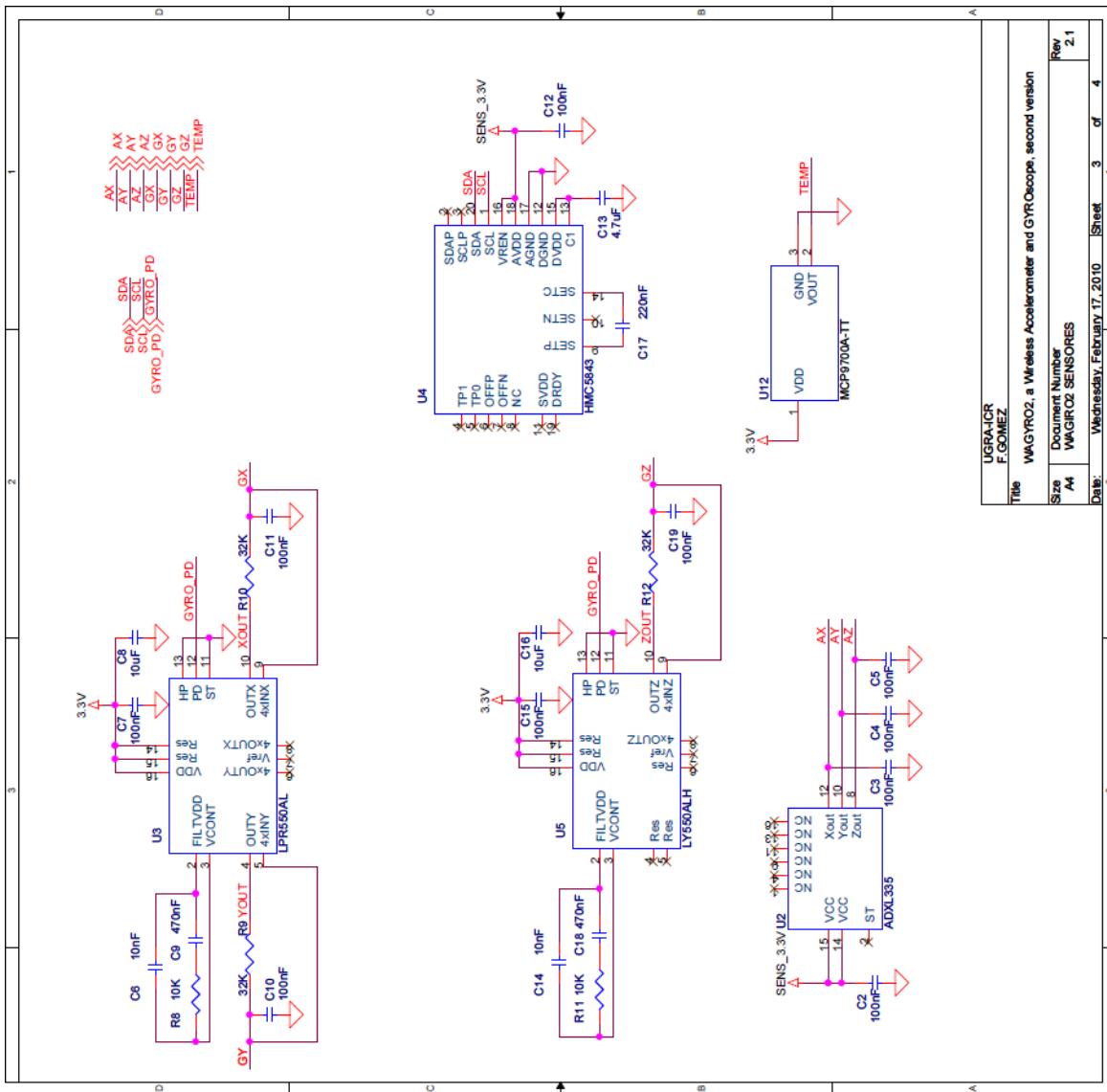


Figure B.3: Schematics of Wagyromag (3/4).

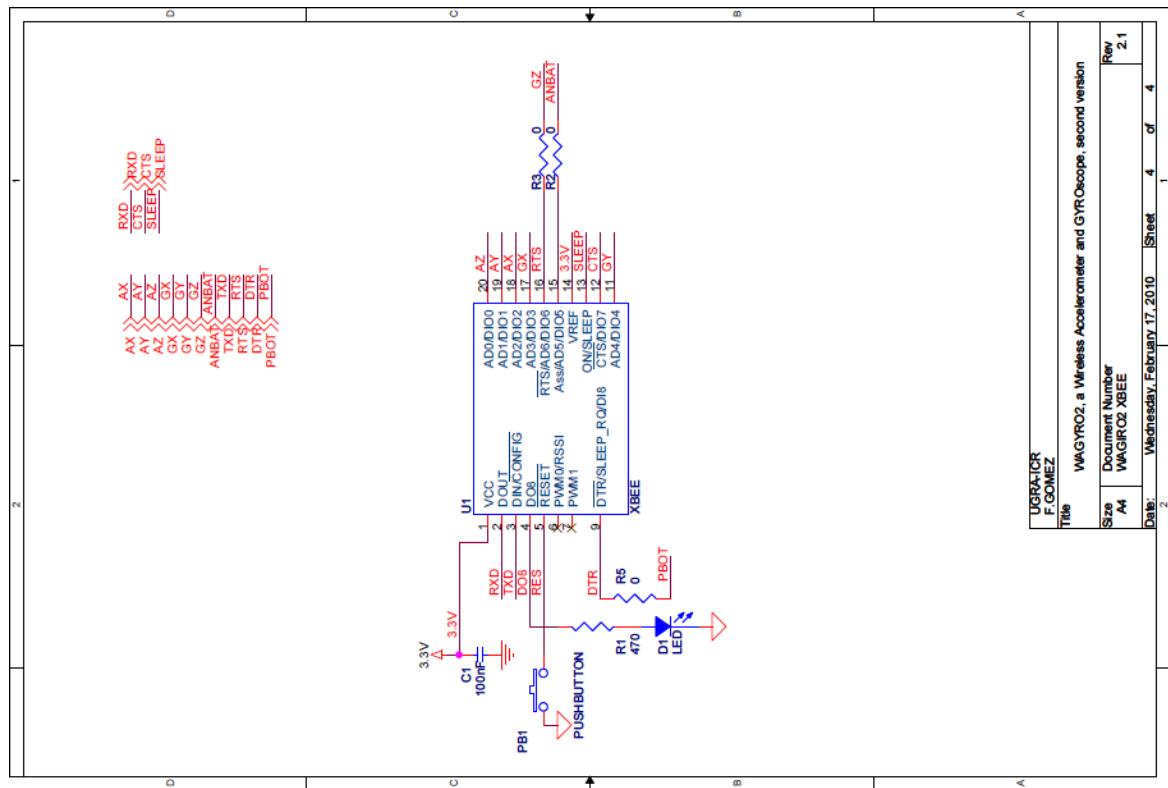


Figure B.4: Schematics of Wagymag (4/4).

B.2 Layouts

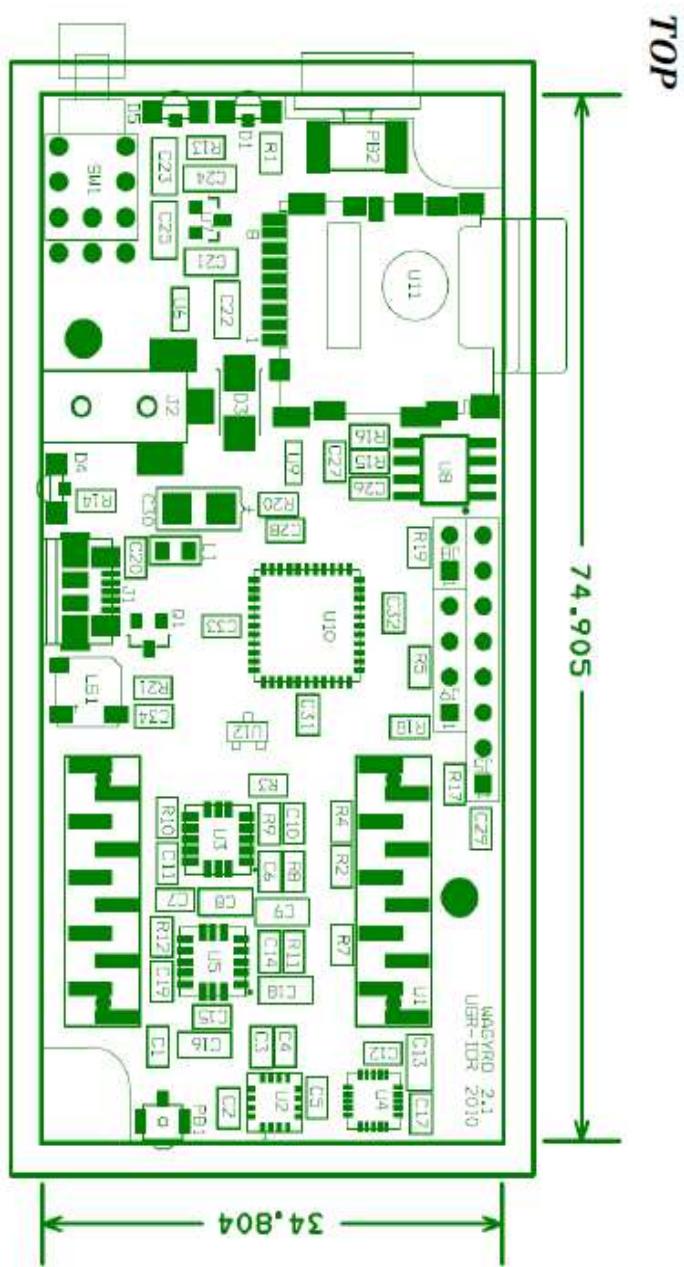


Figure B.5: Components layout: top.

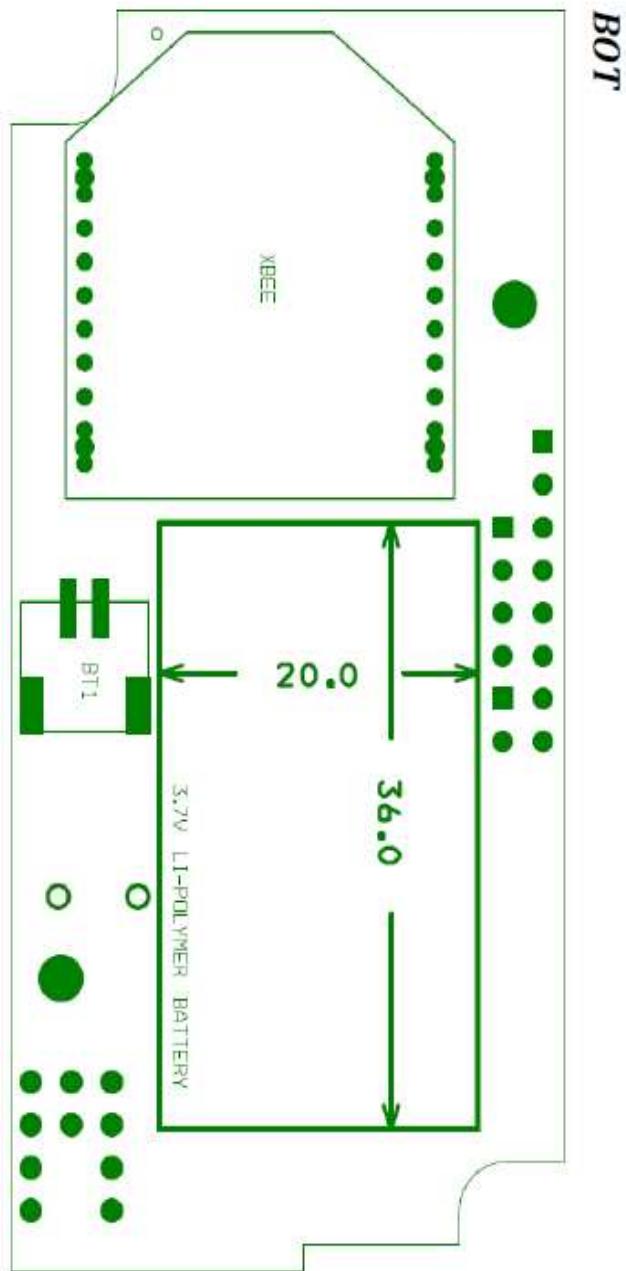


Figure B.6: Components Layout: bottom.

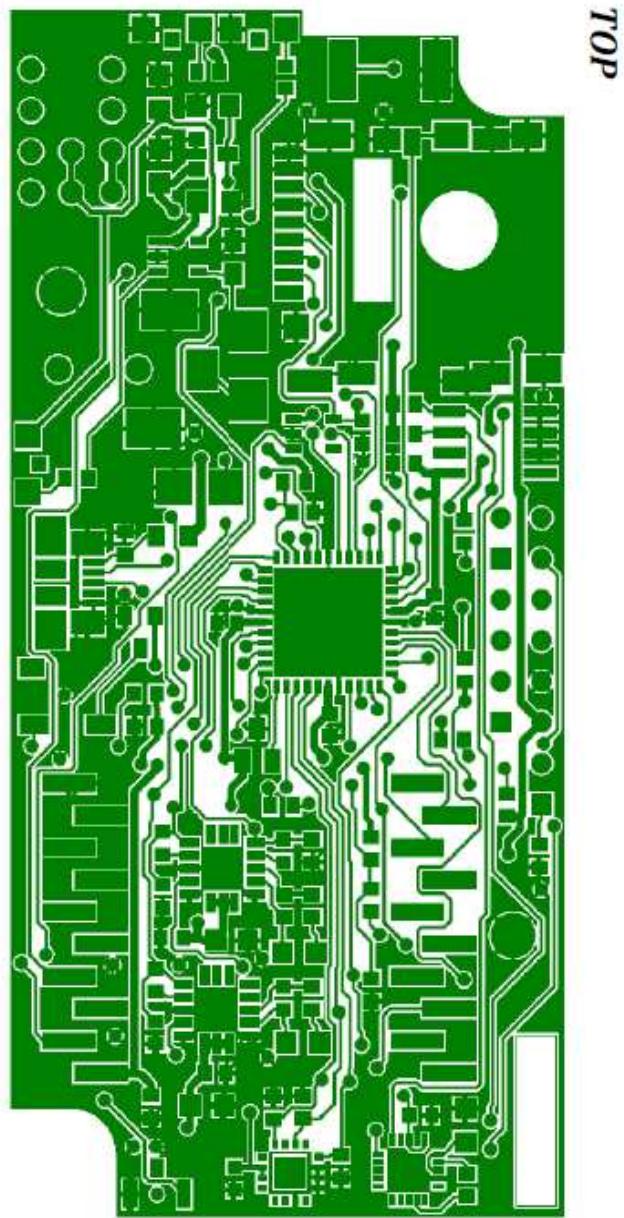


Figure B.7: Conductive traces layout: top

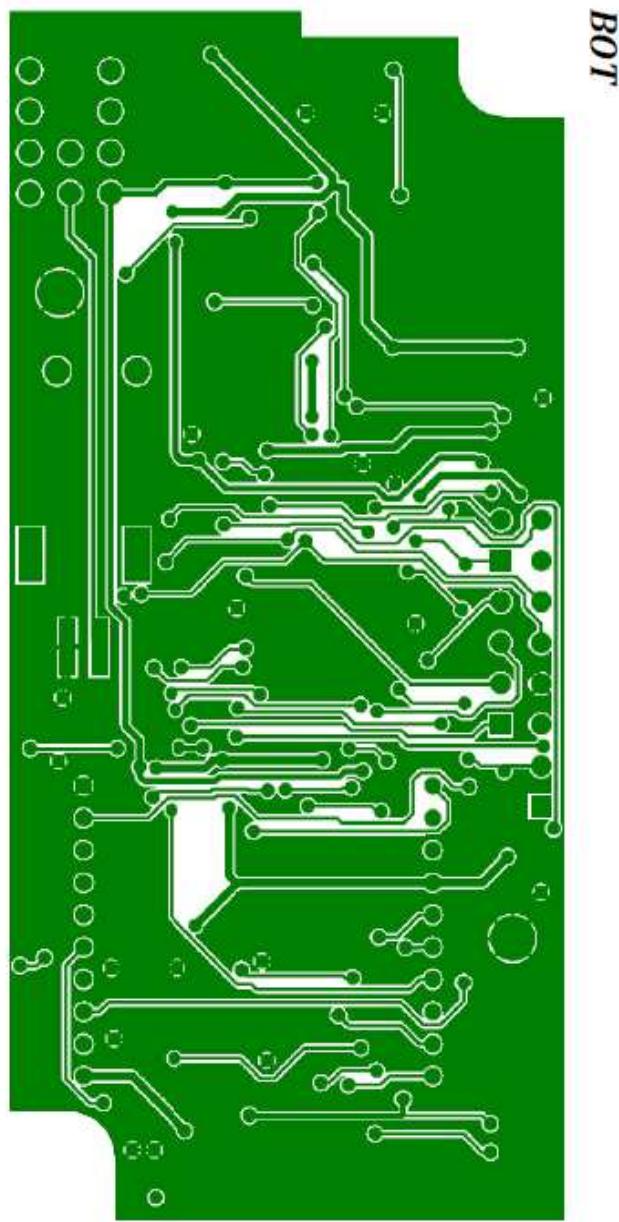


Figure B.8: Conductive traces layout: bottom.

APPENDIX C

VisIMU User Manual

This Appendix presents an user manual for the VisIMU application. VisIMU is a Java based application, therefore, the Java Virtual Machine should be already installed in the user system.

C.1 Running the program

There is no need for installation to run VisIMU. Just unzip the content of the VisIMU1.3.zip file into a folder and then run *VisIMU.exe*. Figure C.1 shows the appearance of the program when it is booting up.

After booting up the main window, showing the empty graphs on the right, will appear. (Figure C.2). To start a monitoring session go to *File→Import* and then click on *IMU Packages Import↔IMU Packages from device*, then click *Next*. (Figure C.2).

After that, you will need to fill up the connection details of the device (default values are for Memsense's Bluetooth IMU). The ports can be chosen after the port scanning process has finished (see red rectangle on figure C.4).

After choosing the port click *Finish*.

A port reader has been created in the left menu. This reader will extract the information out of the data frame and pass it to the function representing the data graphically.

To start reading data, right click on the port and then click on *Read Source*. (Figure C.5).

The data will, then, start to be gathered. Click *Run in background* to minimize the *Read window* and create a *Progress tab*. In order to be able to see the four graphs click on the *Plotter tab*. (Figures C.6 and C.7).

To stop the reading process, go to the *Progress tab* and click the stop button (red square).

Finally, to save the gathered data, go to *File→Export*, then click *Other* and select *Package to File*. (Figure C.8). Select the location where the file will be stored and enter the file name e.g *name.txt*. Three different types of writers can be used: a raw data writer, a real data writer which stores the data in physical units ($^{\circ}/s$, *g* and *Gauss*), and an adjustable writer that applies a gain specified by the user to the raw data. The data columns can be saved with or without header by selecting *true* or *false* in the *With Header* field. After choosing all the writing parameters click *Finish* and the data file will be created in the selected location. (Figure C.9).

To properly visualize the stored file, open it in a spreadsheet software such as Microsoft's *Excel*. The file is divided into columns, one column per sensor, plus an ID and a time stamp column. The appearance of the stored file is depicted in figure C.10.

The appearance and delay data being plotted can also be configured (line width and color). To do so go to *Window→Preferences→General Preferences→Plotter Preferences*. (Figure C.11).

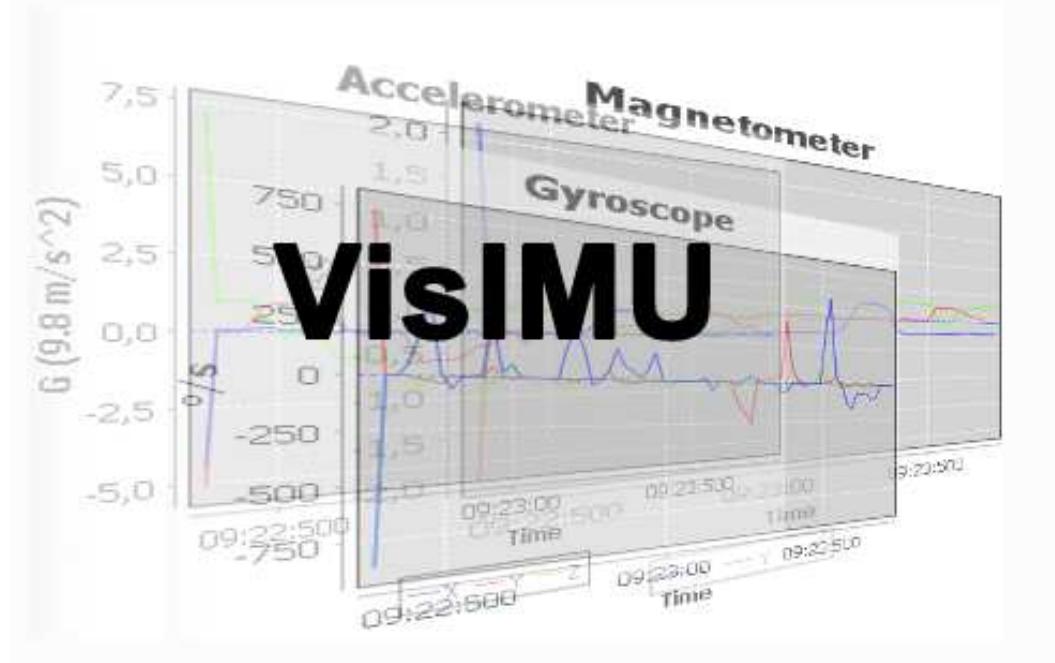


Figure C.1: VisIMU booting up.

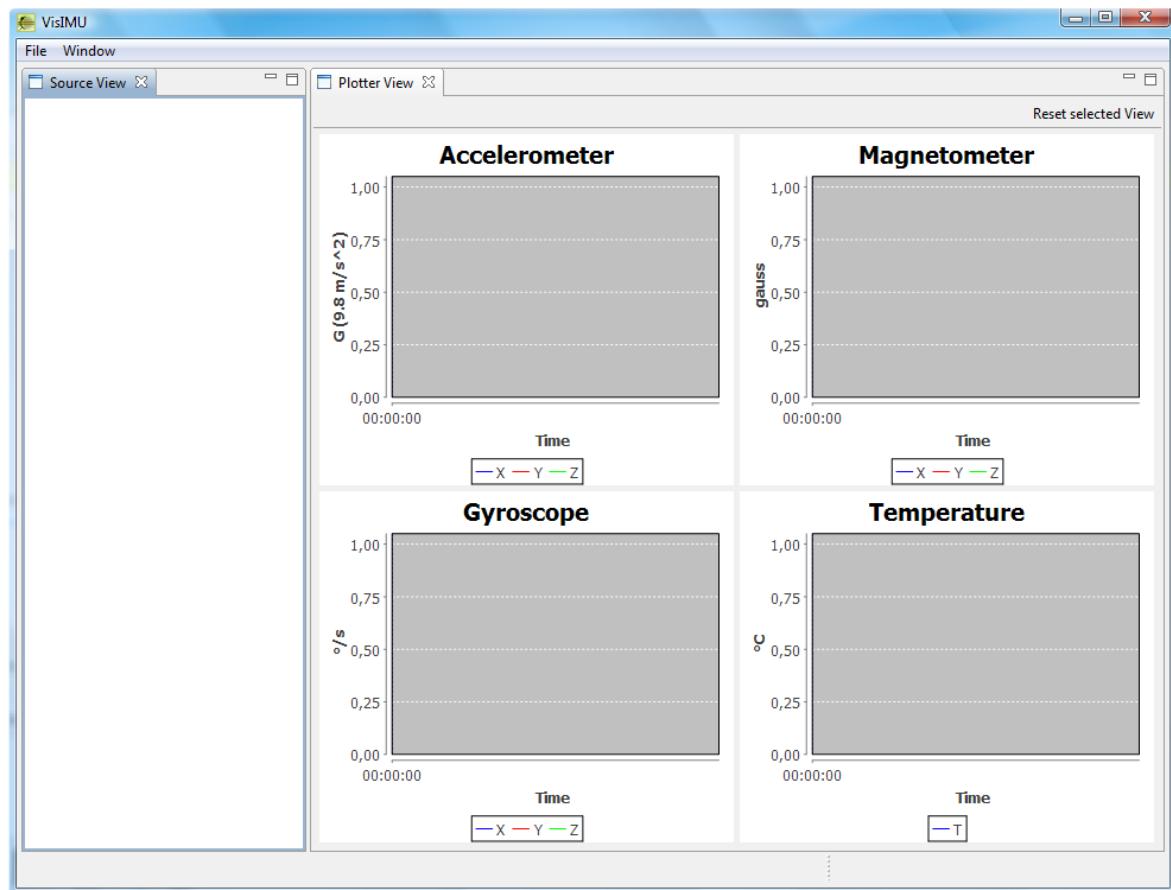


Figure C.2: Main window of VisIMU

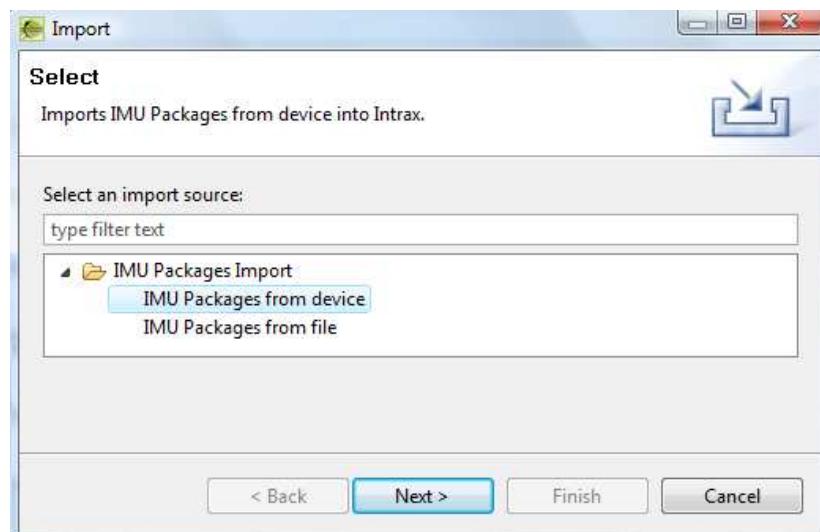


Figure C.3: Selection of packages to be read

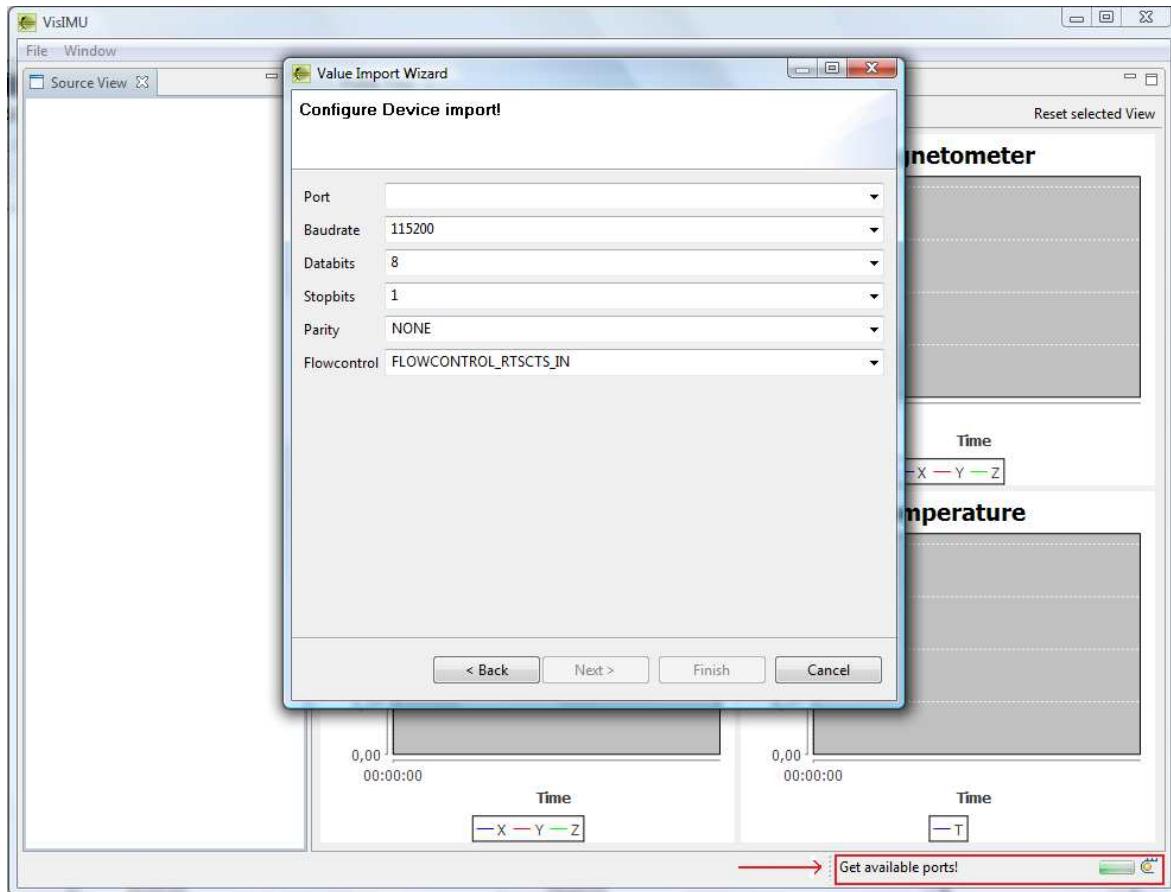


Figure C.4: Connection configuration parameters

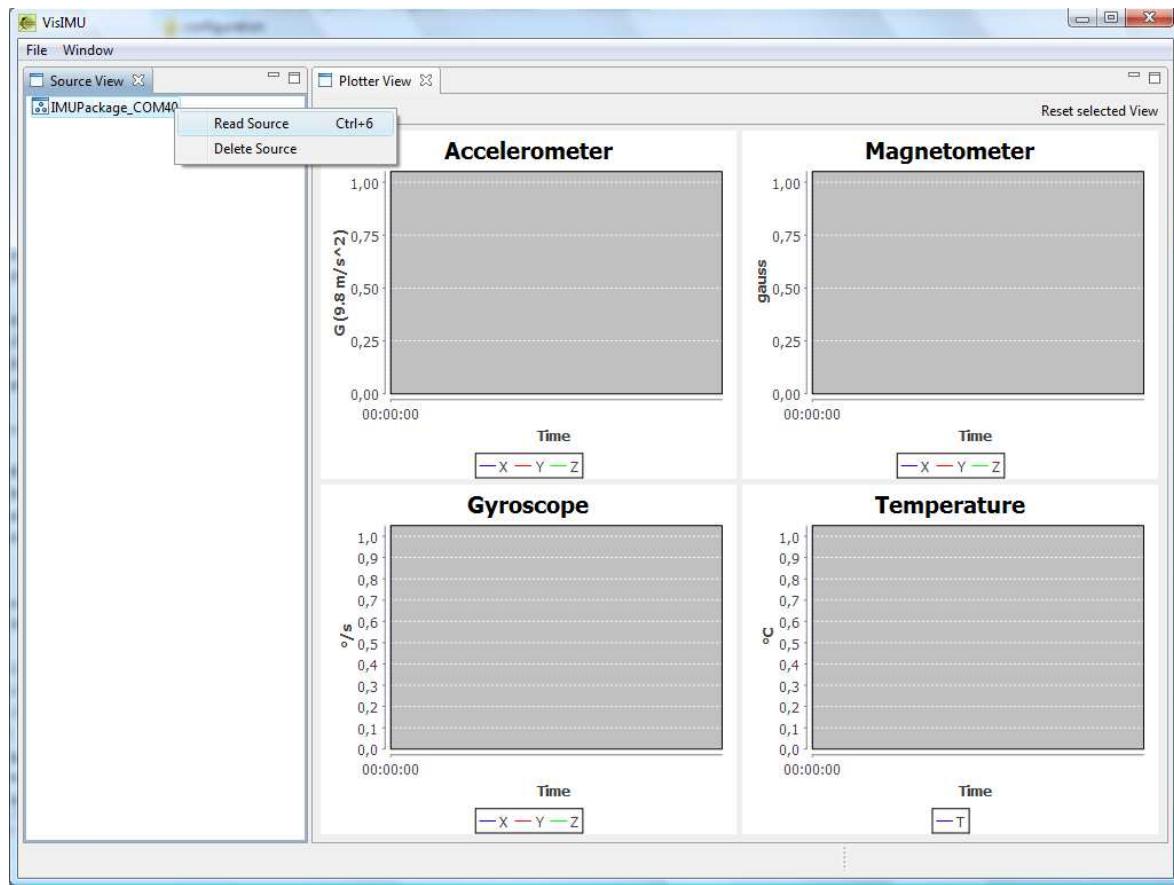


Figure C.5: Start to read data from port reader

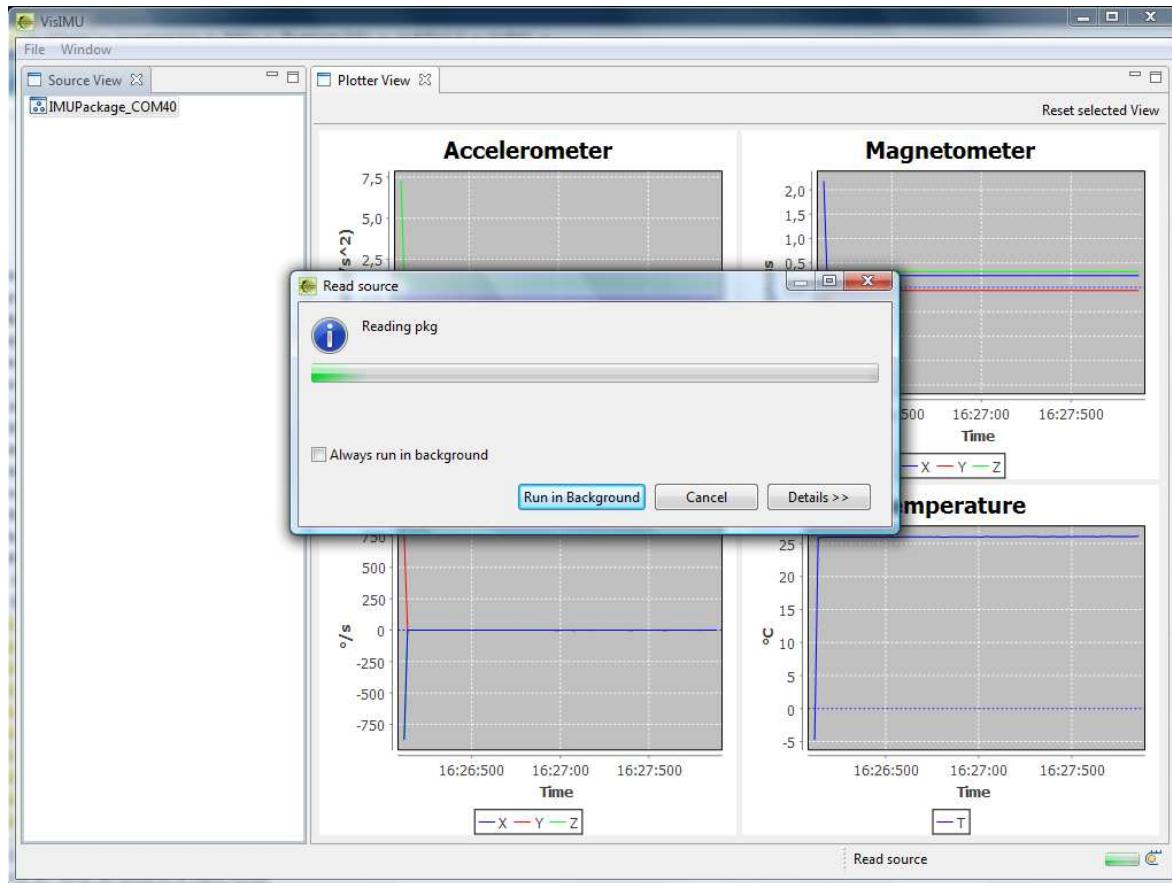


Figure C.6: Read Source dialog window

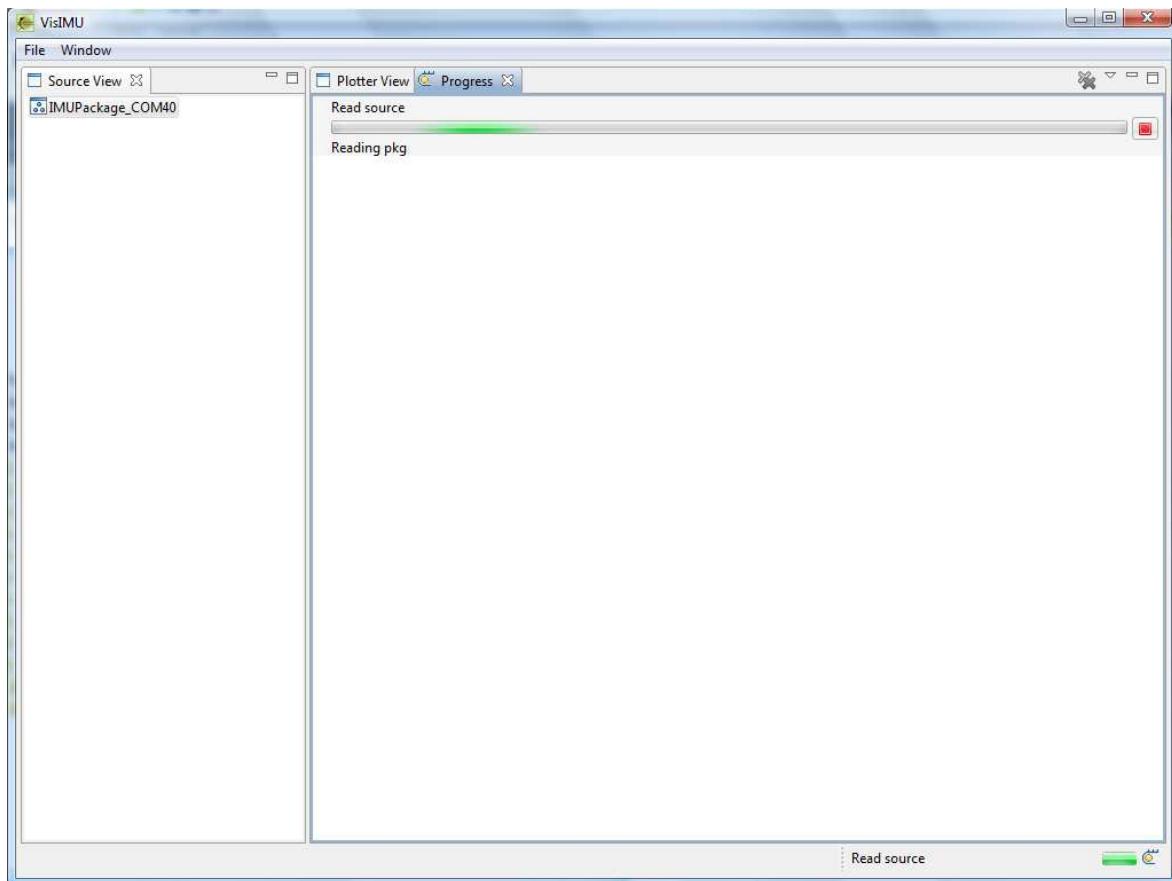


Figure C.7: Read process tab

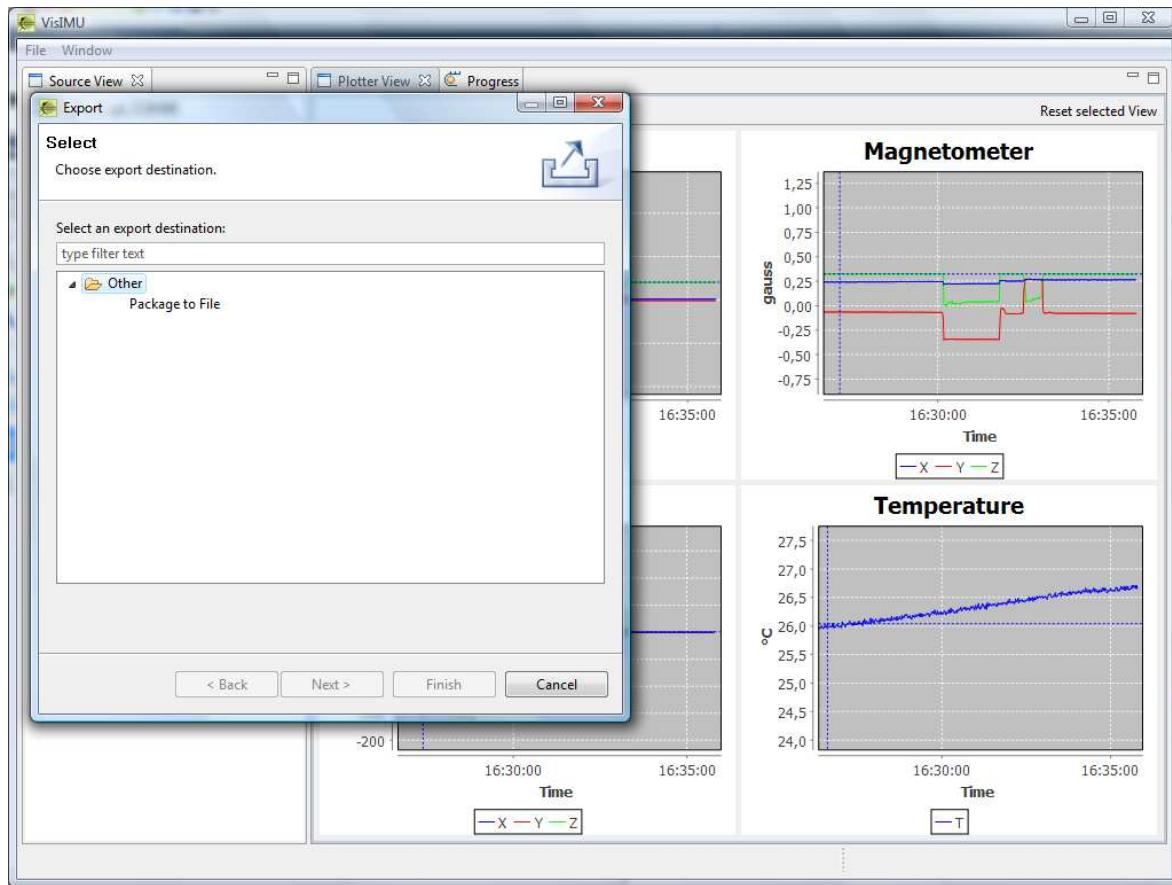


Figure C.8: Data exporting wizzard

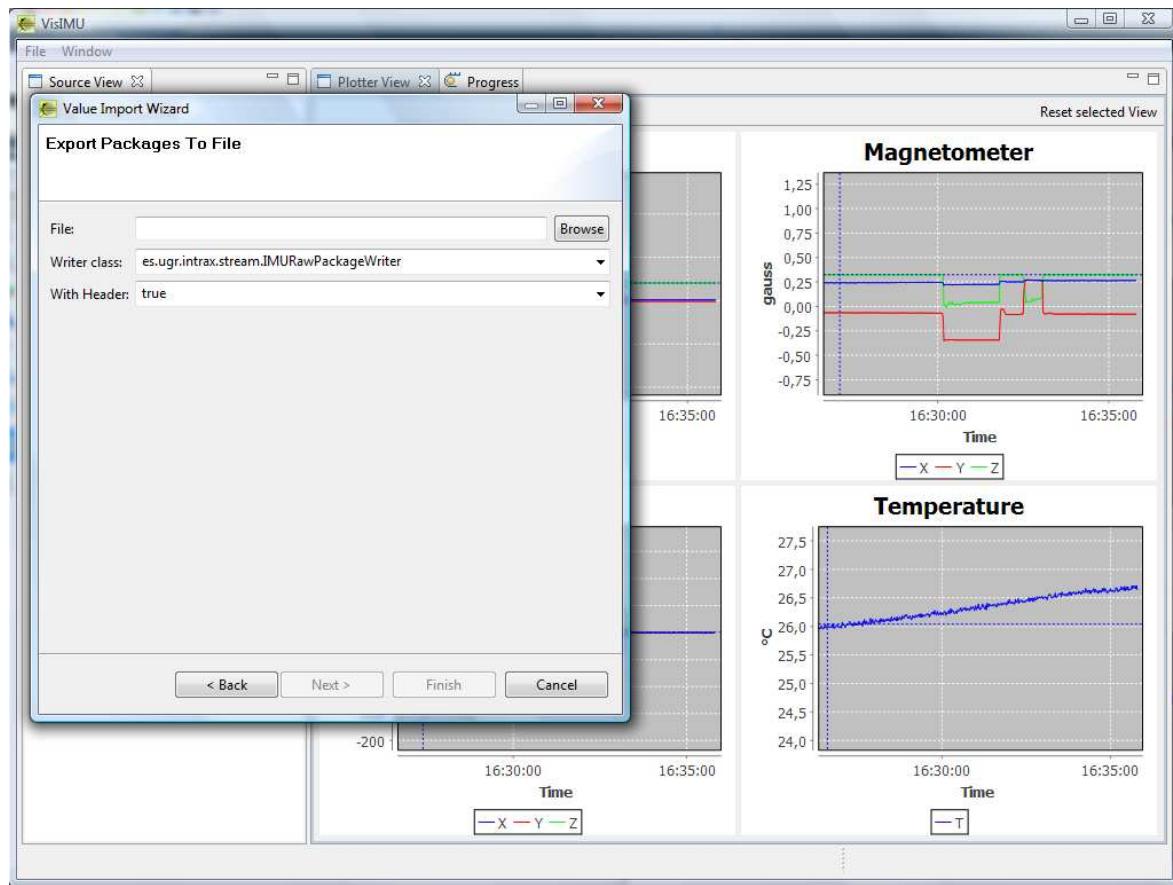


Figure C.9: Data writing parameters

	A	B	C	D	E	F	G	H	I	J	K	L
1	Id	Timer	GyroX	GyroY	GyroZ	AccelerometerX	AccelerometerY	AccelerometerZ	MagnetometerX	MagnetometerY	MagnetometerZ	Temperature
2	0	0.0	0.576765	0.27465	-0.576765	0.04531823999999	-0.02792335999999	1.01142072	0.07184135	0.119068775	0.350596225	25.0119889
3	1	0.006744478	0.137325	-0.10986	-0.082395	0.04531823999999	-0.03112767999999	1.01142072	0.07184135	0.119068775	0.351987825	25.0119889
4	2	0.013488956	0.137325	-0.137325	-0.576765	0.04531823999999	-0.02792335999999	1.00844527999999	0.07184135	0.119068775	0.351987825	25.0119889
5	3	0.607033538	-0.76902	-0.10986	-0.576765	0.04440274	-0.03158544	1.00775864	0.071928325	0.1186339	0.350422275	25.01246119
6	4	0.613747498000001	0.27465	0.302115	0.357045	0.04531823999999	-0.03112767999999	1.01142072	0.070536725	0.119068775	0.350596225	25.012152385
7	5	0.6204919760000001	0.137325	0.302115	-0.082395	0.04531823999999	-0.03112767999999	1.01142072	0.07184135	0.119068775	0.350596225	25.0119889
8	6	0.627236454	0.137325	0.27465	-0.576765	0.04531823999999	-0.03112767999999	1.00844527999999	0.07184135	0.119068775	0.351987825	25.0119889
9	7	0.633980932	-0.27465	0.302115	-0.082395	0.04600488	0.03066992	1.01210736	0.06905815	0.119416675	0.3506832	25.011825415
10	8	0.64072541	0.16479	0.302115	-0.082395	0.04600488	0.03066992	1.01210736	0.06905815	0.119416675	0.3506832	25.011825415
11	9	0.647469888	-0.302115	-0.10986	-0.5493	0.04531823999999	-0.03112767999999	1.00844527999999	0.071754375	0.11776415	0.351987825	25.012152385
12	10	0.654214366	-0.302115	-0.10986	-0.5493	0.04531823999999	-0.03112767999999	1.01142072	0.07184135	0.119068775	0.351987825	25.0119889
13	11	0.660958844	-0.302115	0.302115	-0.5493	0.04531823999999	-0.03112767999999	1.01142072	0.070536725	0.119068775	0.351987825	25.0119889
14	12	0.66770332	-0.27465	-0.10986	-0.082395	0.04531823999999	-0.03112767999999	1.00844527999999	0.070536725	0.119068775	0.351987825	25.0119889
15	13	0.6744478	0.16479	-0.10986	-0.082395	0.04600488	0.03066992	1.01210736	0.069145125	0.1207213	0.3520748	25.011825415
16	14	0.681192278	-0.27465	-0.5493	-0.082395	0.04600488	0.0274656	1.01210736	0.06905815	0.119416675	0.3506832	25.011825415
17	15	0.687936756	0.137325	0.27465	-0.576765	0.04531823999999	-0.03112767999999	1.01142072	0.07184135	0.119068775	0.350596225	25.0119889
18	16	0.57587466	-0.32958	-0.10986	0.32958	0.04371608	0.02906775999999	1.01027632	0.07462455	0.118199025	0.3517269	25.012951645
19	17	0.582619138	-0.357045	-0.137325	-0.60423	0.03982512	-0.0297544	1.00958968	0.076103125	0.11776415	0.35016135	25.01311513
20	18	0.589363616	0.082395	0.27465	-0.60423	0.04371608	-0.02906775999999	1.01027632	0.073319925	0.118199025	0.3503353	25.01278816
21	19	0.596108094	0.05493	0.27465	-0.60423	0.04600488	0.0297544	1.00958968	0.0747985	0.117851125	0.35016135	25.01326045
22	20	0.602852572	-0.32958	0.71409	-1.071135	0.04371608	-0.02906775999999	1.01027632	0.073319925	0.118199025	0.3503353	25.01278816
23	21	0.60959705	0.082395	-0.137325	-0.137325	0.04371608	-0.02906775999999	1.01027632	0.0734069	0.11950365	0.3503353	25.01278816
24	22	0.616341528	0.576765	-0.137325	-0.10986	0.04829368	0.02792335999999	1.01142072	0.070536725	0.119068775	0.350596225	25.01231587
25	23	0.623086006	0.10986	-0.10986	0.32958	0.04440272	-0.02861	1.01096295999999	0.071928325	0.1186339	0.350422275	25.012624675
26	24	0.629830484	-0.32958	0.741555	-0.137325	0.04371608	-0.02906775999999	1.01027632	0.073319925	0.118199025	0.3517269	25.01278816
27	25	0.636574962	0.082395	0.27465	-0.137325	0.04600488	0.0297544	1.00958968	0.0747985	0.117851125	0.35155295	25.01326045
28	26	0.64331944	0.082395	-0.576765	-0.137325	0.04371608	-0.02906775999999	1.01027632	0.073319925	0.118199025	0.3503353	25.012951645

Figure C.10: File including the stored sensor data

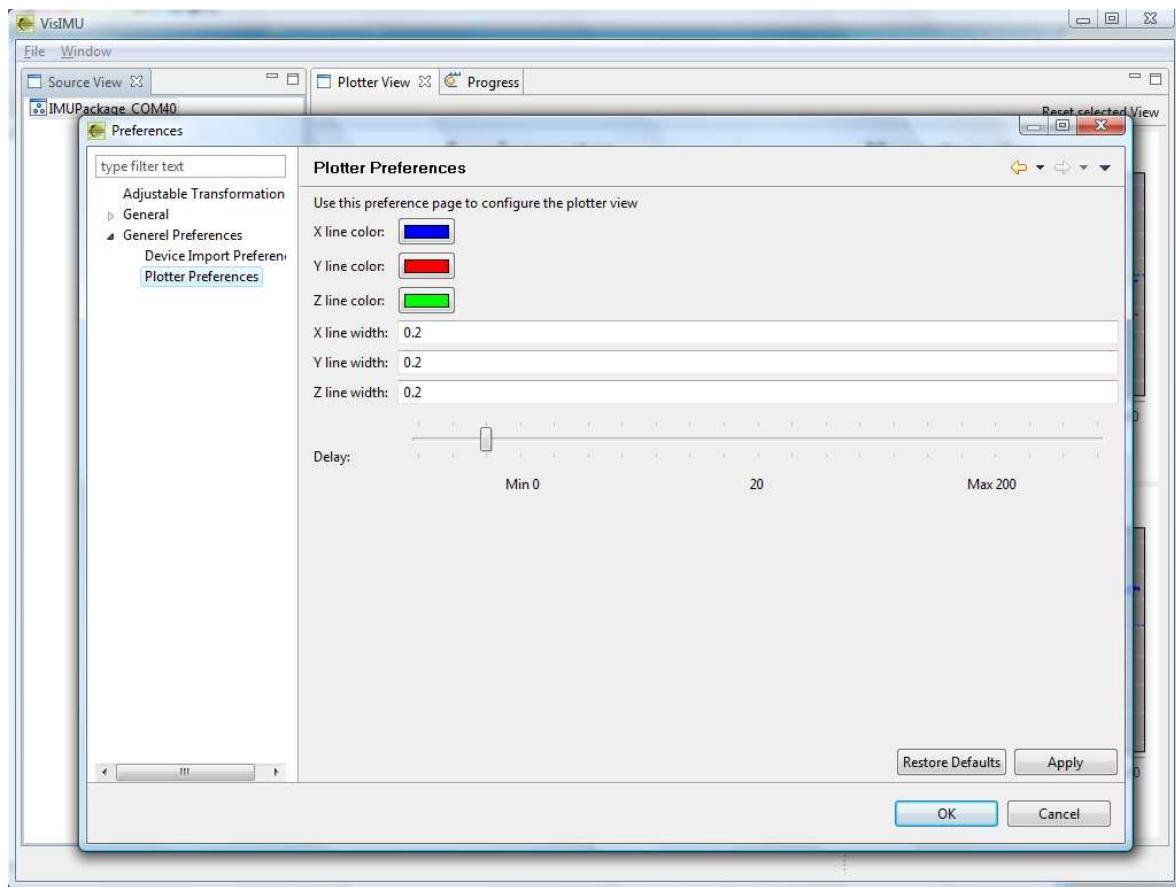


Figure C.11: Plotter preferences

Bibliography

- [1] <http://www.xsens.com>.
- [2] <http://www.memsense.com>.
- [3] <http://www.mide.com/products/volture/peh20w.php>.
- [4] <http://www.fitbit.com>.
- [5] Geomagnetic field calculator, <http://williams.best.vwh.net/magvar.htm>.
- [6] S. T. Alexander, A. L. Ghirnikar, A method for recursive least squares filtering based upon an inverse qr decomposition, *IEEE Trans. Signal Process* 41 (1993) 20–30.
- [7] E. R. Bachmann, X. Yun, D. McKinney, R. McGhee, M. J. Zyda, Design and implementation of marg sensors for 3-dof orientation measurement of rigid bodies, vol. 1, 2003.
- [8] J. K. Bekkeng, Calibration of a novel mems inertial reference unit, *Instrumentation and Measurement, IEEE Transactions on* 58 (6) (2009) 1967 –1974.
- [9] J. Borenstein, H. R. Everett, L. Feng, D. Wehe, Mobile robot positioning sensors and techniques (1997).
- [10] A. Bourke, J. O'Brien, G. Lyons, Evaluation of a threshold-based tri-axial accelerometer fall detection algorithm, *Gait and Posture* 26 (2) (2007) 194 – 199.

- [11] M. J. Caruso, Applications of magnetic sensors for low cost compass systems, 2000.
- [12] K. Cuppens, L. Lagae, B. Ceulemans, S. V. Huffel, B. Vanrumste, Detection of nocturnal frontal lobe seizures in pediatric patients by means of accelerometers: A first study, 2009.
- [13] W. Dong, K. Y. Lim, Y. K. Goh, K. D. Nguyen, I.-M. Chen, S. H. Yeo, B.-L. Duh, A low-cost motion tracker and its error analysis, 2008.
- [14] S. C. Douglas, T. H. Y. Meng, Normalized data nonlinearities for lms adaptation, *Signal Processing, IEEE Transactions on* 42 (6) (1994) 1352–1365.
- [15] A. S. Edelstein, G. A. Fischer, Minimizing $1/f$ noise in magnetic sensors using a microelectromechanical system flux concentrator, *J. Appl. Phys.* 91 (2002) 7795–7797.
- [16] A. S. Edelstein, G. A. Fischer, M. Pedersen, E. R. Nowak, S. F. Cheng, C. A. Nordman, Progress toward a thousandfold reduction in $1/f$ noise in magnetic sensors using an ac microelectromechanical system flux concentrator (invited), *Journal of Applied Physics* 99 (8) (2006) 08B317–08B317–6.
- [17] J. Elevant, Monitoring epilepsy with a wrist carried motion sensor, Ph.D. thesis.
- [18] J. Favre, Ambulatory evaluation of 3D knee joint function in patients with ACL rupture using inertial sensors, Ph.D. thesis, Lausanne (2008).
URL <http://library.epfl.ch/theses/?nr=4047>
- [19] J. Favre, F. Luthi, B. M. Jolles, O. Siegrist, B. Najafi, K. Aminian, A new ambulatory system for comparative evaluation of the three-dimensional knee kinematics, applied to anterior cruciate ligament injuries, *Knee Surgery Sports Traumatology Arthroscopy* 14 (7) (2006) 592–604.
- [20] J. Feenstra, J. Granstroma, H. Sodano, Energy harvesting through a backpack employing a mechanically amplified piezoelectric stack, *Mechanical Systems and Signal Processing* 22 (2008) 721–734.
- [21] F. Ferraris, U. Grimaldi, M. Parvis, Procedure for effortless in-field calibration of three-axis rate gyros and accelerometers, *Sensors and Materials* 42 (1995) 311–330.
- [22] D. Fontaine, D. David, Y. Caritu, Sourceless human body motion capture, *Smart Objects Conference (SOC 2003)*.
- [23] C. C. Foster, G. H. Elkaim, Extension of a two-step calibration methodology to include nonorthogonal sensor axes, *Aerospace and Electronic Systems, IEEE Transactions on* 44 (3) (2008) 1070 –1078.

- [24] J. G. F. Francis, The qr transformation, part 1, *Comput. J.* 4 (1961) 265–271.
- [25] J. G. F. Francis, The qr transformation , part 2, *Comput. J.* 4 (1962) 332–345.
- [26] D. Gebre-egziabher, G. H. Elkaim, J. D. Powell, B. W. Parkinson, A non-linear, two-step estimation algorithm for calibrating solid-state strapdown magnetometers, in: in 8th International St. Petersburg Conference on Navigation Systems, 2001.
- [27] J. W. Harding, C. G. Mackintosh, A. G. Hahn, D. A. James, Classification of aerial acrobatics in elite half-pipe snowboarding using body mounted inertial sensors, *The Engineering of Sport 7 - Vol. 2*.
- [28] S. Haykin, *Adaptive Filter Theory* (4th Edition), Prentice Hall, 2001.
- [29] A. Householder, Unitary triangularization of a nonsymmetric matrix, *J. ACM* 5 (1958) 339–342.
- [30] J. Huddleston, A. Alaiti, D. Goldvasser, D. Scarborough, A. Freiberg, H. Rubash, H. Malchau, W. Harris, D. Krebs, Ambulatory measurement of knee motion and physical activity: preliminary evaluation of a smart activity monitor, *Journal of NeuroEngineering and Rehabilitation* (2006) 3–21.
- [31] M. T. Inc., PIC24FJ64GB004 Family Data Sheet, 28/44-Pin, 16-Bit,Flash Microcontrollers with USB On-The-Go (OTG) and nanoWatt XLP Technology (2009).
- [32] J. D. Jackson, D. W. Callahan, A preliminary examination of inertial navigation using accelerometers, Ph.D. thesis (2006).
- [33] R. E. Kalman, A new approach to linear filtering and prediction problems, *Transaction of the ASME Journal of Basic Engineering* (1960) 35–45.
- [34] W. Kaplan, Advanced calculus, 5th ed., Addison-Wesley, Reading, Mass., 2002.
- [35] Q. Li, V. Naing, J. Hoffer, D. Weber, A. Kuo, J. M. Donelan., Biomechanical energy harvesting: Apparatus and method, *IEEE International Conference on Robotics and Automation*.
- [36] H. Liu, G.K.H.Pang, Accelerometer for mobile robot positioning, *Industry Applications, IEEE Transactions on* 37 (3) (2001) 812–819.
- [37] J. Lötters, J. Schipper, P. Veltink, W. Olthuis, P. Bergveld, Procedure for in-use calibration of triaxial accelerometers in medical applications, *Sensors and Actuators A: Physical* 68 (1-3) (1998) 221 – 228, *euroensors XI*.
- [38] H. Luinge, P. Veltink, C. Baten, Ambulatory measurement of arm orientation, *Journal of Biomechanics* 40 (1) (2007) 78 – 85.

- [39] P. B. Mario Cesarelli, M. Bracale, Quadriceps muscles activation in anterior knee pain during isokinetic exercise, *Medical Engineering and Physics* 21 (6-7) (1999) 469–478.
- [40] M. J. Mathie, A. C. F. Coster, B.G.Celler, N. H. Lovell, Classification of basic daily movements using a triaxial accelerometer, *Med. Biol. Eng. Comput.* 42 (2004) 670–687.
- [41] J. Morris, The value of continuous passive motion in rehabilitation following total knee replacement, *Physiotherapy* 81 (9) (1995) 557–562.
- [42] T. M. E. Nijsen, P. J. M. Cluitmans, J. B. A. M. Arends, P. A. M. Griep, Detection of subtle nocturnal motor activity from 3-d accelerometry recordings in epilepsy patients, *Biomedical Engineering, IEEE Transactions on* 54 (11) (2007) 2073 – 2081.
- [43] Y. Ohgi, Mem sensor application for the motion analysis in sports science, *ABCM Symposium Series in Mechatronics*.
- [44] A. Olivares, J. Gorriz, G. Olivares, J. Ramirez, P. Glösekoetter, A study of vibration-based energy harvesting in activities of daily living, 2010.
- [45] A. Olivares, G. Olivares, P. Glösekötter, J. M. Gorriz, J. Ramirez, A study of vibration-based energy harvesting in activities of daily living, *4th International ICST Conference on Pervasive Computing Technologies for Healthcare 2010*, Munich, Germany.
- [46] S. Patel, D. Sherrill, R. Hughes, T. Hester, N. Huggins, T. Lie-Nemeth, D. Standaert, P. Bonato, Analysis of the severity of dyskinesia in patients with parkinson’s disease via wearable sensors, 2006.
- [47] D. Roetenberg, Inertial and magnetic sensing of human motion, Ph.D. thesis, University of Twente, Enschede (May 2006).
- [48] A. Rontogiannis, S. Theodoridis, On inverse factorization adaptive least-squares algorithms, *Signal Process.* 52 (1) (1996) 35–47.
- [49] L. Sahawneh, M. A. Jarrah, Development and calibration of low cost mems imu for uav applications, 2008.
- [50] A. Salarian, H. Russmann, F. Vingerhoets, P. Burkhard, Y. Blanc, C. Deholain, K. Aminian, An ambulatory system to quantify bradykinesia and tremor in parkinson’s disease, 2003.
- [51] J. R. Sally A. Jessep, Nicola E. Walsh, M. V. Hurley, Long-term clinical benefits and costs of an integrated rehabilitation programme compared with outpatient physiotherapy for chronic knee pain, *Physiotherapy* 95 (2) (2009) 94–102.

- [52] A. SantAnna, N. Wickstrom, Developing a motion language: Gait analysis from accelerometer sensor systems, 2009.
- [53] E.-H. Shin, Estimation techniques for low-cost inertial navigation, Ph.D. thesis (2005).
- [54] E. H. Shin, N. El-Sheimy, A new calibration method for strapdown inertial navigation systems, *Vermess* 127 (2002) 1–10.
- [55] S. Singh, A traceable inertial calibration parameter estimation procedure suited for mems sensing, *RSS Euron GEMSIG*.
- [56] I. Skog, P. Händel, Calibration of a mems inertial measurement unit, in: in Proc. XVII IMEKO WORLD CONGRESS, (Rio de Janeiro, 2006).
- [57] Z. F. Syed, P. Aggarwal, C. Goodall, X. Niu, N. El-Sheimy, A new multi-position calibration method for mems inertial navigation systems, *Measurement Science and Technology* 18 (7).
- [58] Y. Tao, H. Hu, A novel sensing and data fusion system for 3-d arm motion tracking in telerehabilitation, *Instrumentation and Measurement, IEEE Transactions on* 57 (5) (2008) 1029 –1040.
- [59] J.-P. Tiesel, J. Loviscach, A mobile low-cost motion capture system based on accelerometers, in: G. Bebis, R. Boyle, B. Parvin, D. Koracin, P. Remagnino, A. Nefian, G. Meenakshisundaram, V. Pascucci, J. Zara, J. Molineros, H. Theisel, T. Malzbender (eds.), *Advances in Visual Computing*, vol. 4292 of Lecture Notes in Computer Science, Springer Berlin / Heidelberg, 2006, pp. 437–446.
- [60] D. H. Titterton, J. L. Weston, *Strapdown Inertial Navigation Technology*, 2nd ed., Institution of Engineering and Technology, New York, 2004.
- [61] P. Veltink, E. Engberink, B. Van Hilten, R. Dunnewold, C. Jacobi, Towards a new method for kinematic quantification of bradykinesia in patients with parkinson's disease using triaxial accelerometry, vol. 2, 1995.
- [62] K. J. Walchko, Low cost inertial navigation:learning to integrate noise and find your way, Ph.D. thesis, University of Florida (2002).
- [63] J. Wang, Y. Liu, W. Fan, Design and calibration for a smart inertial measurement unit for autonomous helicopters using mems sensors, 2006.
- [64] G. Welch, G. Bishop, An introduction to the kalman filter, *Notes of ACM SIGGRAPH tutorial on the Kalman filter*.
- [65] B. Widrow, J. R. Glover, J. M. McCool, J. Kaunitz, C. S. Williams, R. H. Hearn, J. R. Zeidler, E. D. Jr, R. C. Goodlin, Adaptive noise cancelling: Principles and applications, *Proceedings of the IEEE* 63 (12) (1975) 1692 – 1716.

- [66] K. Zhang, A. Gorjani, D. K. Lester, Gait change after local anesthetic of chronically arthritic knee, *J Long Term Eff Med Implants* 16 (3) (2006) 223–234.