



# **AHRS algorithms and calibration solutions to facilitate new applications using low-cost MEMS**

Sebastian O. H. Madgwick

Department of Mechanical Engineering

Supervised by

Andrew Harrison

A dissertation submitted to the University of Bristol in accordance with the  
requirements for award of the degree of Doctor of Philosophy in the Faculty of  
Engineering

March 2014



# **Declaration**

I declare that the work in this dissertation was carried out in accordance with the requirements of the University's Regulations and Code of Practice for Research Degree Programmes and that it has not been submitted for any other academic award. Except where indicated by specific reference in the text, the work is the candidate's own work. Work done in collaboration with, or with the assistance of, others, is indicated as such. Any views expressed in the dissertation are those of the author.

SIGNED: .....

DATE: .....

## Declaration

# Acknowledgments

I would like to start by thanking my supervisor Dr. Andrew Harrison. Your input has been invaluable as has your patience. Secondly, thank you to Dr. Ravi Vaidyanthan, my initial supervisor before leaving to Imperial Collage London. You offered me a position at the University of Bristol and this project would not have been possible without your ongoing support and inspiration. Thirdly, I would like to thank Prof. William Harwin. Much of the work in this project builds on ideas first explored during my time at the University of Reading, I hope we can work together again.

Over the past four years at the University, I have got to know many technicians whose help and advice has been especially useful. I would like to thank Tony Griffith and Clive Rendall for their advice concerning all things mechanical, Mark Fitzgerald for your help debugging awkward electronics, and Neil Pierce for the countless 3D printed parts.

There are many friends whom have help me out along the way. Thank you to Pete Bennett for all the laser cutting; to Dave Hygh for our the fascinating conversations about manufacturing sensors; and to Jenny Lannon for putting up with me for as long as you did. I should reserve special thank you for Tom Mitchell, our many projects together over the past 2 years have been defining to this project.

I would like to thank Fabio Varesano, creator of the “FreeIMU”. I am sad that we will not be able to speak again but I am very happy that you found my work useful and even more so that you shared it with so many people.

Finally, I would like to thank my parents. I would not be in the position I am in today without you ongoing support and love.

## Acknowledgments

---

# Abstract

Microelectromechanical System (MEMS) technology is advancing rapidly. Gyroscopes, accelerometers and magnetometers, also referred to as an Inertial Measurement Unit (IMU), has traditionally been associated with aerospace and industrial robotics but is now within every smart phone. The proliferation of these low-cost devices has facilitated countless new applications with many more still unrealised. This dissertation presents work towards this end.

A significant contribution of this work was the development of novel Attitude and Heading Reference System (AHRS) algorithms that fuse together sensor data from an IMU to provide an absolute measurement of orientation relative to the Earth. The novel work presented on non-gyro IMUs demonstrated the potential practical benefits of such kinematically redundant sensor arrays.

Low-cost MEMS can only be fully utilised if they are combined with a calibration solution to provide precise measurements with a determined accuracy. This dissertation presents a comprehensive calibration solution to the specific requirements of these sensors based on extensive characterisations investigations. The calibration solutions enable sensors costing <10 United States Dollar (USD) to achieve a static pitch/roll accuracy of <1° and a static heading accuracy of <2°. This performance is equivalent to commercial IMUs costing up to 3000 USD.

The AHRS algorithm and sensor calibration works were brought together in the development of three IMU hardware platforms. To date, >500 have been sold and the open-source associated algorithm downloaded >10,000 times. Each platform addressed a specific design need and together these facilitated a wide range of new applications; demonstrated by the numerous scientific publications that resulted from collaborative projects and user projects.

## Abstract

# **Publications arising from this research**

## **Journal papers**

1. Sebastian O.H. Madgwick; Andrew Harrison; Paul Sharkey; Ravi Vaidyanathan; William Harwin. “Measuring motion with kinematically redundant accelerometer arrays: theory, simulation and implementation”. Mechatronics, Elsevier, vol. 23, no. 5, pp. 518-529, 2013

## **Conference papers**

1. Sebastian O.H. Madgwick, Andrew J.L. Harrison, Ravi Vaidyanathan. “Estimation of IMU and MARG orientation using a gradient descent algorithm”. IEEE International Conference on Rehabilitation Robotics (ICORR), 2011
2. James Carberry, Graham Hinchly, James Buckerfield, Edward Tayler. Thomas Burton, Sebastian Madgwick, Ravi Vaidyanathan. “Parametric Design of an Active Ankle Foot Orthosis with Passive Compliance”. IEEE International Conference on Computer-Based Medical Systems
3. Thomas Mitchell, Sebastian Madgwick, Imogen Heap. “Musical Interaction with Hand Posture and Orientation: A Toolbox of Gestural Control Mechanisms”. Proceedings of the International Conference on New Interfaces for Musical Expression (NIME2012), 2012
4. Sebastian Madgwick, Thomas Mitchell. “x-OSC: A Versatile Wireless I/O Device For Creative/Music Applications”. Proceedings of Sound and Music Computing Conference (SMC), 2013
5. Thomas Mitchell, Sebastian Madgwick, Simon Rankin, Geoff Hilton, Andrew Nix, “Making the Most of Wi-Fi: Optimisations for Wireless Robust Live Music Performance”. Proceedings of the International Conference on New Interfaces for Musical Expression (NIME2014)



# Table of contents

<b>Declaration</b>	<b>i</b>
<b>Acknowledgments</b>	<b>iii</b>
<b>Abstract</b>	<b>v</b>
<b>Publications arising from this research</b>	<b>vii</b>
<b>Table of contents</b>	<b>ix</b>
<b>List of tables</b>	<b>xv</b>
<b>List of figures</b>	<b>xvii</b>
<b>Glossary</b>	<b>xxiii</b>
<b>1 Introduction</b>	<b>1</b>
1.1 Low-cost MEMS sensors and AHRS applications . . . . .	1
1.1.1 Measuring motion with IMUs . . . . .	2
1.1.2 IMUs platforms . . . . .	4
1.2 Research objectives . . . . .	5
1.3 Contribution to knowledge . . . . .	5
1.4 Thesis structure . . . . .	7
<b>2 Background</b>	<b>11</b>
2.1 Introduction . . . . .	11
2.2 MEMS gyroscopes, accelerometers and magnetometers . . . . .	11
2.2.1 Accelerometers . . . . .	13
2.2.2 Gyroscopes . . . . .	14
2.2.3 Magnetometers . . . . .	14
2.2.4 Monolithic devices . . . . .	15
2.3 Attitude and Heading Reference System algorithms . . . . .	15
2.3.1 Complementary filter AHRS algorithms . . . . .	17
2.4 Non-gyro IMUs . . . . .	18
2.5 Representation and manipulation of spatial quantities . . . . .	20
2.5.1 Rotation matrices . . . . .	20
2.5.1.1 Matrix transpose . . . . .	22

2.5.1.2	Matrix multiplications . . . . .	22
2.5.1.3	Rotation of a vector . . . . .	22
2.5.2	Quaternions . . . . .	23
2.5.2.1	Quaternion conjugate . . . . .	24
2.5.2.2	Quaternion product . . . . .	25
2.5.2.3	Quaternion rotation of a vector . . . . .	25
2.5.2.4	Converting a quaternion to a rotation matrix . . .	25
2.5.2.5	Converting a rotation matrix to quaternion . . .	26
2.5.3	Euler angles . . . . .	27
2.5.3.1	Converting a rotation matrix to Euler angles . . .	28
2.5.3.2	Converting a quaternion to Euler angles . . . .	29
<b>3</b>	<b>Initial Attitude and Heading Reference System algorithm</b>	<b>31</b>
3.1	Introduction . . . . .	31
3.1.1	Chapter summary . . . . .	32
3.2	Algorithm derivation . . . . .	32
3.2.1	Orientation from angular rate . . . . .	32
3.2.2	Orientation from a homogeneous field . . . . .	33
3.2.3	Algorithm fusion process . . . . .	38
3.2.4	Magnetic distortion compensation . . . . .	40
3.2.5	Algorithm adjustable parameter . . . . .	41
3.3	Experimental equipment . . . . .	42
3.4	Experimental results . . . . .	43
3.5	Conclusions . . . . .	46
<b>4</b>	<b>Kinematically redundant non-gyro IMUs</b>	<b>49</b>
4.1	Introduction . . . . .	49
4.2	Solving inverse kinematics with accelerometers . . . . .	50
4.2.1	Forward kinematics . . . . .	51
4.2.2	Inverse kinematics solutions . . . . .	52
4.2.2.1	Combinatorial method . . . . .	52
4.2.2.2	Pseudoinverse method . . . . .	53
4.2.3	Angular velocity estimation . . . . .	54
4.2.3.1	Stability and response to initial conditions . . .	57
4.3	Orientation estimation and angular velocity drift compensation . . .	58
4.4	2 axis pendulum kinematics . . . . .	59
4.5	Simulation studies . . . . .	61
4.6	Experimental studies . . . . .	64
4.6.1	Hardware . . . . .	65
4.6.1.1	Data acquisition equipment . . . . .	65
4.6.1.2	Accelerometer probes . . . . .	65
4.6.1.3	Test rig . . . . .	66
4.6.2	Calibration . . . . .	67
4.6.2.1	Axis gain and bias . . . . .	67
4.6.2.2	Alignment . . . . .	69
4.6.3	Results . . . . .	70
4.7	Discussion . . . . .	73

---

4.8	Conclusion . . . . .	74
<b>5</b>	<b>Characterisation of low-cost MEMS sensors</b>	<b>77</b>
5.1	Introduction . . . . .	77
5.1.1	Devices selected for characterisation . . . . .	79
5.1.2	Scope of investigations . . . . .	80
5.2	Gyroscope bias thermal response . . . . .	80
5.2.1	Thermal chamber . . . . .	81
5.2.2	Monotonic thermal response . . . . .	83
5.2.3	Thermal hysteresis . . . . .	85
5.3	Gyroscope bias acceleration sensitivity . . . . .	87
5.4	Gyroscope bias random walk . . . . .	88
5.4.1	Allan variance . . . . .	89
5.5	Gyroscope, accelerometer and magnetometer thermal response . . . . .	90
5.5.1	Three-axis thermal chamber . . . . .	92
5.5.2	Gyroscope sensitivity and bias thermal response . . . . .	93
5.5.3	Accelerometer sensitivity and bias thermal response . . . . .	95
5.5.4	Magnetometer sensitivity and bias thermal response . . . . .	97
5.6	Part variation . . . . .	99
5.6.1	Gyroscope part variation . . . . .	100
5.6.2	Accelerometer part variation . . . . .	101
5.6.3	Magnetometer . . . . .	102
5.7	Conclusions . . . . .	103
<b>6</b>	<b>Calibration of low-cost MEMS sensors for AHRS applications</b>	<b>105</b>
6.1	Introduction . . . . .	105
6.1.1	Scope of calibration . . . . .	106
6.1.2	Chapter summary . . . . .	106
6.2	Commercial IMUs chosen as benchmarks . . . . .	107
6.3	Calibration cube . . . . .	108
6.3.1	Calibration cube alignment errors . . . . .	110
6.3.2	Calibration cube protractor . . . . .	111
6.4	Gyroscope calibration . . . . .	112
6.4.1	Gyroscope bias calibration . . . . .	114
6.4.2	Gyroscope calibration dataset . . . . .	114
6.4.3	Gyroscope sensitivity calibration . . . . .	115
6.4.4	Gyroscope alignment calibration . . . . .	116
6.5	Accelerometer calibration . . . . .	116
6.5.1	Accelerometer calibration dataset . . . . .	118
6.5.2	Accelerometer sensitivity and bias calibration . . . . .	119
6.5.3	Accelerometer alignment calibration . . . . .	119
6.5.4	Calibrated accelerometer performance . . . . .	120
6.6	Magnetometer calibration . . . . .	123
6.6.1	Magnetometer calibration datasets . . . . .	124
6.6.2	Calibrated magnetometer performance . . . . .	126
6.6.3	Offset calibration . . . . .	128
6.6.3.1	Offset calibration performance . . . . .	130

6.6.4	Ellipsoid calibration . . . . .	131
6.6.4.1	Ellipsoid calibration performance . . . . .	135
6.6.5	Alignment calibration . . . . .	137
6.6.5.1	Alignment calibration performance . . . . .	139
6.6.6	Summary of magnetometer calibration solutions . . . . .	142
6.7	Simplifying sensor alignment arithmetic . . . . .	143
6.8	Conclusion . . . . .	144
<b>7</b>	<b>Revised AHRS algorithm</b>	<b>145</b>
7.1	Introduction . . . . .	145
7.1.1	Algorithm overview . . . . .	146
7.2	Complementary filter estimator . . . . .	147
7.2.1	Algorithm gain and initialisation . . . . .	148
7.3	Sensor conditioning . . . . .	149
7.3.1	Gyroscope bias compensation . . . . .	149
7.3.2	Magnetic distortion rejection . . . . .	150
7.3.3	Linear acceleration rejection . . . . .	150
7.4	'Zero-g' and global acceleration calculation . . . . .	151
7.5	Algorithm block diagram . . . . .	151
7.6	Discussion . . . . .	153
7.6.1	Algorithm gain . . . . .	153
7.6.2	Gyroscope bias compensation . . . . .	154
7.6.3	Magnetic distortion rejection . . . . .	155
7.6.4	Linear acceleration rejection . . . . .	156
7.6.5	Algorithm outputs . . . . .	156
7.6.6	Omitting the magnetometer . . . . .	157
7.7	Conclusion . . . . .	157
<b>8</b>	<b>IMU platforms and applications</b>	<b>159</b>
8.1	Introduction . . . . .	159
8.2	x-IMU: A versatile data acquisition platform . . . . .	160
8.2.1	x-IMU overview . . . . .	160
8.2.1.1	x-IMU hardware . . . . .	161
8.2.1.2	x-IMU software . . . . .	163
8.2.2	Real-time wired and wireless communication . . . . .	164
8.2.3	Autonomous data logging . . . . .	165
8.2.4	Auxiliary port . . . . .	165
8.3	x-BIMU: A low-power, wireless IMU for real-time applications . . . . .	166
8.3.1	x-BIMU overview . . . . .	167
8.3.1.1	x-BIMU hardware . . . . .	169
8.3.1.2	x-BIMU software . . . . .	170
8.3.2	Modular communication interface . . . . .	171
8.3.3	Low-power optimisation and features . . . . .	173
8.3.3.1	802.15.4 low-power optimisations . . . . .	173
8.3.3.2	Sleep timer and motion trigger wake up . . . . .	174
8.3.3.3	Measured current consumption . . . . .	174
8.3.4	Synchronisation of multiple devices . . . . .	175

8.4	x-OSC: High-performance wireless IMU and I/O interface . . . . .	176
8.4.1	x-OSC overview . . . . .	177
8.4.1.1	x-OSC hardware . . . . .	178
8.4.1.2	x-OSC browser interface . . . . .	179
8.4.2	I/O channels . . . . .	180
8.4.3	Network modes . . . . .	181
8.4.4	Practical throughput and latency . . . . .	182
8.4.4.1	Round-trip Latency . . . . .	182
8.4.4.2	Throughput . . . . .	183
8.4.4.3	Throughput-latency trade-off . . . . .	185
8.5	Collaborative applications . . . . .	186
8.5.1	The gloves . . . . .	186
8.5.1.1	Prototype gloves . . . . .	187
8.5.1.2	The performance gloves . . . . .	188
8.5.2	Gait analysis . . . . .	190
8.6	User applications . . . . .	192
8.6.1	Estimating upper limb kinematics . . . . .	192
8.6.2	Digits: Wrist-worn gloveless sensor . . . . .	193
8.6.3	MONSUN II: A small and inexpensive AUV for underwater swarms . . . . .	194
8.6.4	SMART-E: An omnidirectional AUV . . . . .	195
8.6.5	Tidal turbulence spectra from a compliant mooring . . . . .	196
8.6.6	Virtual reality based minimally invasive surgery training system . . . . .	197
8.6.7	Instrumented soccer equipment to collected ankle joint kinematics in the field . . . . .	199
8.6.8	Evaluating horse lameness using standalone ‘low-cost’ IMUs . . . . .	200
8.6.9	Telepresences through “Flobi”, an anthropomorphic robot head . . . . .	201
8.6.10	Haptic communication between dance performers . . . . .	202
8.6.11	SLAM for an autonomous car . . . . .	203
8.6.12	PERSIA: An autonomous humanoid robot to compete in RoboCup . . . . .	204
8.6.13	Low-cost structural usage monitoring of small-fleet military helicopters . . . . .	206
8.6.14	Danceroom spectroscopy . . . . .	207
8.6.15	Changibles: Shape changing constructive assembly . . . . .	207
8.7	Future work . . . . .	209
<b>9</b>	<b>Conclusions</b>	<b>211</b>
<b>References</b>		<b>213</b>
<b>A</b>	<b>Non-gyro IMU planar simplification</b>	<b>231</b>
A.1	Planar simplification . . . . .	231
A.1.1	Angular velocity estimation . . . . .	232
A.2	Optimal array configurations . . . . .	232

A.2.1	Array volume and kinematic origin . . . . .	233
<b>B</b>	<b>Crystal oscillator thermal response</b>	<b>237</b>
B.1	Introduction . . . . .	237
B.2	Crystal error model . . . . .	238
B.3	Crystal characterisation . . . . .	239
B.4	Crystal calibration . . . . .	240
B.4.1	Crystal calibration assuming typical parameters . . . . .	240
B.4.2	Crystal calibration based on error model . . . . .	241
B.4.3	Crystal calibration through third order model . . . . .	241
B.5	Conclusion . . . . .	242
<b>C</b>	<b>Effect of geographic location on AHRS heading accuracy</b>	<b>243</b>

# List of tables

3.1	Static and dynamic RMS error of Kalman-based algorithm and proposed algorithm IMU and MARC implementations . . . . .	44
5.1	Gyroscope bias error assuming an $n^{th}$ order temperature calibration model . . . . .	84
5.2	Temperature sensitivity of sensor parameters as specified in device datasheets. “-” indicates that the information is not provided. . . . .	91
5.3	Gyroscope sensitivity temperature coefficients . . . . .	95
5.4	Accelerometer sensitivity temperature coefficients . . . . .	96
5.5	Accelerometer bias temperature coefficients . . . . .	97
5.6	Magnetometer sensitivity and bias temperature coefficients . . . . .	98
6.1	Specified performance of each IMU as specified in the manufacturer’s documentation. “-” indicates that a parameter is not specified. . . . .	108
6.2	Heading error for each magnetometer calibration solution achieved for ‘typical’ and ‘worst case’ magnetic distortion scenarios . . . . .	142
8.1	Measured current consumption of x-BIMU and XBee 802.15.4 module	174
8.2	Summary of measured round-trip latency . . . . .	183
B.1	Crystal error model parameters as specified in datasheet [215] . . .	238
B.2	Crystal error model parameters for eight devices over temperature range . . . . .	239
C.1	Extreme values of $b$ and $\gamma$ selected from [30] to demonstrate the least disparity in $\theta_{error}$ for differing geographic locations . . . . .	246

List of tables

---

# List of figures

1.1	Subject coordinate system, $S$ , relative to reference coordinate system, $R$ . . . . .	3
2.1	Coordinate system $B$ with an arbitrary orientation relative to coordinate system $A$ . . . . .	21
2.2	Any orientation of coordinate system $B$ relative to coordinate system $A$ can be achieved through a rotation of angle $\theta$ around the axis ${}^A\hat{\mathbf{r}}$ . . . . .	23
2.3	The <i>aerospace sequence</i> of Euler angles describe orientation as the result of angular rotations $\phi$ , $\theta$ and $\psi$ around the $x$ , $y$ and $z$ axes respectively . . . . .	27
2.4	The Euler angle sequence describes successive rotations of around each axis to achieve the ultimate orientation of $B$ relative to $A$ . . . . .	28
3.1	Block diagram representation of the complete orientation estimation algorithm for an IMU implementation . . . . .	40
3.2	Block diagram representation of the complete orientation estimation algorithm for an MARG implementation including magnetic distortion compensation . . . . .	42
3.3	Typical results for measured and estimated angle $\theta$ (top) and error (bottom) . . . . .	43
3.4	The effect of the adjustable parameter, $\beta$ , on the performance of the proposed algorithm IMU (left) and MARG (right) implementations	45
3.5	The effect of sampling rate on the performance of the proposed algorithm IMU (left) and MARG (right) implementations . . . . .	45
4.1	Schematic describing the $i$ 'th accelerometer relative to the ridged body kinematic frame . . . . .	50
4.2	Complementary filter used for angular velocity estimation . . . . .	57
4.3	2 axis pendulum schematic . . . . .	60
4.4	Schematic of pendulum with triple-axis accelerometers mounted in helix pattern. Graphics generated by simulation software . . . . .	61
4.5	Angular acceleration error vs. sensor redundancy . . . . .	63
4.6	Angular velocity error vs. sensor redundancy . . . . .	63
4.7	Angular accelerations of pendulum measured by accelerometer array (top) and residuals (bottom) . . . . .	64

4.8	Estimated angular velocity compensated by orientation estimation algorithm (top) and residuals (bottom) . . . . .	64
4.9	Complete accelerometer probe next to a 150 mm ruler . . . . .	66
4.10	2 axis pendulum with attached accelerometer probes and annotated potentiometer axes $\theta_1$ and $\theta_2$ . . . . .	67
4.11	Validation the calibrated accelerometer gains and biases . . . . .	69
4.12	Angular accelerations of pendulum measured by accelerometer array (top) and residuals (bottom) . . . . .	71
4.13	Estimated angular velocity compensated by orientation estimation algorithm (top) and residuals (bottom) . . . . .	72
4.14	Relationship between error and sensor redundancy for linear acceleration (left), angular acceleration (middle), angular velocity (right). Whiskers indicate $1.5 \times$ interquartile range . . . . .	72
5.1	Ideal Aerosmith 2003HP three-axis thermal chamber of achieve precise velocities and orientations for temperatures between -65°C to +85°C. Image source: [115] . . . . .	78
5.2	MPU-6050 (top IC) and HMC5883L (bottom IC) mounted on the x-IMU PCB. The larger IC is $4 \times 4$ mm in size. . . . .	79
5.3	Thermal chamber internal components including the prototype x-IMU being tested . . . . .	81
5.4	Sealed thermal chamber with thermocouple meter and separate variable power supplies for fan and heater . . . . .	82
5.5	Measured gyroscope temperature during experiments for each of the eight devices tested . . . . .	83
5.6	Gyroscope bias thermal response of eight devices over a -20°C to 70°C temperature range . . . . .	83
5.7	Measured gyroscope temperature during an experiment to demonstrate thermal hysteresis . . . . .	85
5.8	Thermal response of the gyroscope bias indicating thermal hysteresis	86
5.9	Gyroscope bias deviation for six different orientations relative to gravity. Box plots show the distribution for 27 devices (whiskers indicate $1.5 \times$ interquartile range). . . . .	87
5.10	Gyroscope bias random walk over 10 hours . . . . .	88
5.11	Allan variance plot of the gyroscope bias random walk . . . . .	90
5.12	Three-axis thermal chamber components including: polystyrene box with internal thermal mass and fan, cube structure to mount four prototype x-IMUs, and SL-1200MK2 turntable . . . . .	93
5.13	Measured temperature ascent of sensors within the three-axis thermal chamber over 2.5 hours . . . . .	94
5.14	Thermal response of gyroscope sensitivity and bias from -15°C to 40°C . . . . .	94
5.15	Thermal response of accelerometer sensitivity and bias from -15°C to 40°C . . . . .	96
5.16	Thermal response of magnetometer sensitivity and bias from -15°C to 40°C . . . . .	98

5.17 Gyroscope sensitivity and bias normalised distribution . . . . .	100
5.18 Accelerometer sensitivity and bias normalised distribution . . . . .	101
5.19 Magnetometer sensitivity and bias normalised distribution . . . . .	102
 6.1 Commercial IMUs chosen to benchmark performance. From left to right: MicroStrain 3DM-GX3-25, Xsens MTw, VectorNav VN-100 and CH Robotics UM6 . . . . .	107
6.2 Calibration cube with nine prototype x-BIMUs held in alignment with the cube axes . . . . .	109
6.3 A misalignment between the calibration cube and sensor $x$ axes of angle $\gamma$ prevents the sensor axis from being exposed to the full magnitude of the reference signal $r$ . . . . .	110
6.4 Relationship between calibration cube alignment error and attenuation of reference signal . . . . .	111
6.5 Calibration cube mounted on the protractor (left) and close up of cog teeth and engraved angular increments (right) . . . . .	112
6.6 The gyroscope calibration dataset is represented by the six orientations of the calibration cube on the turntable to expose each axis to a $+\omega$ and $-\omega$ reference . . . . .	115
6.7 The accelerometer calibration dataset is represented by the six orientations of the calibration cube on a level surface to expose each axis to a $+1\text{ g}$ and $-1\text{ g}$ reference . . . . .	118
6.8 Calibrated accelerometer performance was evaluated by using the calibration cube protractor to rotate through precise inclinations . .	121
6.9 Error of each uncalibrated accelerometer's measurement of inclination for rotations around each axis. The bottom three box plots show the distribution of the error in the top three plots (whiskers indicate $1.5 \times$ interquartile range). . . . .	122
6.10 Error of each calibrated accelerometer's measurement of inclination for rotations around each axis. The bottom three box plots show the distribution of the error in the top three plots (whiskers indicate $1.5 \times$ interquartile range). . . . .	123
6.11 Magnetometer in the ‘typical’ (left) and ‘worst case’ (right) magnetic distortion arrangements. The magnetometer chip is indicated by the orange circle. . . . .	124
6.12 Example 3D plot of uncalibrated magnetometer measurements for ‘typical’ (left) and ‘worst case’ (right) magnetic distortion datasets .	125
6.13 Calibrated magnetometer performance was evaluated by using the calibration cube protractor to rotate through precise heading angles	127
6.14 Error of each uncalibrated magnetometer's measurement of heading for rotations around each axis for the ‘typical’ magnetic distortion dataset. The bottom three plots show the distribution of the error in the top three plots. . . . .	128
6.15 Example 3D plot of magnetometer measurements after offset calibration for ‘typical’ (left) and ‘worst case’ (right) magnetic distortion datasets . . . . .	130

6.16 Heading measurement error provided by the nine magnetometers for the ‘typical’ magnetic distortion scenario after offset calibration.	131
6.17 Heading measurement error provided by the nine magnetometers for the ‘worst case’ magnetic distortion scenario after offset calibration.	132
6.18 Example 3D plot of magnetometer measurements after ellipsoid calibration for ‘typical’ (left) and ‘worst case’ (right) magnetic distortion datasets . . . . .	136
6.19 Heading measurement error provided by the nine magnetometers for the ‘typical’ magnetic distortion scenario after ellipsoid calibration.	136
6.20 Heading measurement error provided by the nine magnetometers for the ‘worst case’ magnetic distortion scenario after ellipsoid calibration.	137
6.21 Example 3D plot of magnetometer measurements after alignment calibration for ‘typical’ (left) and ‘worst case’ (right) magnetic distortion datasets . . . . .	140
6.22 Heading measurement error provided by the nine magnetometers for the ‘typical’ magnetic distortion scenario after alignment calibration.	141
6.23 Heading measurement error provided by the nine magnetometers for the ‘worst case’ magnetic distortion scenario after alignment calibration. . . . .	141
7.1 Complete AHRS block diagram . . . . .	152
8.1 x-IMU board alone (left) and enclosed in its plastic housing with 1000 mAh battery (right) . . . . .	161
8.2 x-IMU top and bottom view with labelled key hardware components	162
8.3 x-IMU GUI displaying real-time data with 2D and 3D graphics . . .	164
8.4 x-BIMU with an unattached XBee 802.15.4 module . . . . .	167
8.5 x-BIMU in plastic housing with XBee 802.15.4 module and 320 mAh battery (left) and Velcro strap designed for human motion applications (right) . . . . .	168
8.6 x-BIMU top with labelled key hardware components . . . . .	169
8.7 x-BIMU Terminal displaying real-time data with 2D and 3D graphics	171
8.8 Example XBee-style communication modules compatible with the x-BIMU, for: 802.15.4 (a) (b), Bluetooth (c), ZigBee (d), Proprietary 900 MHz (e), Wi-Fi (f), Bluetooth 4.0 Low Energy (g) and wired serial (h). . . . .	172
8.9 Four 802.15.4 receivers (Digi XSticks) in a USB hub provide synchronised measurements from x-BIMUs . . . . .	175
8.10 x-OSC . . . . .	177
8.11 x-OSC top and bottom view with labelled key hardware components	178
8.12 x-OSC settings viewed on web browser . . . . .	180
8.13 Normalised distribution of measured round-trip latency . . . . .	183
8.14 Throughput of 1 to 15 x-OSCs sending to a single AP on one channel	184
8.15 Throughput of 1 to 15 x-OSCs sending to three Access Points (APs) on three non-overlapping channels . . . . .	185
8.16 SoundGrasp gloves with wrist-mounted microphone and postures with associated control function. Image source: [170] . . . . .	187

8.17 Prototype gloves system diagram (left) and being worn by musician (right) . . . . .	188
8.18 Performance glove parts . . . . .	189
8.19 Gloves being used to create and manipulate visuals projected on to a screen . . . . .	190
8.20 Measured foot position during walking (3 steps) obtained using an x-IMU . . . . .	191
8.21 Mechanical model of upper limb (a), placement of x-IMUs (b) and instrumented anthropomorphic upper limb used for evaluation (c). Image source: [187] . . . . .	193
8.22 Digits wrist-worn hardware components. Image source: [188] . . . . .	194
8.23 MONSUN II AUV is 60 cm long, has a hull diameter of 10 cm and weighs <3kg. Image source: [190] . . . . .	194
8.24 Key internal hardware complements including an x-IMU. Image source: [190] . . . . .	195
8.25 SMART-E AUV uses three actuated thrusters for omnidirectional movement and a band of RGB LEDs to provide visual feedback under water. Image source: [195] . . . . .	196
8.26 Internal electronics and communication networks between sensors, actuators and controllers. Image source: [195] . . . . .	196
8.27 Dimensional drawings of Tidal Turbulence Moorings deployed in Admiralty Inlet (left) and Chacao Channel (right). Image source: [196] . . . . .	198
8.28 Instrumented surgical tool uses x-IMU to provide orientation data. Image source: [196] . . . . .	199
8.29 Virtual reality game controlled using the instrumented surgical tool. Image source: [196] . . . . .	199
8.30 Instrumented equipment comprised an a soccer boot and shin guard; each with an x-IMU mounted internally. The equipment is shown here attached the the synchronisation trigger prior to a data acquisition session. Image source: [198] . . . . .	200
8.31 Combined motion capture and display helmet (left) and robot head (right). Image source: [202] . . . . .	202
8.32 Latest generation autonomous car shown fully assembled (left) and with shell removed (right) . . . . .	203
8.33 Screenshot showing view from camera mounted on car and the plotted location of the car as estimated by the RatSLAM algorithm	204
8.34 ‘Adult size’ robot (150 cm) robot walking towards football (Photograph courtesy of Hamidreza Kasaei) . . . . .	205
8.35 The x-IMU is being used to collect data from the Army Historic Aircraft Flight Scout AH Mk 1 to monitor structural usage . . . . .	206
8.36 An x-BIMU mounted on a dancer’s wrist allows her to control the real-time visualisations of the molecular simulation. Photographer: Paul Blakemore . . . . .	208
8.37 Changibles analysis software and resultant physical construction. Image source: [209] . . . . .	208

8.38 An individual block containing x-OSC for Wi-Fi control and 2 servos with scissor lift mechanism. Image source: [209] . . . . .	209
A.1 Schematic describing the planar sensor array characterised by parameters $p$ (origin displacement) and $l$ (sensor separation) . . . . .	233
B.1 Thermal repose of eight crystal oscillators . . . . .	239
B.2 Crystal oscillator error using the generic calibration model and typical parameters specified in the datasheet . . . . .	240
B.3 Crystal oscillator error using the generic calibration model and specific parameters evaluated empirically . . . . .	241
B.4 Crystal oscillator error using a third-order calibration model . . . . .	242
C.1 Contour interval: 2 degrees, red contours positive (down); blue negative (up); green zero line. Image source: [30] . . . . .	244
C.2 Heading error achieved for an inclination of $0^\circ$ (on the equator), $30^\circ$ , $70^\circ$ (in the UK) and $80^\circ$ and fixed values of $b = 0.5$ Gauss and $e = 0.0044$ Gauss . . . . .	246

# Glossary

<b>2D</b> Two Dimensional .....	124
<b>3D</b> Three Dimensional .....	3
<b>ADC</b> Analogue-to-Digital Converter .....	79
<b>ADV</b> Acoustic Doppler Velocimeter .....	196
<b>AHRS</b> Attitude and Heading Reference System.....	2
<b>ANN</b> Artificial Neural Network.....	16
<b>AP</b> Access Point .....	xx
<b>API</b> Application Programming Interface.....	164
<b>ASCII</b> American Standard Code for Information Interchange .....	168
<b>AUV</b> Autonomous Underwater Vehicle .....	194
<b>BFGS</b> Broyden-Fletcher-Goldfarb-Shanno .....	69
<b>CO<sub>2</sub></b> Carbon Dioxide.....	78
<b>CSV</b> Comma-Separated Values .....	163
<b>DCM</b> Direction Cosine Matrix.....	20
<b>DMC</b> Digital Magnetic Compass .....	148

## Glossary

---

<b>DMD</b> Digital Micromirror Device .....	12
<b>DMP</b> Digital Motion Processor .....	174
<b>DOF</b> Degrees of Freedom .....	8
<b>DSC</b> Digital Signal Controller .....	161
<b>ENU</b> East, North, Up .....	146
<b>FTDI</b> Future Technology Devices International .....	161
<b>GPS</b> Global Positioning System .....	3
<b>GUI</b> Graphical User Interface .....	163
<b>I/O</b> Input/Output .....	160
<b>I<sup>2</sup>C</b> Inter-Integrated Circuit .....	79
<b>IC</b> Integrated Circuit .....	12
<b>IMU</b> Inertial Measurement Unit .....	1
<b>IP</b> Internet Protocol .....	181
<b>IR</b> Infra-Red .....	2
<b>LED</b> Light-Emitting Diode .....	162
<b>LN<sub>2</sub></b> Liquid Nitrogen .....	78
<b>lsb</b> least-significant bit .....	79
<b>MAC</b> Medium Access Control .....	171

<b>MARG</b> Magnetic, Angular Rate and Gravity.....	41
<b>MDF</b> Medium-Density Fibreboard.....	109
<b>MEMS</b> Microelectromechanical System.....	1
<b>MIPS</b> Million Instructions Per Second.....	161
<b>MIS</b> Minimally Invasive Surgery.....	197
<b>OCXO</b> Oven-Controlled Crystal Oscillator .....	237
<b>OS</b> Operating System.....	183
<b>OSC</b> Open Sound Control.....	177
<b>PCB</b> Printed Circuit Board.....	79
<b>ppm</b> parts per million .....	114
<b>PWM</b> Pulse-Width Modulation .....	165
<b>RAM</b> Random-Access Memory.....	161
<b>RF</b> Radio Frequency.....	3
<b>RGB</b> Red, Green, Blue.....	187
<b>RMS</b> Root Mean Squared .....	43
<b>RPM</b> Rotations Per Minute .....	92
<b>RTC</b> Real-Time Clock.....	160
<b>SD</b> Secure Digital.....	79
<b>SLAM</b> Simultaneous Localisation And Mapping.....	203

## Glossary

---

<b>SPP</b> Serial Port Profile .....	164
<b>SRAM</b> Static Random-Access Memory .....	12
<b>TCP</b> Transmission Control Protocol .....	172
<b>TCXO</b> Temperature Compensated Crystal Oscillator .....	237
<b>TTM</b> Tidal Turbulence Moorings .....	196
<b>UART</b> Universal Asynchronous Receiver/Transmitter .....	181
<b>UAV</b> Unmanned Aerial Vehicle .....	4
<b>UDP</b> User Datagram Protocol .....	172
<b>UK</b> United Kingdom .....	35
<b>USB</b> Universal Serial Bus .....	65
<b>USD</b> United States Dollar .....	7
<b>VR</b> Virtual Reality .....	197
<b>XLP</b> eXtreme Low Power .....	170

# Chapter 1

## Introduction

### 1.1 Low-cost MEMS sensors and AHRS applications

Gyroscopes and accelerometers have traditionally been associated with the flight control systems of an aircraft or the precision instruments within industrial robotics. Today, they are ubiquitous; they are in our smart phones, cameras, hard drives and toys. Advances in Microelectromechanical System (MEMS) technology have facilitated the miniaturisation of these sensors with ever increasing performance. In the 90s, performance of inertial MEMS was increasing by a factor of five every year [1]. Today, performance can be achieved equally that of *tactical grade* to meet the demands of the military and aerospace industry [2].

An Inertial Measurement Unit (IMU) consists of both a gyroscope and an accelerometer. The gyroscope provides a measurement of angular velocity and the accelerometer provides a measurement of acceleration combined with the direction of gravity. While these quantities are useful in themselves, it is their combined use that is special. Using a sensor fusion algorithm, the sensors can be combined to

provide an absolute measurement of orientation relative to the Earth's surface. If a magnetometer is used in conjunction with the IMU, the absolute measurement of orientation may also be determined relative to magnetic north to create an Attitude and Heading Reference System (AHRS).

The proliferations of these sensors has facilitated countless new application. Smart phones and tablets typically incorporate these sensors alongside cameras and touch screens and provide an excellent example of the new applications that have been achieved, including: navigation, image stabilisation, gestural control, augmented reality and personal health/biometrics.

This project set out to demonstrate how new applications can be realised with greater precision than previously using modern low-cost MEMS gyroscopes, accelerometers and magnetometers. This was achieved through the development of publicly available software and hardware platforms that have facilitated a wide range of commercial and academic research projects exploring a diverse range of applications.

### 1.1.1 Measuring motion with IMUs

The purpose of an IMU is to quantify a subject coordinate frame relative to a reference point. This is represented in Figure 1.1 where the coordinate system  $S$ , defined by the axes  $x_s$ ,  $y_s$  and  $z_s$ , is the subject; and  $R$ , defined by the axes  $x_r$ ,  $y_r$  and  $z_r$ , is the reference coordinate system. The origin of  $S$  is distance  $d$  from  $R$  and at an arbitrary orientation. For example, the reference coordinate system may be that of the Earth and the subject may be the position of a robot or a point on a person's body.

There are several technologies that may be used for the measurement of motion. Optical systems typically use Infra-Red (IR) light in conjunction with cameras

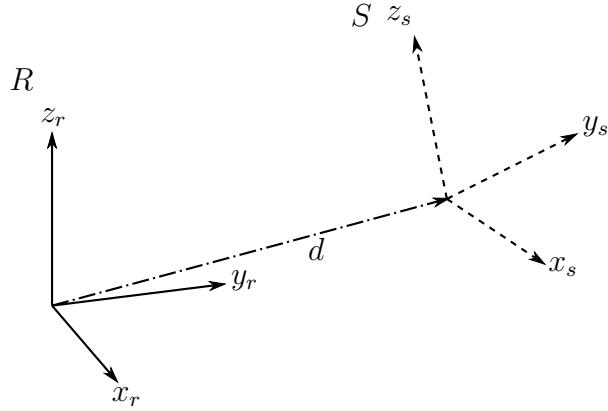


Figure 1.1: Subject coordinate system,  $S$ , relative to reference coordinate system,  $R$

to determine a Three Dimensional (3D) position; notable examples include the Vicon system, Microsoft Kinect and Leap Motion. Radio Frequency (RF) systems such as the Xsens MVN MotionGrid or Global Positioning System (GPS) rely on transmitters and receivers fixed to the subject and reference. Magnetic systems operate in a similar way to RF but use active magnetic fields; a notable example is the Razer Hydra games controller. IMU technology is exceptional in that it does not require any components external to the subject; the measurement unit is completely self-contained. Unlike other measurement systems, this means that IMU systems are not constrained to a specific operating environment and are not vulnerable to external interferences.

Whereas other measurement systems determine the 3D position and orientation of the subject, inertial systems are only able to determine the orientation directly. A gyroscope is used to measure the angular velocity around the subject axes; these measurements may be numerically integrated to determine the change in orientation from a known starting point. An accelerometer is used to measure the accelerations in the directions of the subject axes including the direction of gravity. As the direction of gravity is known relative to the reference axes, this may also be used to determine an orientation. Similarly, a magnetometer may be used to determine orientation relative to the Earth's magnetic field. Each one of

these sensors provides an indicate of direction but only through their combination in an AHRS algorithm can an absolute measurement of orientation be determined.

An AHRS algorithm provides a best estimate of orientation given the measured angular rates and observed directions of the Earth's gravitational and magnetic fields relative to the subject. An absolute measurement of orientation is of immediate use to many applications, such as a flight control system stabilising the attitude of a Unmanned Aerial Vehicle (UAV). IMUs may be used to determine relative positions of joints such as a robotic manipulator or the human body. By fusing data from all three types of sensor, an AHRS is able to determine the direction of gravity independent of other linear accelerations. This allows an AHRS to subtract gravity from accelerometer measurement to yield a measurement of acceleration in the Earth coordinate system. Integrating this would yield a measurement of velocity; integrating the velocity would yield the displacement of the IMU. The ability to measure displacement is of benefit to many applications. In practice, this can only be successful if an exponentially increasing error inherent to numerical integration can be compensated for. For example, pedestrian dead-reckoning applications achieve this through zero velocity detection [3].

### 1.1.2 IMUs platforms

Until recently, commercial IMUs were limited to expensive instruments for use in robotics and aerospace systems, and human motion capture systems. It is only with the advent of smart phones and the proliferation of low-cost MEMS sensors that these devices have been more widely available. Many low-cost IMU platforms are development boards featuring sensors and a processor, for example, the systems offered by Sparkfun, Polulu and Seeedstudio. Although such platforms are cheap, they are often of little use as they lack the essential sensor calibration and AHRS processing. Furthermore, these platforms do not

have the necessary infrastructure for application; for example, wired or wireless communication and interfacing software. Platforms that do offer a complete solution typically target human motion capture applications. There is an absence of a generic IMU platforms incorporating calibrated sensors and AHRS processing alongside a versatile communication interface.

## 1.2 Research objectives

The principal goal of this project is to demonstrate how new applications can be realised using modern low-cost MEMS gyroscopes, accelerometers and magnetometers. This can be broken down into the following objectives.

- Development of a AHRS algorithm applicable to modern low-cost MEMS devices
- Calibration of gyroscopes, accelerometers and magnetometers for use in AHRS applications
- Design and manufacture of IMU platforms that combine the calibration and AHRS algorithm solutions
- Demonstration of a broad range of applications using the IMU platforms through collaborations

## 1.3 Contribution to knowledge

The contributions made by this project are summarised as the following.

## AHRS algorithms

The initial AHRS algorithm presented in Chapter 3 was obtained through a novel derivation and offers advantages compared to existing algorithms [4] in computational efficiency and robust magnetic performance. The algorithm was published as a conference paper in 2011 [5] with the source code made available on Google Code<sup>1</sup>. The paper and its associated technical report have since totalled over 100 citations [6] and the code downloaded over 10,000 times [7].

The revised AHRS algorithm presented in Chapter 7 was developed after the work on sensor characterisation and calibration presented in Chapters 5 and 6, and simultaneously with the IMU platforms and applications presented in Chapter 8. The algorithm offers several improvements over the initial algorithm and other solutions available within the subject literature.

## Kinematically redundant non-gyro IMUs

Non-gyro IMUs comprised of only accelerometers may incorporate a redundant number of sensors with many potential advantages in application. However, few empirical studies exist for redundant accelerometer arrays; those that do are limited to only a small margin of redundancy. The work presented in Chapter 4 investigates the relationship between accuracy and sensor redundancy is investigated for arrays of up to 100 triple axis (300 accelerometer axes) accelerometers in simulation and 10 equivalent sensors (30 accelerometer axes) in a laboratory test rig. An article covering this work was published in the journal, Mechatronics, in 2013 [8].

---

<sup>1</sup><http://code.google.com/p/imumargalgorithm30042010sohm/>

### A complete calibration solution for modern low-cost IMUs

The calibration solutions presented in Chapter 6 build on the characterisation studies presented in Chapter 5 to address the specific requirements of modern low-cost gyroscopes, accelerometers and magnetometers. The solutions demonstrate how sensors costing <10 United States Dollars (USD) can achieve an accuracy equivalent to commercial IMUs priced up to 3000 USD. This opens the possibilities for new applications.

### Facilitation of novel work through IMU platforms

Three IMU platforms, presented in Chapter 8, were created to facilitate new applications; the x-IMU, x-BIMU and x-OSC. Each device fulfils a different design specification. x-OSC addresses shortcomings of existing commercial platforms and was originally presented in a 2013 conference paper [9]; a forthcoming 2014 conference paper focusses on its specific performance abilities [10]. The three IMU platforms facilitated a number of collaborative investigations including a novel gestural control system for music which was presented in a 2012 conference paper [11]; and a gait tracking algorithm which was demonstrated as an evaluation tool for an active ankle orthosis in a 2011 conference paper [12]. Chapter 8 also presents selected example user applications that have utilised the platforms developed. The research projects were selected to demonstrate the impact of the platforms in peer-reviewed publications.

## 1.4 Thesis structure

This section details how this dissertation is organised along with a synopsis of each chapter.

## **Chapter 2: Background**

This chapter summaries the relevant subject backgrounds; including a review of the subject literature. The chapter includes: a description of modern MEMS sensor technology; a literature review of AHRS algorithms with a particular focus on complementary filter solutions; a literature review of non-gyro IMU works; and an introduction to the representation and manipulation of spatial quantities.

## **Chapter 3: Initial Attitude and Heading Reference System algorithm**

Some of the first work conducted in this research project was the development of an AHRS algorithm. Although the algorithm would be significantly revised over the proceeding years (see Chapter 7), this initial algorithm offered a novel solution that would become widely adopted within the academic and commercial projects. The algorithm was intended to provide an alternative solution to computationally expensive Kalman-based solutions and explore an alternative approach to the complementary filter solutions available. The algorithm is benchmarked against that of a Kalman-based algorithm and demonstrated equivalent performance.

## **Chapter 4: Kinematically redundant non-gyro IMUs**

This chapter presents two schemes to measure the linear and angular kinematics of a rigid body using a kinematically redundant array of triple-axis accelerometers. A novel angular velocity estimation algorithm is proposed and evaluated that can compensate for angular velocity errors using measurements of the direction of gravity. Analysis and discussion of optimal sensor array characteristics are provided. A damped 2 axis pendulum was used to excite all 6 Degrees of Freedom (DOF) of the a suspended accelerometer array through determined complex motion and is the basis of both simulation and experimental studies. The relationship

between accuracy and sensor redundancy is investigated for arrays of up to 100 triple axis (300 accelerometer axes) accelerometers in simulation and 10 equivalent sensors (30 accelerometer axes) in the laboratory test rig. The work also reports on the sensor calibration techniques and hardware implementation.

## **Chapter 5: Characterisation of low-cost MEMS sensors**

This chapter presents the methods and findings of the sensor characterisation studies that form the basis of the calibration solutions presented in Chapter 6. Several different gyroscopes, accelerometers and magnetometers from a range of manufacturers were used throughout this project. This chapter presents the characterisation of the two devices that would ultimately be incorporated into the IMU platforms described in Chapter 8.

## **Chapter 6: Calibration of low-cost MEMS sensors for AHRS applications**

This chapter builds on the findings in Chapter 5 to develop calibration solutions applicable to low-cost MEMS gyroscopes, accelerometers and magnetometers. The calibrated performance was benchmarked against the performance of commercial IMUs to demonstrate equivalent accuracy.

## **Chapter 7: Revised AHRS algorithm**

The original AHRS algorithm presented in Chapter 3 represents some of the first work of this project. The subsequent characterisation studies presented in Chapter 5 and calibration solutions in Chapter 6 demonstrated how modern low-cost MEMS sensors can achieve equivalent performance of such commercial IMUs. The revised AHRS algorithm presented in this chapter was designed to operate with

these sensors based on the findings of Chapters 5 and 6. The revised algorithm addresses the shortcomings of contemporary complementary filter solutions while introducing several new features developed in conjunction with the IMU platforms and applications presented in Chapter 8.

## **Chapter 8: IMU platforms and applications**

The chapter describes three IMU platforms; the x-IMU, the x-BIMU and the x-OSC. Each platform fulfils a different design specification to facilitate a wide range of applications. Two collaborative application utilising these platforms are presented; a novel gestural music control system and a gait tracking algorithm. This chapter also presents selected example user applications that have utilised the platforms. The research projects chosen to appear in this chapter were selected to demonstrate the diversity of user applications and the impact of the platforms in peer-review publications.

## **Chapter 9: Conclusions**

This final chapter summarises the achievements of the project, with reference to the original research objectives. A brief description of future work is also provided.

# **Chapter 2**

## **Background**

### **2.1 Introduction**

This chapter provides an overview of the background subjects central to the project including a review of the subject literature. The following sections describe modern MEMS sensors and AHRS algorithms that are used in conjunctions with these sensors. A survey of non-gyro IMU work is also presented. The chapter concludes with an overview of the mathematics associated with the representation and manipulation of spatial quantities which will be fundamental throughout this dissertation.

### **2.2 MEMS gyroscopes, accelerometers and magnetometers**

MEMS is a term used to describe a wide range of miniaturised devices combining mechanical structures with electrical systems on a micrometer scale ( $1 \mu\text{m} = 10^{-6} \text{ m}$ ) scale. The devices are typically fabricated from silicon using

photolithography techniques derived from the Integrated Circuit (IC) industry to create moving structures such as levers, masses and actuators on the substrate alongside conventional digital and analogue ICs. Arguably the first MEMS sensors were air pressure sensors developed in 1970s that would sense a difference in pressure through the deflection of a diaphragm [13]. Perhaps the world's most complex [13] MEMS structure is the Digital Micromirror Device (DMD) created by Texas Instruments which incorporates up to two million actuated  $16 \mu\text{m}^2$  aluminium micro-mirrors [14]. The mirrors are fabricated over an array of Static Random-Access Memory (SRAM) cells so that each pixel may be addressed as a bit in digital memory. MEMS technology has facilitated the miniaturisation of a wide range of technologies but it is inertial sensors, gyroscopes and accelerometers, that have seen the greatest surge in development.

The inertial MEMS market has traditionally been dominated by the automotive sector with almost 80% of applications being related to automotive safety features in 2005 [15]. Today it is consumer electronics that drives the market [2] with diverse applications ranging from hard-disk protection to complete autopilot systems within flying toys. The Nintendo Wii games console, released in 2006, is a notable example which attracted broad public interest in inertial MEMS and provided researchers with a platform to innovate new applications [16, 17, 18]. The demands of consumer electronics have focussed development on low-cost and low-power devices. The average price of an accelerometer used in smart phones was 1.7 USD in 2006 and is predicted to be 0.2 USD in 2017 [2]. In 2012, the industries lowest power accelerometer could provide measurements at 100 Hz while drawing only  $2 \mu\text{A}$  [19]. Smart phones and tablets typically incorporate these sensors alongside cameras and touch screens and provide an excellent example of the new applications that may be achieved, including: navigation, imagine stabilisation, gestural control, augmented reality and personal health/biometrics.

### 2.2.1 Accelerometers

An accelerometer provides a measurement of acceleration; for example, in units of g or  $\text{ms}^{-2}$ . Accelerometers are also sensitive to gravity. The principle of operation is that a deflection of a mass is measured through a change in an electrical characteristic. When accelerating, the inertia of the mass exerts a force according to Newton's second law of motion. The required mechanical structure within the MEMS is recognisable as the mass and spring components with fingers on the silicon substrate that form a variable capacitor array.

Accelerometers can operate open-loop or closed-loop. Open-loop sensor performance is limited to the characteristics of the mechanical structure but require simpler interface circuitry and is inherently stable for a bandwidth below their resonance [1]. Closed-loop sensors use feedback to maintain the proof mass in a null position and can achieve an extended dynamic range, increased linearity, flatten frequency response, and improved cross-axis rejection [20].

A three-axis accelerometer requires that the proof mass is able to translate vertically. This requires a more complex mechanical structure and as a result, the  $z$  performance on triple-axis accelerometer is often worse than that of  $x$  and  $y$ . Six-axis accelerometers have been developed and are able to measure rotational accelerations [15]. However, such devices are not yet available to commercial markets. The demands of the consumer electronics industry has driven developments towards low-power and low-cost. However, there have been advances in recent years to develop high-performance *tactical* and *navigational* grade MEMS accelerometers [21, 22, 23, 24] which are predicted to replace conventional technologies in the coming years [2].

### 2.2.2 Gyroscopes

A gyroscope provides a measurement of angular velocity; for example, in units of degrees per second or radians per second. The most common MEMS gyroscopes are Coriolis vibratory gyroscopes as they offer the best performance to price ratio [2]. These devices work by vibrating a mass, when rotated, the Coriolis effect that causes a secondary vibration in a direction orthogonal to the original vibration. This active vibration means that gyroscopes will require more power than accelerometers. Development of alternative technologies attempt to address this issue [25]. Gyroscopes are particularly sensitive to temperature variations and this has been the focus of several works [26, 27, 28]. Advances in recent years have shown that *Tactical grade* performance achievable with MEMS [26] though such devices typically cost 2000 USD to 4000 USD per axis [2].

### 2.2.3 Magnetometers

Triple-axis magnetometers provide a measurement of angular velocity; for example, in units of Gauss or  $\mu\text{T}$  (micro Tesla). They typically use Anisotropic Magnetoresistive (AMR) technology [29]. Magnetic-based heading measurement imposes two fundamental limitations: (1) measurements can be corrupted by local magnetic interfaces; and (2) variations in the Earth's magnetic field [30] will mean measurements of magnetic north may be significantly different from geodetic north. A gyrocompass provides a measurement of geodetic north based on the rotation of the Earth and so solves this problem. However, this requires extremely high-performance and is typically only achieved with ring laser gyroscopes or fibre optic gyroscopes. Achieving this with MEMS is a hot topic and significant advances have been made in recent years [31, 32, 33, 34]

### 2.2.4 Monolithic devices

Some of the most recent developments in MEMS sensors has seen the appearance of monolithic devices that incorporate multiple sensors into a single package. What are commonly referred as ‘9DOF’ sensors incorporate a gyroscope, accelerometer and magnetometer. Development is primarily driven by the low-cost demands of consumer electronics; the budget for 9DOF sensors is less than 2.5 dollars in 2012 [2]. Several companies are also developing ‘10DOF’ devices that also incorporate a barometer [2] though none have reached the market yet. The latest generation of monolithic devices incorporate on-board processes to implement data processing tasks such as calibration and AHRS data fusion.

## 2.3 Attitude and Heading Reference System algorithms

An AHRS algorithm is a sensor fusion algorithm that combines the data from multiple sensors to yield a single measurement of orientation. AHRSs may incorporate a variety of technologies, for example GPS and air speed of a UAV, but in the context of this research project an AHRS is considered to be comprised of a gyroscope, accelerometer and magnetometer. The accurate measurement of orientation plays a critical role in a range of fields including: aerospace [35], robotics [36, 37], navigation [38, 39], human motion analysis [40, 41] and machine interaction [42].

The sensor fusion algorithm operates on the principle that each of the sensors is able to provide some information about the orientation but not able to provide a complete picture. The gyroscopes is the most important sensor within an AHRS. It provides a measurement of angular velocities which may be integrated over

time to yield a measurement of orientation relative to the initial orientation. If the initial orientation of the IMU is known and the gyroscope free from errors then this would be sufficient for a complete AHRS and there would be no need to incorporate other types of sensors. However, in practice, the integration of gyroscope measurements errors will mean that the measurement orientation will drift over time and is necessary to compensate for this drift using an alternative measurement of orientation. An accelerometer is able to provide a measurement of gravity relative the IMU though the measurement will be corrupted by motion. Similarly, a magnetometer provides a measurement of magnetic North but may become unreliable in the presence of magnetic distortions local to the environment. The goal of the AHRS algorithm is therefore to combine the information from all three sensors to yield measurement of orientation relative to gravity and magnetic north.

Most AHRS algorithms are either based on a Kalman filter or complementary filter though there are exceptions that have demonstrated fuzzy logic [37], Artificial Neural Network (ANN) [43, 44] and non-linear observer solutions [45]. The Kalman filter [46, 47] has become the accepted basis for the majority of AHRS algorithms and commercial IMUs; Xsens [48], MicroStrain [49], Intersense [50], VectorNav [51], CH Robotics [52] and YEI [53] all produce systems founded on its use. The widespread use of Kalman-based solutions is a testament to their effectiveness, however, they have a number of disadvantages. They can be complicated to implement which is reflected by the numerous solutions available in the subject literature; see [4] for a survey. The non-linearity and large state matrices inherent to the application present a challenge in tuning the algorithm and impose a computational load that may be inappropriate for low-power or high-bandwidth applications. These shortcomings have been cited as the motivation for the development of complementary filter alternatives [54, 55, 56].

### 2.3.1 Complementary filter AHRS algorithms

A complementary filter fuses a low bandwidth measurement of orientation derived from the accelerometer and magnetometer with the high bandwidth measurement of angular rate provided by the gyroscope based on a first-order kinematic model. Mahony *et al.* provide a review of classical complementary filtering in the appendix of [55]. The classical solution limited initial implementations to one dimensional states [57, 58]. However, subsequent works have demonstrated 3D solutions utilising Euler angles [35, 56], rotation matrices [55] and quaternions [54, 55]. Quaternion implementations are recognised as having an advantage in simplicity [59, 54] while avoiding the problematic singularity of an Euler angles representation [60].

Although the algorithm may be derived as the sum of two filters [57, 61], most authors express the complementary filter as a feedback loop [54, 35, 58, 55, 62, 56]. In this form, different works may be characterised by their computation of the feedback error. For example: as the difference between angular quantities [58, 35, 56], as the cross-product of spatial quantities [62], through rotation matrix operations [55], or through Gauss-Newton iterations [54]. The feedback gain is typically quantified as the crossover frequency [57, 54, 55, 35, 56]. This may provide a convenient means of tuning the gain if the frequency characteristics of an application are known, as in [57, 61, 56]. Most implementations will use a fixed gain but a variable gain may enable greater accuracy if unreliable sensor data can be detected [56]. The incorporation of integral feedback provides a means to compensate for a gyroscope bias [62, 35]. However, this may be an undesirable solution as it detracts from the simplicity of the complementary filter and risks worse performance through poorly tuned second-order dynamics.

Complementary filters that use only a gyroscope and accelerometer [35, 61, 62] are unable to provide an absolute measurement of heading. Some implementations

incorporate open-loop magnetometer measurements [56] though this fails to take advantage of sensor fusion. Solutions that incorporate the magnetometer within within the feedback loop [63, 64] are able to provide an absolute measurement of heading derived from both the gyroscope and magnetometer. However, these solutions require a fixed inclination of the magnetic field in the operating environment to be predefined, this is a significant limitation. Although most commercial system use Kalman-based algorithms, some modern products also offer complementary filter modes to achieve greater update speeds [53].

## 2.4 Non-gyro IMUs

Conventional IMUs consist of an accelerometer to measure accelerations and a gyroscope to measure angular velocities. However, an accelerometer is also sensitive to centripetal, Coriolis and tangential accelerations associated with rotational motion. It is therefore possible for an array of accelerometers be used in conjunction with an inverse kinematic solution to directly measure rotational kinematics and potentially eliminate the need for a gyroscope. There are a number of motivations for the exclusion of gyroscopes. A sensor array consisting only of accelerometers has the potential to incorporate redundancy which may be exploited for robustness, increased accuracy and fault detection. Redundancy of inertial sensor arrays is already common in safety critical applications such as avionics [65]. The ability to measure angular states using an accelerometer array is also of benefit to existing applications that already feature such an array since extra or redundant measurements may be obtained at no extra cost. For example, automotive applications [66, 67] incorporate more than 15 axes of inertial measurement [68].

The use of linear accelerometers to measure rotational kinematics has been a

subject of research since the 1960s [69, 70]. For planar motion, the inverse kinematic solution is trivial [71] and has few practical applications; for example, the motion of a human knee [72]. Most practical applications require motion in all six DOF to be accounted for. The so-called ‘cube configuration’ uses six single axis accelerometers each aligned along the diagonal of a different face of a cube. This configuration has been the focus of several studies [73, 74, 75], including an implementation in hardware [76]. Analytical [77] and empirical [78] studies have shown the six sensor configuration to be intrinsically unstable and limited in application of over short time intervals. Arrays of nine accelerometers provide an alternative, stable solution [79, 80, 81].

The arrays of six and nine accelerometer axis use specific configurations of multiple single-axis accelerometers. Modern MEMS accelerometers are widely available in triple-axis packages containing three mutually orthogonal accelerometer axes. The theoretical minimum number of triple-axis accelerometers required to measure motion in six DOF is four (incorporating a total of 12 linear accelerometers axis). Schemes using triple-axis accelerometers [67, 82, 83, 84, 85] do not require specific geometric configurations, only that the position of a sensor within an array is known. As each sensor is able to provide a vector measurement of acceleration in three dimensions, the physical orientation of the sensor is irrelevant provided that the orientation is known. The theoretical irrelevance of each sensor’s position and orientation within the sensor array gives triple-axis accelerometer arrays a clear practical advantage over single-axis accelerometer arrays.

Accelerometer arrays designed to incorporate a number of redundant sensors exploit the invariance of an inverse kinematic solution to the position of any individual sensor and such schemes typically use a matrix pseudoinverse [86, 87] to resolve the inverse kinematics. Few studies have addressed the relationship between the performance and redundancy of the system; most studies are numerical and limited to sensor arrays of no more than 24 sensitive axes [88]

[87].

Due to past limitations of accelerometer technology, research into accelerometer sensors arrays has been largely limited to numerical and theoretical work. It has only been with the relatively recent developments in MEMS that empirical studies have became feasible [76, 67, 89, 83]. Few empirical studies exist for redundant accelerometer arrays and such studies have been limited to a number of sensors that yield a small margin of redundancy [90] [91]. Work presented in Chapter 4 investigates highly redundant systems through both simulation and empirical studies.

## 2.5 Representation and manipulation of spatial quantities

The representation of orientation and the manipulation of spatial quantities are fundamental to this dissertation. This section provides a brief introduction to the notation and operations of rotation matrix, quaternion and Euler angle representations. An implementation of the methods presented in this section is available as an open-source<sup>1</sup> MATLAB library. The convention used throughout this dissertation is that matrices and vectors are expressed in bold font with upper-case indicating a matrix and the hat accent denoting a normalised vector of unit magnitude. For example,  $\mathbf{M}$  and  $\hat{\mathbf{v}}$  would represent a matrix and normalised vector respectively.

---

<sup>1</sup><https://github.com/xioTechnologies/Quaternion-MATLAB-Library>

### 2.5.1 Rotation matrices

A rotation matrix, sometimes called a Direction Cosine Matrix (DCM), is a  $3 \times 3$  matrix that describes the orientation of a subject in Euclidean space relative to a reference coordinate system. The three columns of the matrix define the directions of the mutually orthogonal  $x$ ,  $y$  and  $z$  axes of the subject coordinate system as unit vectors within the reference coordinate system. For example, Figure 2.1 shows an arbitrary orientation of a subject coordinate system,  $B$ , relative to the reference coordinate system,  $A$ . The principle axes of the coordinate systems  $A$  and  $B$  are defined by the mutually orthogonal units vectors  $\hat{x}_A$ ,  $\hat{y}_A$  and  $\hat{z}_A$ ; and  $\hat{x}_B$ ,  $\hat{y}_B$  and  $\hat{z}_B$  respectively.

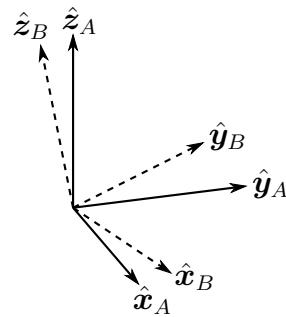


Figure 2.1: Coordinate system  $B$  with an arbitrary orientation relative to coordinate system  $A$

The rotation matrix describing the orientation of  $B$  relative to  $A$  is constructed from the available vector quantities as shown in Equation 2.1. A notation system adopted from Craig [92] is used to denote the coordinate system that a quantity is in reference to as a leading superscript, and the coordinate system the quantity describes with a leading subscript.

$${}^A_B \mathbf{R} = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} = \begin{bmatrix} {}^A \hat{\mathbf{x}}_B & {}^A \hat{\mathbf{y}}_B & {}^A \hat{\mathbf{z}}_B \end{bmatrix} \quad (2.1)$$

If  $A$  and  $B$  are aligned then the rotation matrix would be an identity matrix as

described by Equation 2.2

$${}^A_B \mathbf{R} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \text{ for } A = B \quad (2.2)$$

In application,  ${}^A_B \mathbf{R}$  may be a measurement of orientation provided by an AHRS where  $A$  is a coordinate system fixed to the Earth and  $B$  represents the axes of the IMU.

### 2.5.1.1 Matrix transpose

A matrix transpose operation switches the reference coordinate system. For example, the transpose of a rotation matrix describing coordinate system  $B$  relative to  $A$  would describe  $A$  relative to  $B$  as shown in 2.3. As a rotation matrix is by definition an orthogonal matrix, the transpose is equal to the inverse.

$${}^B_A \mathbf{R} = {}^A_B \mathbf{R}^{-1} = {}^A_B \mathbf{R}^T \quad (2.3)$$

### 2.5.1.2 Matrix multiplications

Matrix multiplications can be used to compound a chain of relative orientations into a single quantity. For example, the orientation of frame  $C$  relative to frame  $A$  can be computed if the separate rotation matrices describing  $C$  relative to  $B$  and  $B$  relative to  $A$  are known as demonstrated in Equation 2.4.

$${}^A_C \mathbf{R} = {}^A_B \mathbf{R} {}^B_C \mathbf{R} \quad (2.4)$$

The rotation matrix can be obtained using Craig's notation system by cancelling the diagonal opposite subscript and superscript. The requirement for these

diagonal matching pairs also provides means of validating a sequence.

### 2.5.1.3 Rotation of a vector

The multiplication of a rotation matrix by a vector quantity will alter the reference coordinate system of the vector and thus achieve a rotation. For example,  ${}^A\mathbf{v}$  is a vector defined in  $A$ , the vector can be equivalently described in  $B$  through the multiplication of Equation 2.5.

$${}^A\mathbf{v} = {}_B^A \mathbf{R} {}^B\mathbf{v} \quad (2.5)$$

An example application could be for  ${}^B\mathbf{v}$  to be an accelerometer measurement and for  ${}_B^A \mathbf{R}$  to describe the known misalignment of the accelerometer due to a mounting error. Equation 2.5 would provide  ${}^A\mathbf{v}$  as the accelerometer measurement corrected for misalignment.

## 2.5.2 Quaternions

Rotation matrices provide an intuitive means of manipulating three-dimensional quantities; however, the necessity for orthogonality is incompatible with processes such as interpolation and integration. Quaternions are four-dimensional complex numbers that offer an alternative, more compact representation to overcome these shortcomings.

Any orientation of coordinate system  $B$  relative to coordinate system  $A$  can be achieved through a single rotation of angle  $\theta$  around an axis  ${}^A\hat{\mathbf{r}}$ . This is demonstrated in Figure 2.2 where the mutually orthogonal vectors  $\hat{\mathbf{x}}_A$ ,  $\hat{\mathbf{y}}_A$  and  $\hat{\mathbf{z}}_A$ ; and  $\hat{\mathbf{x}}_B$ ,  $\hat{\mathbf{y}}_B$  and  $\hat{\mathbf{z}}_B$  define the principle axes of  $A$  and  $B$  respectively.

The quaternion describing the orientation of  $B$  relative to  $A$  is defined by Equation

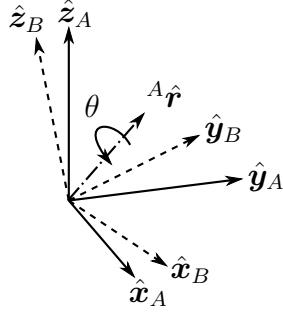


Figure 2.2: Any orientation of coordinate system  $B$  relative to coordinate system  $A$  can be achieved through a rotation of angle  $\theta$  around the axis  ${}^A\hat{\mathbf{r}}$

2.6 where  $r_x$ ,  $r_y$  and  $r_z$  are the elements of  ${}^A\hat{\mathbf{r}}$ . The last three elements form the imaginary parts of the quaternion.

$$\begin{aligned} {}_B^A \hat{\mathbf{q}} &= \begin{bmatrix} q_w & q_x & q_y & q_z \end{bmatrix} \\ &= \begin{bmatrix} \cos \frac{\theta}{2} & -r_x \sin \frac{\theta}{2} & -r_y \sin \frac{\theta}{2} & -r_z \sin \frac{\theta}{2} \end{bmatrix} \end{aligned} \quad (2.6)$$

A quaternion used to represent an orientation must be of unit magnitude. If the coordinate systems were to be in alignment then the quaternion would be an identity as described by Equation 2.7.

$${}_B^A \hat{\mathbf{q}} = \begin{bmatrix} 1 & 0 & 0 & 0 \end{bmatrix}, \text{ for } A = B \quad (2.7)$$

The convention used throughout this dissertation positions  $q_w$  as the first element, this is consistent with Kuipers [93] and MATLAB [94]. Other implementations may position  $q_w$  as the last element; notable examples include OpenGL libraries [95] and the Microsoft XNA Framework [96].

### 2.5.2.1 Quaternion conjugate

The quaternion conjugate, denoted by  $*$ , switches the reference coordinate system. For example, the conjugate of a quaternion describing coordinate system  $B$  relative to  $A$  would describe  $A$  relative to  $B$ . A conjugate is obtained by negating the

imaginary parts of the quaternion as shown in 2.8.

$${}^B_A \hat{\mathbf{q}} = {}^A_B \hat{\mathbf{q}}^* = \begin{bmatrix} q_w & -q_x & -q_y & -q_z \end{bmatrix} \quad (2.8)$$

### 2.5.2.2 Quaternion product

The quaternion product, denoted by  $\otimes$ , can be used to compound a chain of relative orientations into single quantity. For example, a quaternion describing  $C$  relative to frame  $A$  can be computed from  ${}^B_C \hat{\mathbf{q}}$  and  ${}^A_B \hat{\mathbf{q}}$  through the multiplication of Equation 2.9.

$${}^A_C \hat{\mathbf{q}} = {}^B_C \hat{\mathbf{q}} \otimes {}^A_B \hat{\mathbf{q}} \quad (2.9)$$

The multiplication of two quaternions,  $\hat{\mathbf{q}}_a$  and  $\hat{\mathbf{q}}_b$ , can be determined as the Hamilton product and defined as Equation 2.10 [93]. A quaternion product is not commutative; that is,  $\hat{\mathbf{q}}_a \otimes \hat{\mathbf{q}}_b \neq \hat{\mathbf{q}}_b \otimes \hat{\mathbf{q}}_a$ .

$$\begin{aligned} \hat{\mathbf{q}}_a \otimes \hat{\mathbf{q}}_b &= \begin{bmatrix} q_{aw} & q_{ax} & q_{ay} & q_{az} \end{bmatrix} \otimes \begin{bmatrix} q_{bw} & q_{bx} & q_{by} & q_{bz} \end{bmatrix} \\ &= \begin{bmatrix} q_{aw}q_{bw} - q_{ax}q_{bx} - q_{ay}q_{by} - q_{az}q_{bz} \\ q_{aw}q_{bx} + q_{ax}q_{bw} + q_{ay}q_{bz} - q_{az}q_{by} \\ q_{aw}q_{by} - q_{ax}q_{bz} + q_{ay}q_{bw} + q_{az}q_{bx} \\ q_{aw}q_{bz} + q_{ax}q_{by} - q_{ay}q_{bx} + q_{az}q_{bw} \end{bmatrix}^T \end{aligned} \quad (2.10)$$

### 2.5.2.3 Quaternion rotation of a vector

A three dimensional vector can be rotated by a quaternion using the relationship described in Equation 2.11 [93]. In this equation,  ${}^A \mathbf{v}$  and  ${}^B \mathbf{v}$  are the same vector described in frame  $A$  and frame  $B$  respectively with a relative orientation defined by  ${}^A_B \hat{\mathbf{q}}$ .

$$\begin{bmatrix} 0 & {}^B \mathbf{v}^T \end{bmatrix} = {}^A_B \hat{\mathbf{q}} \otimes \begin{bmatrix} 0 & {}^A \mathbf{v}^T \end{bmatrix} \otimes {}^A_B \hat{\mathbf{q}}^* \quad (2.11)$$

### 2.5.2.4 Converting a quaternion to a rotation matrix

An equivalent rotation matrix can be obtained from a quaternion through the conversion described by Equation 2.12 [93] where  $q_w$ ,  $q_x$ ,  $q_y$  and  $q_z$  are the elements of  ${}^A_B \hat{\mathbf{q}}$ .

$${}^A_B \mathbf{R} = \begin{bmatrix} 2q_w^2 - 1 + 2q_x^2 & 2q_xq_y + 2q_wq_z & 2q_xq_z - 2q_wq_y \\ 2q_xq_y - 2q_wq_z & 2q_w^2 - 1 + 2q_y^2 & 2q_yq_z + 2q_wq_x \\ 2q_xq_z + 2q_wq_y & 2q_yq_z - 2q_wq_x & 2q_w^2 - 1 + 2q_z^2 \end{bmatrix} \quad (2.12)$$

### 2.5.2.5 Converting a rotation matrix to quaternion

A poorly formulated rotation matrix could potentially corrupt a direct conversion to a quaternion. Bar-Itzhack provides an alternative method [97] to extract the ‘best fit’ quaternion from an imprecise and non-orthogonal rotation matrix. The method requires the construction of the symmetric  $4 \times 4$  matrix  $\mathbf{K}$ , constructed from the elements of the rotation matrix  ${}^A_B \mathbf{R}$  as defined by Equation 2.13. The optimal quaternion,  ${}^A_B \hat{\mathbf{q}}$ , is found as the normalised Eigenvector corresponding to the maximum Eigenvalue,  $\lambda_{max}$ , of  $\mathbf{K}$ . The element order must be altered to account for the alternative convention, as described by Equation 2.14 where  $v_0$  to  $v_3$  are the elements of the normalised Eigenvector.

$$\mathbf{K} = \frac{1}{3} \begin{bmatrix} r_{11} - r_{22} - r_{33} & r_{21} + r_{12} & r_{31} + r_{13} & r_{23} - r_{32} \\ r_{21} + r_{12} & r_{22} - r_{11} - r_{33} & r_{32} + r_{23} & r_{31} - r_{13} \\ r_{31} + r_{13} & r_{32} + r_{23} & r_{33} - r_{11} - r_{22} & r_{12} - r_{21} \\ r_{23} - r_{32} & r_{31} - r_{13} & r_{12} - r_{21} & r_{11} + r_{22} + r_{33} \end{bmatrix} \quad (2.13)$$

$${}^A_B \hat{\mathbf{q}} = \begin{bmatrix} v_3 & v_0 & v_1 & v_2 \end{bmatrix} \quad (2.14)$$

Some practical applications require that a ‘best fit’ rotation matrix is obtained from an approximated rotation matrix derived from measurement data. This can

be achieved by first obtaining a quaternion using the method above and then converting the quaternion into a rotation matrix as per Equation 2.12.

### 2.5.3 Euler angles

Although rotation matrices and quaternions provide effective tools for the manipulation of spatial quantities, their numerical values are not immediately meaningful. Euler angles describe an orientation as three angular quantities and so offer a more intuitive representation. Perhaps the most common use of Euler angles is within the *aerospace sequence* where the three angular quantities  $\phi$ ,  $\theta$  and  $\psi$  correspond to rotations around the  $x$ ,  $y$  and  $z$  axes respectively as shown in Figure 2.3.  $\phi$ ,  $\theta$  and  $\psi$  are commonly referred to as roll, pitch and yaw (or heading) respectively.

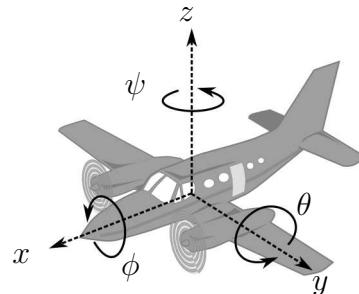


Figure 2.3: The *aerospace sequence* of Euler angles describe orientation as the result of angular rotations  $\phi$ ,  $\theta$  and  $\psi$  around the  $x$ ,  $y$  and  $z$  axes respectively

Euler angles describe an orientation that is achieved through a specific sequence of rotations. The *aerospace sequence*, also referred to as the ZYX sequence, is just one of twelve distinct possibilities and defines the order as a rotation first around the  $z$  axis, then the  $y$  axis and then the  $x$  axis. The direction of each rotational axis relative to the original orientation is therefore dependent on the previous rotation in the sequence. This is demonstrated in Figure 2.4 where the three diagrams from left to right show the successive rotations of  $B$  around each axis to achieve the ultimate orientation of  $B$  relative to  $A$ .

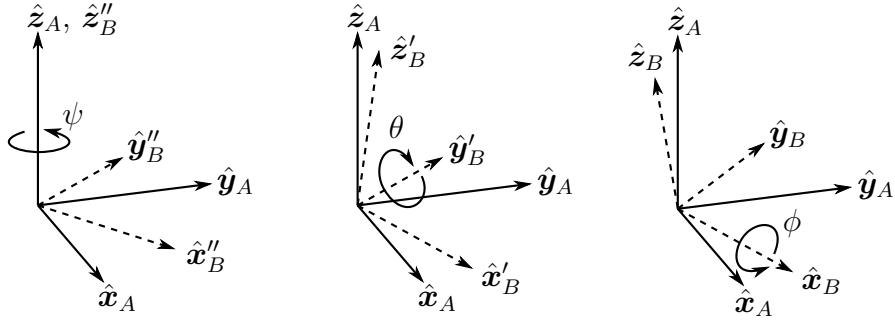


Figure 2.4: The Euler angle sequence describes successive rotations of around each axis to achieve the ultimate orientation of  $B$  relative to  $A$

Although an angular representation may be convenient, the sequence introduces an inherent hazard that must be accounted for in any application utilising Euler angles. When  $\theta$  approaches  $\pm 90^\circ$ , the rotational axis of  $\phi$  and  $\psi$  relative to the original orientation align; this singularity means that there are infinite solutions to the Euler angle sequence. As a result, calculated values of  $\phi$  and  $\psi$  can be expected to become erratic and unreliable. This phenomena is commonly referred to as *gimbal lock*. In practice, applications should utilise a rotation matrix or quaternion representation wherever possible with Euler angles used only as an ultimate output where the singularity hazard does not have an opportunity to feedback into numerical processes.

### 2.5.3.1 Converting a rotation matrix to Euler angles

The ZYX Euler angles  $\phi$ ,  $\theta$  and  $\psi$  describing the orientation of  $B$  relative to  $A$  can be calculated from the rotation matrix  ${}^A_B \hat{\mathbf{R}}$  as Equations 2.15 to 2.17 [64].

$$\phi = \text{atan2}(r_{32}, r_{33}) \quad (2.15)$$

$$\theta = -\arcsin(r_{31}) \quad (2.16)$$

$$\psi = \text{atan2}(r_{21}, r_{11}) \quad (2.17)$$

### 2.5.3.2 Converting a quaternion to Euler angles

The ZYX Euler angles can be obtained from the quaternion  ${}^A_B\hat{\mathbf{q}}$  using the Equations 2.15 to 2.17 by substituting each element of the rotation matrix with function of the quaternion as per Equation 2.12 to yield Equations 2.18 to 2.20.

$$\phi = \text{atan2}(2(q_y q_z - q_w q_x), 2q_w^2 - 1 + 2q_z^2) \quad (2.18)$$

$$\theta = -\arcsin(2(q_x q_z + q_w q_y)) \quad (2.19)$$

$$\psi = \text{atan2}(2(q_x q_y - q_w q_z), 2q_w^2 - 1 + 2q_x^2) \quad (2.20)$$



# **Chapter 3**

## **Initial Attitude and Heading**

### **Reference System algorithm**

#### **3.1 Introduction**

Some of the first work conducted in this research project was the development of an AHRS algorithm. Although the algorithm would be significantly revised over the proceeding years (see Chapter 7), this initial algorithm offered a novel solution that would become widely adopted within academic and commercial projects. The algorithm was intended to provide an alternative solution to computationally expensive Kalman-based solutions and explore an alternative approach to the complementary filter solutions available. The principle goals were:

- to minimise the computational load for applicability to low-power embedded systems.
- address the short comings of magnetometer integration previously discussed in Chapter 2.

The algorithm design was inspired by the novel Gauss-Newton-based solution developed by Bachmann *et al.* [63]. Their solution demonstrated excellent performance. However, the least squares regression associated with the Gauss-Newton process imposes a significant computational load. The solution presented here operates on a similar principle but implements an analytically derived gradient descent algorithm and so avoids the need for a matrix inversion. A quaternion representation facilitates the derivation of the algorithm and avoids the problematic singularities associated with an Euler angle representation.

### 3.1.1 Chapter summary

Section 3.2 delineates the mathematical derivation of the orientation estimation algorithm, including a description of the parametrisation and compensation for magnetic distortion. Section 3.3 describes the experimental equipment used to test and verify the performance of the algorithm. Section 3.4 quantifies the experimental testing and accuracy of the algorithm and compares it to existing systems. Section 3.5 summarises conclusions and contributions of this work.

## 3.2 Algorithm derivation

### 3.2.1 Orientation from angular rate

A tri-axis gyroscope will measure the angular rate about the  $x$ ,  $y$  and  $z$  axes of the sensor frame, termed  $\omega_x$ ,  $\omega_y$  and  $\omega_z$  respectively. If these parameters (in rads $^{-1}$ ) are arranged into the vector  ${}^S\boldsymbol{\omega}$  defined by Equation 3.1, the quaternion derivative describing rate of change of the Earth frame relative to the sensor frame  ${}^S_E\dot{\boldsymbol{q}}$  can be calculated [98] as Equation 3.2. The  $\otimes$  operate denotes a quaternion product

and the  $\hat{\cdot}$  accent denotes a normalised vector of unit length.

$${}^S\boldsymbol{\omega} = \begin{bmatrix} 0 & \omega_x & \omega_y & \omega_z \end{bmatrix} \quad (3.1)$$

$${}^S_E\dot{\boldsymbol{q}} = \frac{1}{2} {}^S_E\hat{\boldsymbol{q}} \otimes {}^S\boldsymbol{\omega} \quad (3.2)$$

The orientation of the Earth frame relative to the sensor frame at time  $t$ ,  ${}^E_S\boldsymbol{q}_{\omega,t}$ , can be computed by numerically integrating the quaternion derivative  ${}^S_E\dot{\boldsymbol{q}}_{\omega,t}$  as described by Equations 3.3 and 3.4, provided that initial conditions are known. In these equations,  ${}^S\boldsymbol{\omega}_t$  is the angular rate measured at time  $t$ ,  $\Delta t$  is the sampling period and  ${}^S_E\hat{\boldsymbol{q}}_{est,t-1}$  is the previous estimate of orientation. The subscript  $\omega$  indicates that the quaternion is calculated from angular rates.

$${}^S_E\dot{\boldsymbol{q}}_{\omega,t} = \frac{1}{2} {}^S_E\hat{\boldsymbol{q}}_{est,t-1} \otimes {}^S\boldsymbol{\omega}_t \quad (3.3)$$

$${}^S_E\boldsymbol{q}_{\omega,t} = {}^S_E\hat{\boldsymbol{q}}_{est,t-1} + {}^S_E\dot{\boldsymbol{q}}_{\omega,t} \Delta t \quad (3.4)$$

### 3.2.2 Orientation from a homogeneous field

In the context of an orientation estimation algorithm, it will initially be assumed that an accelerometer will measure only gravity and a magnetometer will measure only the Earth's magnetic field. If the direction of the Earth's field is known in the Earth frame, a measurement of the field's direction within the sensor frame will allow an orientation of the sensor frame relative to the Earth frame to be calculated. However, for any given measurement there will not be a unique sensor orientation solution, instead there will infinite solutions represented by all those

orientations achieved by the rotation of the true orientation around an axis parallel with the field. A quaternion representation requires a single solution to be found. This may be achieved through the formulation of an optimisation problem where an orientation of the sensor,  ${}_E^S \hat{\mathbf{q}}$ , is found as that which aligns a predefined reference direction of the field in the Earth frame,  ${}^E \hat{\mathbf{d}}$ , with the measured field in the sensor frame,  ${}^S \hat{\mathbf{s}}$ ; thus solving 3.5 where Equation 3.6 defines the objective function.

$$\min_{{}_E^S \hat{\mathbf{q}} \in \Re^4} \mathbf{f}({}_E^S \hat{\mathbf{q}}, {}^E \hat{\mathbf{d}}, {}^S \hat{\mathbf{s}}) \quad (3.5)$$

$$\mathbf{f}({}_E^S \hat{\mathbf{q}}, {}^E \hat{\mathbf{d}}, {}^S \hat{\mathbf{s}}) = {}_E^S \hat{\mathbf{q}}^* \otimes {}^E \hat{\mathbf{d}} \otimes {}_E^S \hat{\mathbf{q}} - {}^S \hat{\mathbf{s}} \quad (3.6)$$

Many optimisation algorithms exist but the gradient descent algorithm is one of the simplest to both implement and compute. Equation 3.7 describes the gradient descent algorithm for  $n$  iterations resulting in an orientation estimation of  ${}_E^S \hat{\mathbf{q}}_{n+1}$  based on an ‘initial guess’ orientation  ${}_E^S \hat{\mathbf{q}}_0$  and a variable step-size  $\mu$ . Equation 3.8 computes an error direction on the solution surface defined by the objective function,  $\mathbf{f}$ , and its Jacobian,  $\mathbf{J}$ .

$${}_E^S \mathbf{q}_{k+1} = {}_E^S \hat{\mathbf{q}}_k - \mu \frac{\nabla \mathbf{f}^T({}_E^S \hat{\mathbf{q}}_k, {}^E \hat{\mathbf{d}}, {}^S \hat{\mathbf{s}})}{\left\| \nabla \mathbf{f}^T({}_E^S \hat{\mathbf{q}}_k, {}^E \hat{\mathbf{d}}, {}^S \hat{\mathbf{s}}) \right\|}, \quad k = 0, 1, 2, \dots, n \quad (3.7)$$

$$\nabla \mathbf{f}({}_E^S \hat{\mathbf{q}}_k, {}^E \hat{\mathbf{d}}, {}^S \hat{\mathbf{s}}) = \mathbf{J}^T({}_E^S \hat{\mathbf{q}}_k, {}^E \hat{\mathbf{d}}) \mathbf{f}({}_E^S \hat{\mathbf{q}}_k, {}^E \hat{\mathbf{d}}, {}^S \hat{\mathbf{s}}) \quad (3.8)$$

Equations 3.7 and 3.8 describe the general form of the algorithm applicable to a field predefined in any direction. However, if the reference direction of the field is defined to only have components within 1 or 2 of the principle axes of the Earth coordinate frame then the equations simplify. An appropriate convention would be to assume that the direction of gravity defines the vertical,  $z$  axis as shown

in Equation 3.10. Substituting  ${}^E\hat{\mathbf{g}}$  and normalised accelerometer measurement  ${}^S\hat{\mathbf{a}}$  for  ${}^E\hat{\mathbf{d}}$  and  ${}^S\hat{\mathbf{s}}$  respectively, yields the simplified objective function and Jacobian defined by Equations 3.12 and 3.13.

$${}^S_E\hat{\mathbf{q}} = \begin{bmatrix} q_w & q_x & q_y & q_z \end{bmatrix} \quad (3.9)$$

$${}^E\hat{\mathbf{g}} = \begin{bmatrix} 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.10)$$

$${}^S\hat{\mathbf{a}} = \begin{bmatrix} 0 & a_x & a_y & a_z \end{bmatrix} \quad (3.11)$$

$$\mathbf{f}_g({}^S_E\hat{\mathbf{q}}, {}^S\hat{\mathbf{a}}) = \begin{bmatrix} 2(q_x q_z - q_w q_y) - a_x \\ 2(q_w q_x + q_y q_z) - a_y \\ 2(\frac{1}{2} - q_x^2 - q_y^2) - a_z \end{bmatrix} \quad (3.12)$$

$$\mathbf{J}_g({}^S_E\hat{\mathbf{q}}) = \begin{bmatrix} -2q_y & 2q_z & -2q_w & 2q_x \\ 2q_x & 2q_w & 2q_z & 2q_y \\ 0 & -4q_x & -4q_y & 0 \end{bmatrix} \quad (3.13)$$

The Earth's magnetic field can be considered to have components in one horizontal axis and the vertical axis; the vertical component due to the inclination of the field which is between 65° and 70° to the horizontal in the United Kingdom (UK) [99]. This can be represented by Equation 3.14. Substituting  ${}^E\hat{\mathbf{b}}$  and normalised magnetometer measurement  ${}^S\hat{\mathbf{m}}$  for  ${}^E\hat{\mathbf{d}}$  and  ${}^S\hat{\mathbf{s}}$  respectively, yields the simplified objective function and Jacobian defined by Equations 3.16 and 3.17.

$${}^E\hat{\mathbf{b}} = \begin{bmatrix} 0 & b_x & 0 & b_z \end{bmatrix} \quad (3.14)$$

$${}^S\hat{\mathbf{m}} = \begin{bmatrix} 0 & m_x & m_y & m_z \end{bmatrix} \quad (3.15)$$

$$\mathbf{f}_b({}^S_E\hat{\mathbf{q}}, {}^E\hat{\mathbf{b}}, {}^S\hat{\mathbf{m}}) = \begin{bmatrix} 2b_x(0.5 - q_y^2 - q_z^2) + 2b_z(q_x q_z - q_w q_y) - m_x \\ 2b_x(q_x q_y - q_w q_z) + 2b_z(q_w q_x + q_y q_z) - m_y \\ 2b_x(q_w q_y + q_x q_z) + 2b_z(0.5 - q_x^2 - q_y^2) - m_z \end{bmatrix} \quad (3.16)$$

$$\mathbf{J}_b({}^S_E\hat{\mathbf{q}}, {}^E\hat{\mathbf{b}}) = \begin{bmatrix} -2b_z q_y & 2b_z q_z \\ -2b_x q_z + 2b_z q_x & 2b_x q_y + 2b_z q_w \\ 2b_x q_y & 2b_x q_z - 4b_z q_x \\ -4b_x q_y - 2b_z q_w & -4b_x q_z + 2b_z q_x \\ 2b_x q_x + 2b_z q_z & -2b_x q_w + 2b_z q_y \\ 2b_x q_w - 4b_z q_y & 2b_x q_x \end{bmatrix} \quad (3.17)$$

As has already been discussed, the measurement of gravity or the Earth's magnetic field alone will not provide a unique orientation of the sensor. To do so, the measurements and reference directions of both fields may be combined as described by Equations 3.18 and 3.19. Whereas the solution surface created by the objective functions in Equations 3.12 and 3.16 have a global minimum defined by a line, the solution surface defined by Equation 3.18 has a minimum defined by a single point, provided that  $b_x \neq 0$ .

$$\mathbf{f}_{g,b}({}^S_E\hat{\mathbf{q}}, {}^S\hat{\mathbf{a}}, {}^E\hat{\mathbf{b}}, {}^S\hat{\mathbf{m}}) = \begin{bmatrix} \mathbf{f}_g({}^S_E\hat{\mathbf{q}}, {}^S\hat{\mathbf{a}}) \\ \mathbf{f}_b({}^S_E\hat{\mathbf{q}}, {}^E\hat{\mathbf{b}}, {}^S\hat{\mathbf{m}}) \end{bmatrix} \quad (3.18)$$

$$\mathbf{J}_{g,b}({}_E^S\hat{\mathbf{q}}, {}^E\hat{\mathbf{b}}) = \begin{bmatrix} \mathbf{J}_g^T({}_E^S\hat{\mathbf{q}}) \\ \mathbf{J}_b^T({}_E^S\hat{\mathbf{q}}, {}^E\hat{\mathbf{b}}) \end{bmatrix} \quad (3.19)$$

A conventional approach to optimisation would require multiple iterations of Equation 3.7 to be computed for each new orientation and corresponding sensor measurements. However, it is acceptable to compute one iteration per time sample provided that the convergence rate of the estimated orientation governed by  $\mu_t$  is equal or greater than the rate of change of physical orientation. Equation 3.20 calculates the estimated orientation  ${}_E^S\mathbf{q}_{\nabla,t}$  computed at time  $t$  based on a previous estimate of orientation  ${}_E^S\hat{\mathbf{q}}_{est,t-1}$  and the objective function error  $\nabla\mathbf{f}$  defined by sensor measurements  ${}^S\hat{\mathbf{a}}_t$  and  ${}^S\hat{\mathbf{m}}_t$  sampled at time  $t$ . The form of  $\nabla\mathbf{f}$  is chosen according to the sensors in use, as shown in Equation 3.21. The subscript  $\nabla$  indicates that the quaternion is calculated using the gradient descent algorithm.

$${}_E^S\mathbf{q}_{\nabla,t} = {}_E^S\hat{\mathbf{q}}_{est,t-1} - \mu_t \frac{\nabla\mathbf{f}^T}{\|\nabla\mathbf{f}^T\|} \quad (3.20)$$

$$\nabla\mathbf{f} = \begin{cases} \mathbf{J}_g^T({}_E^S\hat{\mathbf{q}}_{est,t-1})\mathbf{f}_g({}_E^S\hat{\mathbf{q}}_{est,t-1}, {}^S\hat{\mathbf{a}}_t) \\ \mathbf{J}_{g,b}^T({}_E^S\hat{\mathbf{q}}_{est,t-1}, {}^E\hat{\mathbf{b}})\mathbf{f}_{g,b}({}_E^S\hat{\mathbf{q}}_{est,t-1}, {}^S\hat{\mathbf{a}}_t, {}^E\hat{\mathbf{b}}, {}^S\hat{\mathbf{m}}_t) \end{cases} \quad (3.21)$$

An appropriate value of  $\mu_t$  is that which ensures the convergence rate of  ${}_E^S\mathbf{q}_{\nabla,t}$  is limited to the physical orientation rate as this avoids overshooting due an unnecessarily large step size. Therefore  $\mu_t$  can be calculated as Equation 3.22 where  $\Delta t$  is the sampling period,  ${}_E^S\dot{\mathbf{q}}_{\omega,t}$  is the rate of change of orientation measured by gyroscopes and  $\alpha$  is an augmentation of  $\mu$  to account for noise in accelerometer and magnetometer measurements.

$$\mu_t = \alpha \|{}_E^S\dot{\mathbf{q}}_{\omega,t}\| \Delta t, \quad \alpha > 1 \quad (3.22)$$

### 3.2.3 Algorithm fusion process

In practice,  ${}^S_E \mathbf{q}_{\omega,t}$  may start from incorrect initial conditions and accumulate errors due to gyroscope measurement noise and  ${}^S_E \mathbf{q}_{\nabla,t}$  will provide an incorrect estimate when the accelerometer is not stationary or the magnetometer exposed to interferences. The goal of the fusion algorithm is to provide an orientation estimate where  ${}^S_E \mathbf{q}_{\omega,t}$  is used to filter out high frequency errors in  ${}^S_E \mathbf{q}_{\nabla,t}$ , and  ${}^S_E \mathbf{q}_{\nabla,t}$  is used both to compensate for integral drift in  ${}^S_E \mathbf{q}_{\omega,t}$  and to provide convergence from initial conditions.

An estimated orientation of the Earth frame relative to the sensor frame,  ${}^S_E \mathbf{q}_{est,t}$ , is obtained through the fusion of the two separate orientation calculations,  ${}^S_E \mathbf{q}_{\omega,t}$  and  ${}^S_E \mathbf{q}_{\nabla,t}$  as described by Equation 3.23 where  $\gamma_t$  and  $(1 - \gamma_t)$  are weights applied to each orientation calculation.

$${}^S_E \mathbf{q}_{est,t} = \gamma_t {}^S_E \mathbf{q}_{\nabla,t} + (1 - \gamma_t) {}^S_E \mathbf{q}_{\omega,t}, \quad 0 \leq \gamma_t \leq 1 \quad (3.23)$$

An optimal value of  $\gamma_t$  is therefore that which ensures the weighted rate of divergence of  ${}^S_E \mathbf{q}_{\omega}$  due to integral drift is equal to the weighted rate of convergence of  ${}^S_E \mathbf{q}_{\nabla}$ . This is represented by Equation 3.24 where  $\frac{\mu_t}{\Delta t}$  is the convergence rate of  ${}^S_E \mathbf{q}_{\nabla}$  and  $\beta$  is the divergence rate of  ${}^S_E \mathbf{q}_{\omega}$  expressed as the magnitude of a quaternion derivative corresponding to the gyroscope measurement error. Equation 3.24 can be rearranged to define  $\gamma_t$  as Equation 3.25.

$$(1 - \gamma_t)\beta = \gamma_t \frac{\mu_t}{\Delta t} \quad (3.24)$$

$$\gamma_t = \frac{\beta}{\frac{\mu_t}{\Delta t} + \beta} \quad (3.25)$$

The fusion process ensures the optimal fusion of  ${}^S_E\mathbf{q}_{\omega,t}$  and  ${}^S_E\mathbf{q}_{\nabla,t}$  assuming that the convergence rate of  ${}^S_E\mathbf{q}_{\nabla}$  governed by  $\alpha$  is equal or greater than the physical rate of change of orientation. Therefore  $\alpha$  has no upper bound. If  $\alpha$  is assumed to be very large then  $\mu_t$ , defined by Equation 3.22, also becomes very large and the equations simplify. A large value of  $\mu_t$  used in Equation 3.20 means that  ${}^S_E\hat{\mathbf{q}}_{est,t-1}$  becomes negligible and the equation can be re-written as Equation 3.26.

$${}^S_E\mathbf{q}_{\nabla,t} \approx -\mu_t \frac{\nabla \mathbf{f}}{\|\nabla \mathbf{f}\|} \quad (3.26)$$

The definition of  $\gamma_t$  in Equation 3.25 also simplifies if the  $\beta$  term in the denominator becomes negligible and the equation can be rewritten as Equation 3.27. It is possible from Equation 3.27 to also assume that  $\gamma_t \approx 0$ .

$$\gamma_t \approx \frac{\beta \Delta t}{\mu_t} \quad (3.27)$$

Substituting Equations 3.4, 3.26 and 3.27 into Equation 3.23 directly yields Equation 3.28. It is important to note that in Equation 3.28,  $\gamma_t$  has been substituted as both as Equation 3.26 and 0.

$${}^S_E\mathbf{q}_{est,t} = \frac{\beta \Delta t}{\mu_t} \left( -\mu_t \frac{\nabla \mathbf{f}}{\|\nabla \mathbf{f}\|} \right) + (1 - 0) \left( {}^S_E\hat{\mathbf{q}}_{est,t-1} + {}^S_E\dot{\mathbf{q}}_{\omega,t} \Delta t \right) \quad (3.28)$$

Equation 3.28 can be simplified to Equation 3.29 where  ${}^S_E\dot{\mathbf{q}}_{est,t}$  is the estimated orientation rate defined by Equation 3.30.

$${}^S_E\mathbf{q}_{est,t} = {}^S_E\hat{\mathbf{q}}_{est,t-1} + {}^S_E\dot{\mathbf{q}}_{est,t} \Delta t \quad (3.29)$$

$${}^S_E\dot{\mathbf{q}}_{est,t} = {}^S_E\dot{\mathbf{q}}_{\omega,t} - \beta \frac{\nabla \mathbf{f}}{\|\nabla \mathbf{f}\|} \quad (3.30)$$

It can be seen from Equations 3.29 and 3.30 that the algorithm calculates the orientation  ${}^S_E \mathbf{q}_{est}$  by numerically integrating the estimated rate of change of orientation  ${}^S_E \dot{\mathbf{q}}_{est}$ . The algorithm computes  ${}^S_E \dot{\mathbf{q}}_{est}$  as the rate of change of orientation measured by the gyroscopes,  ${}^S_E \dot{\omega}_t$ , with the magnitude of the gyroscope measurement error,  $\beta$ , removed in a direction based on accelerometer and magnetometer measurements. Figure 3.1 shows a block diagram representation of the complete orientation estimation algorithm implementation for an IMU.

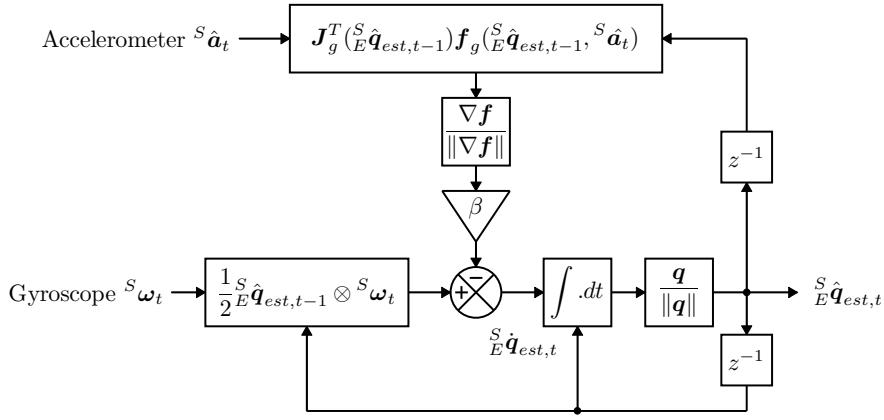


Figure 3.1: Block diagram representation of the complete orientation estimation algorithm for an IMU implementation

### 3.2.4 Magnetic distortion compensation

Investigations into the effect of magnetic distortions on an orientation sensor's performance have shown that substantial errors may be introduced by sources including electrical appliances, metal furniture and metal structures within a building's construction [100, 101]. Soft-iron and hard-iron distortions are interference fixed in the sensor frame and can be removed through calibration [102, 29, 103]. Sources of interference in the Earth frame may only be removed if an additional reference of orientation is available. An accelerometer provides a reference of attitude and so may be used to compensate for inclination errors in the measured Earth's magnetic field.

The measured direction of the Earth's magnetic field in the Earth frame at time  $t$ ,  ${}^E\hat{\mathbf{h}}_t$ , can be computed as Equation 3.31. The effect of an erroneous inclination of the measured direction Earth's magnetic field,  ${}^E\hat{\mathbf{h}}_t$ , can be corrected if the algorithm's reference direction of the Earth's magnetic field,  ${}^E\hat{\mathbf{b}}_t$ , is of the same inclination. This is achieved by computing  ${}^E\hat{\mathbf{b}}_t$  as  ${}^E\hat{\mathbf{h}}_t$  normalised to have only components in the Earth frame  $x$  and  $z$  axes; as described by Equation 3.32.

$${}^E\hat{\mathbf{h}}_t = \begin{bmatrix} 0 & h_x & h_y & h_z \end{bmatrix} = {}^S_E\hat{\mathbf{q}}_{est,t-1} \otimes {}^S\hat{\mathbf{m}}_t \otimes {}^S_E\hat{\mathbf{q}}_{est,t-1}^* \quad (3.31)$$

$${}^E\hat{\mathbf{b}}_t = \begin{bmatrix} 0 & \sqrt{h_x^2 + h_y^2} & 0 & h_z \end{bmatrix} \quad (3.32)$$

Compensating for magnetic distortions in this way ensures that magnetic disturbances are limited to only affect the estimated heading component of orientation. The approach also eliminates the need for the reference direction of the Earth's magnetic field to be predefined; a potential disadvantage of other orientation estimation algorithms [60, 104]. Figure 3.2 shows a block diagram representation of the complete algorithm implementation for a Magnetic, Angular Rate and Gravity (MARG) sensor array, including the magnetic distortion compensation.

### 3.2.5 Algorithm adjustable parameter

The orientation estimation algorithm requires one adjustable parameter,  $\beta$ , representing the gyroscope measurement error expressed as the magnitude of a quaternion derivative. It is convenient to define  $\beta$  using the angular quantity  $\tilde{\omega}_{max}$  representing the maximum gyroscope measurement error of each axis. Using the relationship described by Equation 3.2,  $\beta$  may be defined by Equation 3.33 where

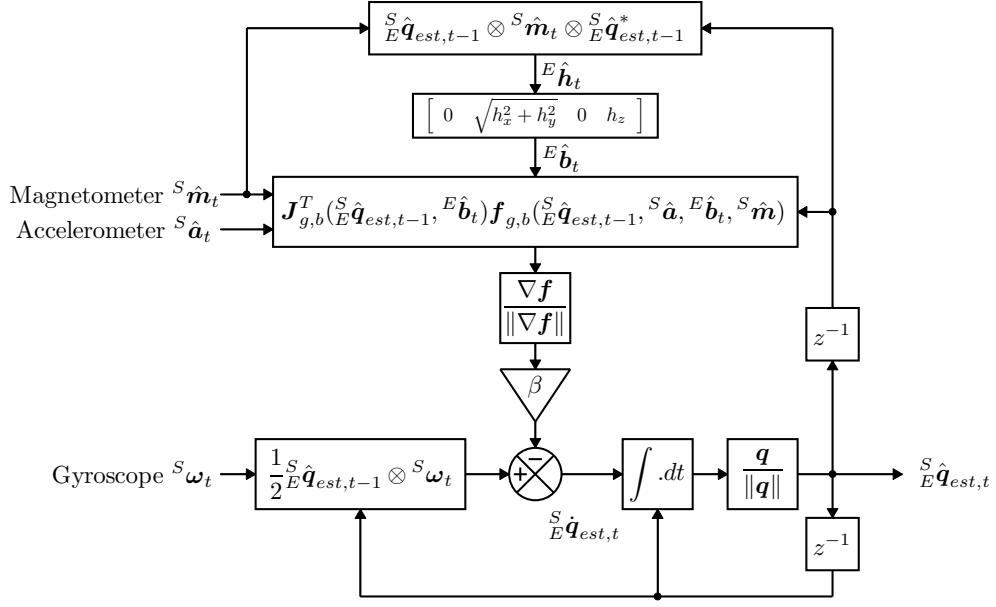


Figure 3.2: Block diagram representation of the complete orientation estimation algorithm for an MARG implementation including magnetic distortion compensation

$\hat{q}$  is any unit quaternion.

$$\beta = \left\| \frac{1}{2} \hat{q} \otimes \begin{bmatrix} 0 & \tilde{\omega}_{max} & \tilde{\omega}_{max} & \tilde{\omega}_{max} \end{bmatrix} \right\| = \sqrt{\frac{3}{4}} \tilde{\omega}_{max} \quad (3.33)$$

### 3.3 Experimental equipment

The algorithm was tested using the Xsens MTx orientation sensor [105] containing 16-bit resolution tri-axis gyroscopes, accelerometers and magnetometers. Raw sensor data was logged to a computer at 512 Hz and imported using the accompanying software to provide calibrated sensor measurements which were then processed by the proposed orientation estimation algorithm. The software also incorporates a propriety Kalman-based orientation estimation algorithm. As both the Kalman-based algorithm and proposed algorithm's estimates of orientation were computed using identical sensor data, the performance of each algorithm could be evaluated relative to one-another, independent of sensor performance.

The Xsens Kalman-based algorithm was used with the factory default settings.

A Vicon system, consisting of eight MX3+ cameras connected to an MXultranet server [106] and Nexus [107] software, was used to provide reference measurements of the orientation sensor's actual orientation. To do so, the sensor was fixed to an orientation measurement platform. The positions of optical markers attached to the platform were logged at 120 Hz and then post-processed to compute the orientation of the measurement platform and sensor. In order for the measurements of an orientation in the camera coordinate frame to be comparable to the algorithm estimate of orientation in the Earth frame, an initial calibration procedure was required where the direction of the Earth's magnetic and gravitational fields in the camera coordinate frame were measured using a magnetic compass and pendulum with attached optical markers.

### 3.4 Experimental results

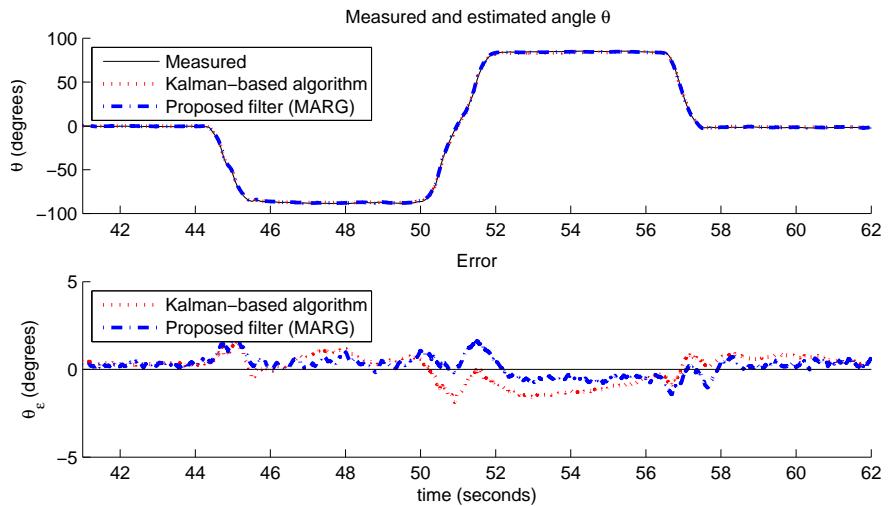


Figure 3.3: Typical results for measured and estimated angle  $\theta$  (top) and error (bottom)

It is common [104, 105, 108, 109, 110] to quantify orientation sensor performance as the static and dynamic Root Mean Squared (RMS) errors in the decoupled

Euler parameter	Kalman-based algorithm	MARG algorithm	IMU algorithm
RMS[ $\phi_\epsilon$ ] static	0.789°	0.581°	0.594°
RMS[ $\phi_\epsilon$ ] dynamic	0.769°	0.625°	0.623°
RMS[ $\theta_\epsilon$ ] static	0.819°	0.502°	0.497°
RMS[ $\theta_\epsilon$ ] dynamic	0.847°	0.668°	0.668°
RMS[ $\psi_\epsilon$ ] static	1.150°	1.073°	N/A
RMS[ $\psi_\epsilon$ ] dynamic	1.344°	1.110°	N/A

Table 3.1: Static and dynamic RMS error of Kalman-based algorithm and proposed algorithm IMU and MARG implementations

Euler parameters describing the pitch,  $\phi$ , roll,  $\theta$  and heading,  $\psi$  components of an orientation, corresponding to rotations around the sensor frame  $x$ ,  $y$ , and  $z$  axis respectively. A total of four sets of Euler parameters were computed, corresponding to the calibrated optical measurements of orientation, the Kalman-based algorithm estimated orientation and the proposed algorithm estimates orientation for both the MARG and IMU implementations. The errors of estimated Euler parameters,  $\phi_\epsilon$ ,  $\theta_\epsilon$  and  $\psi_\epsilon$ , were computed as the difference between estimated values and the calibrated optical measurements. Results were obtained for a sequence of rotations around each axis preformed by hand. The experiment was repeated eight times to compile a dataset representative of system performance. The proposed algorithm's adjustable parameter,  $\beta$ , was set to 0.033 for the MARG implementation and 0.041 for the IMU implementation. Trials summarised in Figure 3.4, found these values to provide optimal performance. Figure 3.3 shows the Kalman-based algorithm and proposed algorithm MARG implementation results, typical of the eight experiments.

The static and dynamic RMS values of  $\phi_\epsilon$ ,  $\theta_\epsilon$ , and  $\psi_\epsilon$  were calculated assuming a static state when the measured corresponding angular rate was  $< 5^\circ/\text{s}$ , and a dynamic when  $\geq 5^\circ/\text{s}$ . This threshold was chosen to be sufficiently greater than the noise floor of the data. The results are summarised in Table 3.4 where each value represents the mean of all eight experiments.

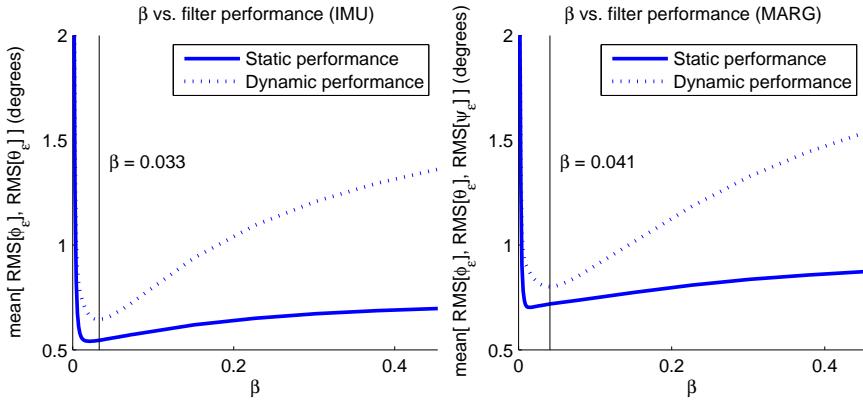


Figure 3.4: The effect of the adjustable parameter,  $\beta$ , on the performance of the proposed algorithm IMU (left) and MARG (right) implementations

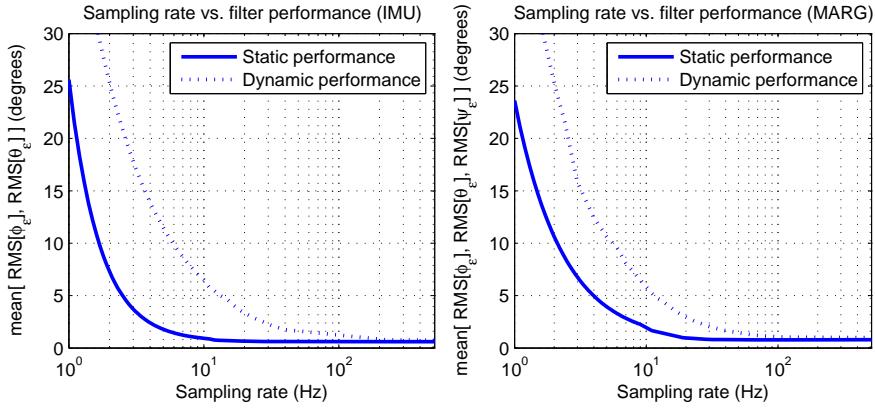


Figure 3.5: The effect of sampling rate on the performance of the proposed algorithm IMU (left) and MARG (right) implementations

The results of an investigation into the effect of the adjustable parameter  $\beta$  on algorithm performance are summarised in Figure 3.4. The experimental data was processed through the separate proposed algorithm IMU and MARG implantations, using fixed values of  $\beta$  between 0 to 0.5. There is a clear optimal value of  $\beta$  high enough to minimise errors due to integral drift but sufficiently low enough that unnecessary noise is not introduced by large steps of gradient descent iterations.

The results of an investigation into the effect of sampling rate on algorithm performance is summarised in Figure 3.5. The experimental data was processed through the separate proposed algorithm IMU and MARG implantations, using the previously defined, optimal values  $\beta$ . Experimental data was decimated to

simulate sampling rates between 1 Hz and 512 Hz. It can be seen from Figure 3.5 that the proposed algorithm achieves similar levels of performance at 50 Hz as at 512 Hz. Both algorithm implementations are able to achieve a static error  $< 2^\circ$  and dynamic error  $< 7^\circ$  while sampling at 10 Hz. This level of accuracy may be sufficient for human motion applications though the sampling rate will limit the bandwidth of the motion that may be measured.

## 3.5 Conclusions

Orientation estimation algorithms for inertial/magnetic sensors is a mature field of research. Modern techniques [63, 59, 111] have focused on simpler algorithms that ameliorate the computational load and parameter tuning burdens associated with conventional Kalman-based approaches. The algorithm presented in this chapter employs processes similar to other complementary filters but through a novel derivation, is able to offer some key advantages:

- Computing an error based on an analytically derived Jacobian results in a significant reduction in the computation load relative to a Gauss-Newton method [63]; quantified as 109 and 248 scalar arithmetic operations per update for C code implementations of the IMU and MARG implementations respectively.
- Normalisation of the feedback error permits optimal gains to be defined based on observable system characteristics.
- Magnetic distortion compensation algorithm eliminates the need for a direction of magnetic field to be predefined by the designer.

The elimination of a predefined direction of magnetic field is an advantage over all other algorithms cited by this chapter; though this feature may be easily incorporated to other algorithms. Experimental studies have been presented for an

off-the-shelf, leading commercial unit with reference measurements obtained from a precision optical measurement system. These studies enabled the algorithm to be benchmarked and have indicated that the algorithm performs as well as the proprietary Kalman-based system; and is able to maintain this accuracy even with a full order of magnitude in reduction of sampling rate.

The algorithm presented in this chapter was published in 2011 [5] with the source code initially made available on Google Code<sup>1</sup>. The paper and its associated technical report have since totalled over 100 citations [6]. Google Code indicates over 10,000 downloads [7]. However, as the files have since been published in multiple locations the total number of downloads is assumed to be higher.

---

<sup>1</sup><http://code.google.com/p/imumargalgorithm30042010sohm/>



# **Chapter 4**

## **Kinematically redundant non-gyro IMUs**

### **4.1 Introduction**

This chapter presents two schemes of measuring the linear and angular kinematics of a rigid body using a kinematically redundant array of triple-axis accelerometers with potential applications in biomechanics. A novel angular velocity estimation algorithm is proposed and evaluated that can compensate for angular velocity errors using measurements of the direction of gravity. Analysis and discussion of optimal sensor array characteristics are provided. A damped 2 axis pendulum was used to excite all 6 DOF of the a suspended accelerometer array through determined complex motion and is the basis of both simulation and experimental studies. The relationship between accuracy and sensor redundancy is investigated for arrays of up to 100 triple axis (300 accelerometer axes) accelerometers in simulation and 10 equivalent sensors (30 accelerometer axes) in the laboratory test rig. The work also reports on the sensor calibration techniques and hardware implementation.

## 4.2 Solving inverse kinematics with accelerometers

Consider  $n$  triple-axis accelerometers fixed and aligned to a rigid body such as an orthosis on a subject's limb at known positions relative to the body coordinate frame. The position of the  $i$ 'th sensor is described by the vector  ${}^B\mathbf{d}_i$  defined in the body frame and provides the measurement vector  ${}^i\boldsymbol{\alpha}$ . The motion of the body is described by the linear accelerations  $\ddot{x}$ ,  $\ddot{y}$  and  $\ddot{z}$  of the body origin and the rotation velocities  $p$ ,  $q$  and  $r$  around the body  $x$ ,  $y$  and  $z$  axes respectively. This is represented as the schematic shown as Figure 4.1 where the mutually orthogonal unit vectors  $\hat{\mathbf{x}}_B$ ,  $\hat{\mathbf{y}}_B$  and  $\hat{\mathbf{z}}_B$ , and  $\hat{\mathbf{x}}_i$ ,  $\hat{\mathbf{y}}_i$  and  $\hat{\mathbf{z}}_i$  define the coordinated frames of the body and  $i$ 'th sensor respectively. If the  $i$ 'th sensor is not aligned to the body

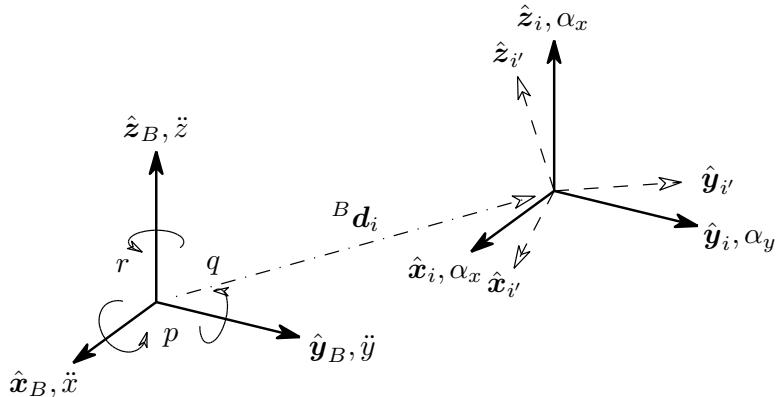


Figure 4.1: Schematic describing the  $i$ 'th accelerometer relative to the ridged body kinematic frame

frame then it can be considered aligned to the arbitrary frame  $i'$  defined by the mutually orthogonal unit vectors  $\hat{\mathbf{x}}_{i'}$ ,  $\hat{\mathbf{y}}_{i'}$  and  $\hat{\mathbf{z}}_{i'}$  (shown in Figure 4.1), providing the measurement vector  ${}^i'\boldsymbol{\alpha}$ . The orientation of frame  $i'$  relative to frame  $i$  is described by the rotation matrix  ${}_{i'}^i\mathbf{R}$  so that the aligned acceleration measurement  ${}^i\boldsymbol{\alpha}$  may be obtained using Equation 4.1.

$${}^i\boldsymbol{\alpha} = {}_{i'}^i\mathbf{R} {}^{i'}\boldsymbol{\alpha} \quad (4.1)$$

### 4.2.1 Forward kinematics

The linear acceleration measured by the  $i$ 'th sensor,  ${}^i\boldsymbol{\alpha}$ , may be defined by the kinematics of the body origin using Equation 4.2 where  ${}^B\boldsymbol{\omega}$  and  ${}^B\boldsymbol{a}$  describe the angular velocity and linear acceleration of the rigid body origin respectively. This represents the sum of linear, tangential and centripetal accelerations. There is no Coriolis term as the accelerometer position is fixed relative to the body.

$${}^i\boldsymbol{\alpha} = {}^B\dot{\boldsymbol{\omega}} \times {}^B\boldsymbol{d}_i + {}^B\boldsymbol{\omega} \times ({}^B\boldsymbol{\omega} \times {}^B\boldsymbol{d}_i) + {}^B\boldsymbol{a} \quad (4.2)$$

$${}^i\boldsymbol{\alpha} = \begin{bmatrix} \alpha_x & \alpha_y & \alpha_z \end{bmatrix}^T \quad (4.3)$$

$${}^B\boldsymbol{\omega} = \begin{bmatrix} p & q & r \end{bmatrix}^T \quad (4.4)$$

$${}^B\boldsymbol{d}_i = \begin{bmatrix} d_x & d_y & d_z \end{bmatrix}^T \quad (4.5)$$

$${}^B\boldsymbol{a} = \begin{bmatrix} \ddot{x} & \ddot{y} & \ddot{z} \end{bmatrix}^T \quad (4.6)$$

Equation 4.2 may be rearranged as Equation 4.7 where  $\mathbf{G}_i$  is a constant matrix defined by the position of the  $i$ 'th sensor and  $\mathbf{s}$  is a state vector describing the kinematics of the body origin.

$${}^i\boldsymbol{\alpha} = \mathbf{G}_i \mathbf{s} \quad (4.7)$$

$$\mathbf{G}_i = \begin{bmatrix} 0 & -d_x & -d_x & 0 & d_z & -d_y & d_y & 0 & d_z \\ \mathbf{I}_3 & -d_y & 0 & -d_y & -d_z & 0 & d_x & d_x & d_z & 0 \\ -d_z & -d_z & 0 & d_y & -d_x & 0 & 0 & d_y & d_x \end{bmatrix} \quad (4.8)$$

and

$$\mathbf{s} = \begin{bmatrix} \ddot{x} & \ddot{y} & \ddot{z} & p^2 & q^2 & r^2 & \dot{p} & \dot{q} & \dot{r} & pq & qr & rp \end{bmatrix}^T \quad (4.9)$$

## 4.2.2 Inverse kinematics solutions

### 4.2.2.1 Combinatorial method

The inverse kinematic solution can be found directly from Equation 4.7 through an inversion of  $\mathbf{G}_i$ . If the number of sensors fixed to the body is four then accelerometer measurements may be arranged as the vector  $\boldsymbol{\alpha}_C$  and the corresponding square matrix  $\mathbf{C}$  constructed as described by Equations 4.10 and 4.11.  $\mathbf{C}$  may be inverted to provide the inverse kinematic solution shown as Equation 4.12.

$$\boldsymbol{\alpha}_C = \begin{bmatrix} {}^1\boldsymbol{\alpha}^T & {}^2\boldsymbol{\alpha}^T & {}^3\boldsymbol{\alpha}^T & {}^4\boldsymbol{\alpha}^T \end{bmatrix}^T \quad (4.10)$$

$$\mathbf{C} = \begin{bmatrix} \mathbf{G}_1^T & \mathbf{G}_2^T & \mathbf{G}_3^T & \mathbf{G}_4^T \end{bmatrix}^T \quad (4.11)$$

$$\mathbf{s} = \mathbf{C}^{-1}\boldsymbol{\alpha}_C \quad (4.12)$$

This solution cannot directly incorporate  $n$  triple-axis accelerometer for  $n \neq 4$ . If  $n > 4$  then there exists  $m$  unique combinations of sensors and possible  $\mathbf{C}$  matrices where  $m$  is defined by Equation 4.13. A single value of  $\mathbf{s}$  may be computed as the mean of the  $m$  possible values, as described by Equation 4.14 where  $\mathbf{s}_j$  a value for a given combination of four triple-axis accelerometer.

$$m = \binom{n}{4} = \frac{n!}{24(n-4)!} \quad (4.13)$$

$$\bar{\mathbf{s}} = \frac{1}{m} \sum_{j=0}^m \mathbf{s}_j \quad (4.14)$$

If the positions of any sensors are equal or if all sensors exist on a plane (for a non-planar system) then  $\mathbf{G}_i$  is singular and cannot be inverted. For  $n > 4$ , a unique value of  $\mathbf{C}^{-1}$  must be computed for each combination. This method may result in a considerable computational load as an array of  $n$  sensors requires  $m$   $12 \times 12$  matrix inversions and multiplications.

#### 4.2.2.2 Pseudoinverse method

For  $n$  accelerometer measurements arranged in the vector  $\boldsymbol{\alpha}_P$ , the non-square matrix  $\mathbf{P}$  may be constructed and the inverse kinematic solution may be computed as Equation 4.17 where  $+$  denotes the pseudoinverse. This is a similar approach to that of existing inverse kinematic solutions for redundant accelerometer arrays [86, 87].

$$\boldsymbol{\alpha}_P = \begin{bmatrix} {}^1\boldsymbol{\alpha}^T & {}^2\boldsymbol{\alpha}^T & {}^3\boldsymbol{\alpha}^T & \dots & {}^n\boldsymbol{\alpha}^T \end{bmatrix}^T \quad (4.15)$$

$$\mathbf{P} = \begin{bmatrix} {}^1\mathbf{G}^T & {}^2\mathbf{G}^T & {}^3\mathbf{G}^T & \dots & {}^n\mathbf{G}^T \end{bmatrix}^T \quad (4.16)$$

$$\mathbf{s} = \mathbf{P}^+ \boldsymbol{\alpha}_P \quad (4.17)$$

The Moore-Penrose pseudoinverse of  $\mathbf{P}$  may be calculated as Equation 4.18 provided that the rows of  $\mathbf{P}$  are linearly independent and the matrix represents  $> 4$  sensors.

$$\mathbf{P}^+ = (\mathbf{P}^T \mathbf{P})^{-1} \mathbf{P}^T \quad (4.18)$$

If the positions of any sensors are equal or if all sensors exist on a plane (for a non-planar system) then the rows of  $\mathbf{P}$  are not linearly independent and Equation

4.18 may not be used. The computational requirement of this method considerably is less than that of the combinatorial method as only one  $n \times 12$  matrix inversion and multiplication is required for  $n$  sensors.

It should also be noted that the pseudoinverse is identical to an inverse computed using the non-zero singular values from a singular value decomposition of the matrix. Thus the sensitivity of the solution can be evaluated from the ratio of the largest to the smallest singular value (equivalent to the condition number of the matrix) as well as the composition of the two unitary matrices associated with the decomposition. This analysis confirms the findings presented later in Appendix A, optimal array configurations. Other matrix methods are available to solve Equation 4.17 such as QR decomposition with back substitution, or minimising the  $l_1$  norm.

### 4.2.3 Angular velocity estimation

The state vector  $\mathbf{s}$  does not directly yield individual angular velocity terms. This is a problem common to all accelerometer-only inertial measurement methods and many solutions have proposed. Cardou *et al.* provide a discussion and analysis of the different approaches to this problem and present a method more robust than many existing solutions [112].

Here we propose a novel angular velocity estimation algorithm that employs the gradient descent approach but combines this with a complementary filter structure to achieve advantages in computational efficiency and well described error dynamics. The method is similar to that described in Madgwick *et al.* [5]. The angular velocity estimation is further improved by the orientation estimation algorithm presented in Section 4.3 to compensate for bias errors in the estimated angular velocity.

The available angular velocity terms may be used to construct an objective function  $\mathbf{f}(\tilde{\boldsymbol{\omega}}, \mathbf{s})$  as shown in Equation 4.19 where  $\tilde{\boldsymbol{\omega}}$  is the estimated angular velocity. If  $\mathbf{f}(\tilde{\boldsymbol{\omega}}, \mathbf{s}) = 0$  then  $\tilde{\boldsymbol{\omega}} = \pm \boldsymbol{\omega}$ . As each angular velocity term is only available as the product of itself or another, the sign of each quantity is lost and cannot be recovered directly.

$$\mathbf{f}(\tilde{\boldsymbol{\omega}}, \mathbf{s}) = \begin{bmatrix} \tilde{p}^2 - p^2 \\ \tilde{q}^2 - q^2 \\ \tilde{r}^2 - r^2 \\ \tilde{p}\tilde{q} - pq \\ \tilde{q}\tilde{r} - qr \\ \tilde{r}\tilde{p} - rp \end{bmatrix} \quad (4.19)$$

$$\tilde{\boldsymbol{\omega}} = \begin{bmatrix} \tilde{p} & \tilde{q} & \tilde{r} \end{bmatrix}^T \quad (4.20)$$

The gradient of the solution surface (defined by  $\eta_p$ ,  $\eta_q$  and  $\eta_r$ ) created by  $\mathbf{f}(\tilde{\boldsymbol{\omega}}, \mathbf{s})$  may be used to compute the direction of the error in the estimated angular velocity

as Equation 4.23 where  $\mathbf{J}_f$  is the Jacobian of  $\mathbf{f}(\tilde{\boldsymbol{\omega}}, \mathbf{s})$ .

$$\begin{bmatrix} \eta_p \\ \eta_q \\ \eta_r \end{bmatrix} = \mathbf{J}_f^T(\tilde{\boldsymbol{\omega}}) \mathbf{f}(\tilde{\boldsymbol{\omega}}, \mathbf{s}) \quad (4.21)$$

$$= \begin{bmatrix} 2\tilde{p} & 0 & 0 & \tilde{q} & 0 & \tilde{r} \\ 0 & 2\tilde{q} & 0 & \tilde{p} & \tilde{r} & 0 \\ 0 & 0 & 2\tilde{r} & 0 & \tilde{q} & \tilde{p} \end{bmatrix}^T \begin{bmatrix} \tilde{p}^2 - p^2 \\ \tilde{q}^2 - q^2 \\ \tilde{r}^2 - r^2 \\ \tilde{p}\tilde{q} - pq \\ \tilde{q}\tilde{r} - qr \\ \tilde{r}\tilde{p} - rp \end{bmatrix} \quad (4.22)$$

$$= \begin{bmatrix} 2\tilde{p}(\tilde{p}^2 - p^2) + \tilde{q}(\tilde{p}\tilde{q} - pq) + \tilde{r}(\tilde{r}\tilde{p} - rp) \\ 2\tilde{q}(\tilde{q}^2 - q^2) + \tilde{p}(\tilde{p}\tilde{q} - pq) + \tilde{r}(\tilde{q}\tilde{r} - qr) \\ 2\tilde{r}(\tilde{r}^2 - r^2) + \tilde{q}(\tilde{q}\tilde{r} - qr) + \tilde{p}(\tilde{r}\tilde{p} - rp) \end{bmatrix} \quad (4.23)$$

The direction of this vector may be combined with a magnitude computed as the difference between the magnitudes of the estimated angular velocities and actual angular velocities. The error in the angular velocity is therefore estimated,  $\tilde{\mathbf{e}}$ , as the dot product shown in Equation 4.24. Due to the loss of sign in the available angular velocity terms the error will be computed relative to the closest value of  $\pm \boldsymbol{\omega}$ .

$$\tilde{\mathbf{e}} = \begin{bmatrix} \text{sign}(\eta_p) \\ \text{sign}(\eta_q) \\ \text{sign}(\eta_r) \end{bmatrix} \cdot \begin{bmatrix} |\tilde{p}| - \sqrt{p^2} \\ |\tilde{q}| - \sqrt{q^2} \\ |\tilde{r}| - \sqrt{r^2} \end{bmatrix} \quad (4.24)$$

The estimated angular velocity is computed from the angular acceleration and estimated error in the angular velocity using Equation 4.25. This is represented as the block diagram shown as Figure 4.2.

$$\tilde{\boldsymbol{\omega}} = \int (\boldsymbol{\dot{\omega}} - K\tilde{\mathbf{e}}).dt \quad (4.25)$$

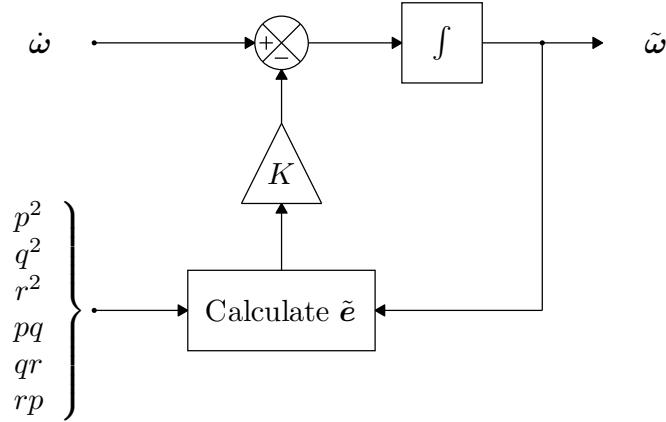


Figure 4.2: Complementary filter used for angular velocity estimation

#### 4.2.3.1 Stability and response to initial conditions

A simplified analysis of the proposed complementary filter is possible if the estimated error term,  $\tilde{e}$ , is assumed to be equal to the true error term,  $e$ . The complementary filter can then be seen to represent a first order system with error dynamics governed by Equation 4.26. The system is therefore stable for all values of  $K > 0$  and the error can be expected to decay exponentially at a rate governed by  $K$ .

$$\dot{e} = -Ke \quad (4.26)$$

As the estimated error may only be computed relative to the closest value of  $\pm\omega$ ,  $\tilde{\omega}$  will converge to the closest value of either  $\pm\omega$ . A correct sign assertion may be assured if the dynamics in  $\dot{\omega}$  are of a sufficiently large magnitude relative to the convergence rate of  $\tilde{\omega}$  governed by  $K$ , at the zero-crossing point of  $\omega$ . A sufficient magnitude of  $\dot{\omega}$  relative to the convergence rate of  $\tilde{\omega}$  would mean that integral drift would not dominate the  $\tilde{\omega}$  dynamics and the value of  $\tilde{\omega}$  would not drift across zero and hence cause an incorrect sign.

## 4.3 Orientation estimation and angular velocity drift compensation

In many applications it can be assumed that the mean linear acceleration measured over an extended period of time represents gravity. The direction of gravity measured relative to the accelerometer array may therefore be used to estimate the attitude of the array relative to the Earth's surface. As will be shown in Section 4.6.3, sensor measurement and alignment errors result in a bias error in the estimated angular velocity at low velocities. Knowledge of the accelerometer array orientation may be used to correct for steady state errors in the estimation angular velocity and so compensate for integral drift.

Mahony *et al.* [59] have presented an orientation estimation algorithm for conventional IMUs consisting of gyroscopes and accelerometers. The algorithm is equally applicable to the accelerometer array presented in this work as the inverse kinematic solution yields linear accelerations and angular velocities. Equations 4.27 to 4.30 represent a quaternion implementation [62] of this algorithm where  ${}^B_E\dot{\mathbf{q}}$  is the estimated orientation of the Earth relative to the body, represented as a quaternion of elements  $q_w$  to  $q_z$ . The gains  $K_P$  and  $K_I$  represent proportional and integral gains respectively. The  $\hat{\cdot}$  accent denotes the normalised unit vector and the  $\otimes$  operator denotes a quaternion product.

$${}^B_E\mathbf{q} = \int {}^B_E\dot{\mathbf{q}} \quad (4.27)$$

$${}^B_E\dot{\mathbf{q}} = \frac{1}{2} {}^B_E\hat{\mathbf{q}} \otimes \begin{bmatrix} 0 & ({}^B\boldsymbol{\omega} - \boldsymbol{\delta})^T \end{bmatrix} \quad (4.28)$$

$$\boldsymbol{\delta} = K_P \boldsymbol{\epsilon} + K_I \int \boldsymbol{\epsilon} \quad (4.29)$$

$$\boldsymbol{\epsilon} = {}^B\boldsymbol{a} \times \begin{bmatrix} 2q_x q_z - q_w q_y \\ 2q_w q_x + q_y q_z \\ q_w^2 - q_x^2 - q_y^2 + q_z^2 \end{bmatrix} \quad (4.30)$$

The error in the estimated angular velocity is represented by the integral feedback term. The compensated estimated angular velocity is therefore obtained as Equation 4.31.

$$\tilde{\boldsymbol{\omega}}' = \tilde{\boldsymbol{\omega}} - K_i \int \boldsymbol{\epsilon} \quad (4.31)$$

## 4.4 2 axis pendulum kinematics

A 2 axis pendulum was the chosen subject of simulation and experimental studies as it approximates a number of human and robotic manipulation joints and the oscillatory motion is analogous to gait and excites all 6 DOF of the suspended body. The damped motion results in both low and high magnitude rates and the complex motion of the 2 axis joint results in rotational kinematics in all three axes of the suspended body. The pendulum is represented by the schematic shown as Figure 4.3 where  $\theta_1$  and  $\theta_2$  represent the angle in each axis of the joint. The 2 axis pendulum equations of motion have been derived as Equations 4.32 and 4.33 where  $c_1$  and  $c_2$  are the damping of each axis of the joint.

$$\ddot{\theta}_1 = 2\dot{\theta}_1\dot{\theta}_2 \tan \theta_2 - \frac{g \sin \theta_1}{l \cos \theta_2} - \frac{c_1 \dot{\theta}_1}{m \cos^2 \theta_2} \quad (4.32)$$

$$\ddot{\theta}_2 = -\dot{\theta}_1^2 \cos \theta_2 \sin \theta_2 - \frac{g}{l} \cos \theta_1 \sin \theta_2 - \frac{c_2 \dot{\theta}_2}{m} \quad (4.33)$$

The body kinematics are computed as the forward kinematics described by Equations 4.34 to 4.37 where  ${}^B\boldsymbol{a}$ ,  ${}^B\boldsymbol{\omega}$  and  ${}^B\dot{\boldsymbol{\omega}}$  describe the translational

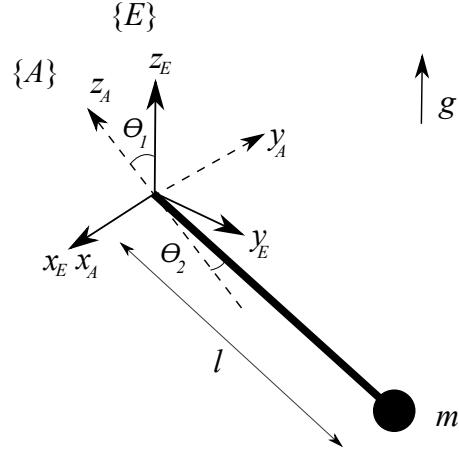


Figure 4.3: 2 axis pendulum schematic

acceleration, angular velocity and angular acceleration respectively of the body frame in directions relative to the body frame.  ${}^E_B \mathbf{R}$  describes the orientation of the body frame relative to the Earth frame.

$${}^B \mathbf{a} = g \begin{bmatrix} -\sin \theta_2 \\ \sin \theta_1 \cos \theta_2 \\ \cos \theta_1 \cos \theta_2 \end{bmatrix} \quad (4.34)$$

$${}^B \boldsymbol{\omega} = \begin{bmatrix} \dot{\theta}_1 \cos \theta_2 \\ \dot{\theta}_2 \\ \dot{\theta}_1 \sin \theta_2 \end{bmatrix} \quad (4.35)$$

$${}^B \dot{\boldsymbol{\omega}} = \begin{bmatrix} \ddot{\theta}_1 \cos \theta_2 - \dot{\theta}_1 \dot{\theta}_2 \sin \theta_2 \\ \ddot{\theta}_2 \\ \ddot{\theta}_1 \sin \theta_2 + \dot{\theta}_1 \dot{\theta}_2 \cos \theta_2 \end{bmatrix} \quad (4.36)$$

$${}^E_B \mathbf{R} = \begin{bmatrix} \cos \theta_2 & 0 & \sin \theta_2 \\ \sin \theta_1 \sin \theta_2 & \cos \theta_1 & -\sin \theta_1 \cos \theta_2 \\ -\cos \theta_1 \sin \theta_2 & \sin \theta_1 & \cos \theta_1 \cos \theta_2 \end{bmatrix} \quad (4.37)$$

## 4.5 Simulation studies

Simulation studies were used for the initial testing of the proposed methods and to investigate the relationship between system accuracy and redundancy. Simulations used the damped 2 axis pendulum model with an array of  $n$  triple-axis accelerometers positioned evenly around a  $360^\circ$  helix concentric with the length of the pendulum and with a radius of 0.05 m. Figure 4.4 shows the pendulum and triple-axis accelerometers as drawn by the simulation software. For convenience, all sensors were aligned with the pendulum body coordinate frame; arbitrary orientations were used and calibrated for in the experimental studies discussed in Section 4.6.

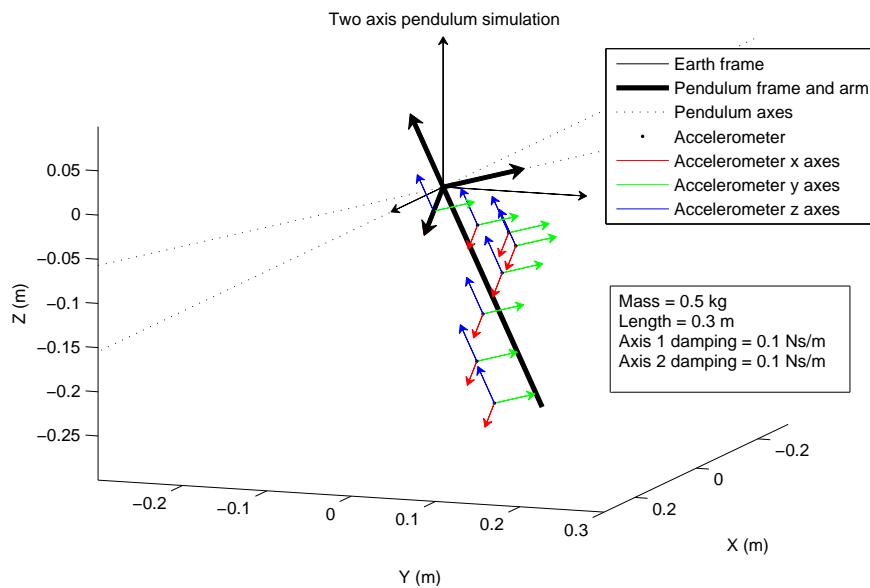


Figure 4.4: Schematic of pendulum with triple-axis accelerometers mounted in helix pattern. Graphics generated by simulation software

The forward kinematics were computed to provide ideal sensor accelerometer measurements and processed through a sensor model to simulate the frequency response and noise characteristics. The sensor model was a simplification of a Simulink based accelerometer behavioural model provided by the manufacturer (Analog Devices). The noise model consisted of zero mean additive Gaussian noise

with a standard deviation of  $0.045\text{ms}^{-2}$  chosen to match the experimental data. The signal was then processed through a first order low-pass filter with a corner frequency of 50 Hz. The angular velocity estimation filter used a fixed gain of  $K = 0.5$ . Pendulum model parameters were fixed so that  $m = 0.5 \text{ kg}$ ,  $l = 0.3 \text{ m}$ ,  $g = 9.8 \text{ ms}^{-2}$ ,  $c_1 = 0.1 \text{ Nsm}^{-1}$  and  $c_2 = 0.3 \text{ Nsm}^{-1}$ , with initial conditions of  $\theta_1 = \theta_2 = 60^\circ$ , and  $\dot{\theta}_1 = \dot{\theta}_2 = 0$ . The simulation was run for 30 seconds with a discrete-time step of 1 ms.

To investigate the relationship between the redundancy and accuracy of a sensor array, simulations were conducted for arrays consisting of between 5 and 100 triple-axis sensors. For each array the mean of the RMS errors in the  $x$ ,  $y$  and  $z$  dimensions was computed for the linear acceleration, angular acceleration and angular velocity. The error in the estimated orientation was not investigated as integral drift would fail to represent a meaningful relationship. Results were obtained using both the combinatorial and pseudoinverse methods to enable comparison.

The integral drift inherent to the angular velocity estimation algorithm would introduce a bias in the estimated velocity at low rates. This random steady state error meant that the apparent relationship between the angular velocity error and redundancy was non-monotonic. To compensate for this, the mean result of 5 simulations was computed for each value of redundancy. The results for angular acceleration and velocity are shown as Figure 4.5 and Figure 4.6. It can be seen that the combinatorial method results in a greater error than the pseudoinverse method at greater computational expense. Simulations of the combinatorial method for arrays  $> 33$  sensors were abandoned as the time taken to process arrays of this size became impractical. The non-monotonic relationship observed in the estimated angular velocity is due to the ‘random walk’ in this state that comes about at low velocities and is constrained by the feedback in the angular velocity estimation filter. This ‘random walk’ is due to the noise

(in simulation) of the Analog Devices accelerometers. As the angular velocity is reduced (due to damping), noise dominates the signal and the estimated angular velocity is then largely governed by the integration of the angular acceleration which again is largely made up of the sensor noise causing a random walk. This error does not directly relate to the number of sensors.

Oscillatory motion of the pendulum and the frequency response of the sensors results in a minimum achievable error represented by the oscillatory error components due to the phase-lag of the sensor outputs (due to a simulated anti-aliasing filter). This minimum achievable error was computed by running a simulation using an accelerometer error model comprised of only the anti-aliasing filter, i.e. zero noise. The resultant error is the same for arrays of all sizes and represents the minimum achievable error. The respective minimum achieve errors are indicated in Figure 4.5 and Figure 4.6.

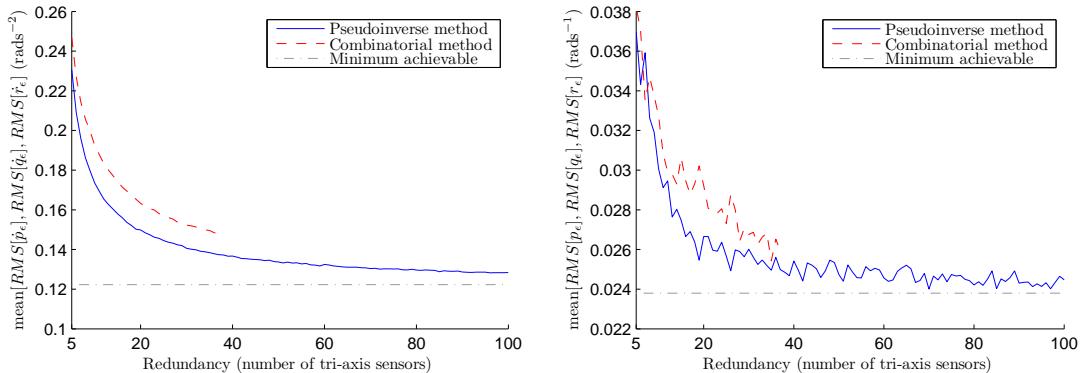


Figure 4.5: Angular acceleration error vs. sensor redundancy

Figure 4.6: Angular velocity error vs. sensor redundancy

The pendulum testbed allows more complex motion to be evaluated. If the initial conditions use a non-zero velocity vector and dissimilar damping coefficients, all cross terms in Equation 4.7 can be elicited at a range of frequencies. The spectrum of any individual accelerometer axis will contain frequencies up to the natural frequency of the pendulum and hence the results can be demonstrated to work across this frequency range. The upper graphs in Figures 4.7 and 4.8 demonstrate

this result for a simulation of a cluster of 20 accelerometers placed in a 5 cm radius helix beginning half way down the pendulum and continuing to the end. When viewed in isolation this is readily observed to be a non obvious complex motion.

The error residuals are given in the lower graphs of Figures 4.7 and 4.8.

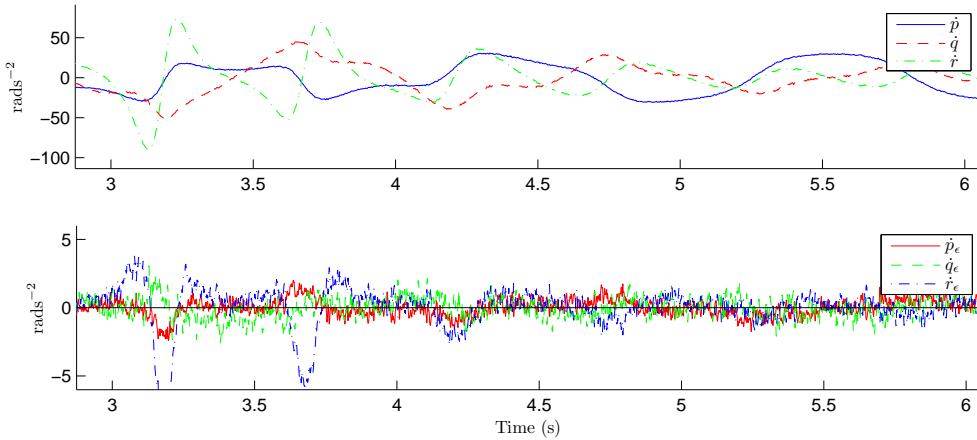


Figure 4.7: Angular accelerations of pendulum measured by accelerometer array (top) and residuals (bottom)

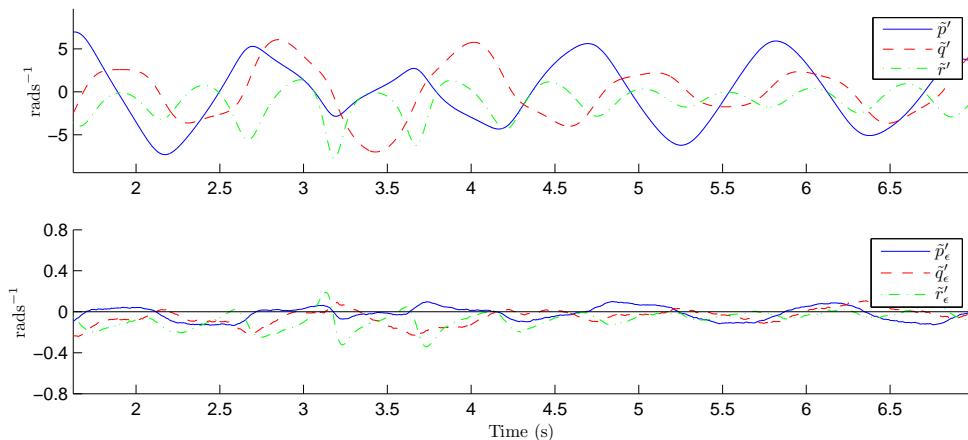


Figure 4.8: Estimated angular velocity compensated by orientation estimation algorithm (top) and residuals (bottom)

## 4.6 Experimental studies

Experimental studies were conducted to verify the above simulation results. The studies were designed to be as similar to the simulation studies as possible and

used a redundant array of 10 triple-axis accelerometers. Unlike the simulation studies, the experimental studies were subject to 2 significant sources of error: sensor calibration errors (of gains and biases) and sensor installation errors (of position and alignment).

### 4.6.1 Hardware

Experimental studies required data to be acquired from an array of 10 triple-axis accelerometers at a sample rate of at least 1 kHz (to match simulation investigations) and with as high a resolution as practicable; sensor resolution had been identified as a limiting factor in similar studies [71]. To meet these requirements, custom equipment was developed for this research project, consisting of a DAQ32 data acquisition board PC Interface software and 10 triple-axis accelerometers.

#### 4.6.1.1 Data acquisition equipment

The DAQ32 is an open-source<sup>1</sup> 32 channel Universal Serial Bus (USB) data acquisition board built around Microchip's high-end dsPIC33FJ256GP710 Digital Signal Controller. The DAQ32 features a filtered power supply for analogue sensors and 32 analogue input channels each sampled at 1.024 kHz. Each channel is oversampled by a factor of 8 and the truncated mean result used for enhanced filtering and accuracy.

#### 4.6.1.2 Accelerometer probes

Each accelerometer probe consisted of an Analog Devices' ADXL335 triple-axis accelerometer with signal and power supply conditioning circuitry assembled on

---

<sup>1</sup><https://github.com/xioTechnologies/DAQ32>

the accelerometer package to minimise size and weight. The probe's 0.4 m lead consisted of wires twisted together so that the power supply wires would provide a level of electromagnetic shielding to the accelerometer signals. Fine wire connections were sunk in a silicon sealant to provide mechanical durability. The complete probe (shown in Figure 4.9) occupied a volume of a 5 mm cube and weighed less than 1 g (excluding lead).

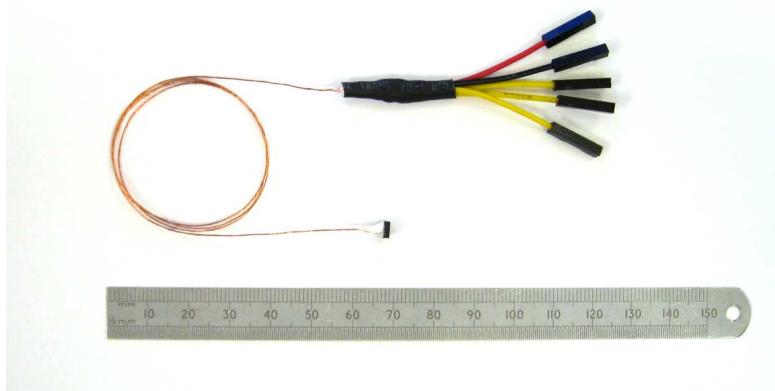


Figure 4.9: Complete accelerometer probe next to a 150 mm ruler

#### 4.6.1.3 Test rig

The pendulum, shown in Figure 4.10, was constructed from 2 perpendicular sheets of aluminium so that the 10 accelerometer probes on the surface of the sheets formed an approximate helix with a length of 350 mm and radius of 50 mm. Each accelerometer was mounted at arbitrary orientations varying up to  $180^\circ$ . The position of each accelerometer was measured. The pendulum joint was constructed from 2 precision servo potentiometers so that angular position of each axis could be measured. The 32 analogue channels required by the 10 triple-axis accelerometer probes and 2 potentiometers were all accommodated by the DAQ32. Figure 4.10 shows the pendulum with attached accelerometer probes and annotated potentiometer axes  $\theta_1$  and  $\theta_2$ .

Simulation was based on a helix, the experimental rig needed to be as stiff as possible so a cross structure was chosen. This is equivalent to a four-spoke helix. This is a reasonable compromise to allow the experimental rig to be manufactured. The cross structure in the pendulum allows for greater stiffness in the pendulum and higher accuracy when placing the sensors.

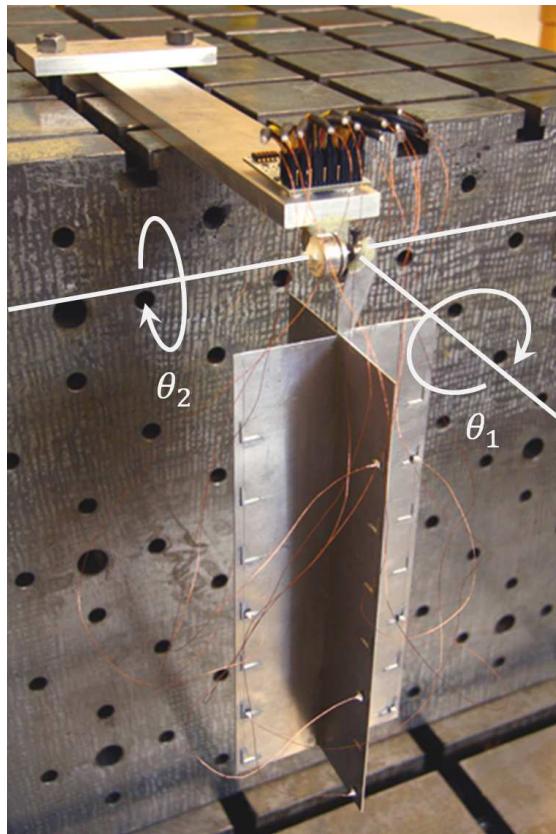


Figure 4.10: 2 axis pendulum with attached accelerometer probes and annotated potentiometer axes  $\theta_1$  and  $\theta_2$

## 4.6.2 Calibration

### 4.6.2.1 Axis gain and bias

Assuming the accelerometer output to be linear, the calibrated measurement,  $\alpha$ , may be obtained from the un-calibrated measurement,  $u$ , using the relationship described by 4.38. The vector  $b$  and diagonal matrix  $K$  account for the

individual axis bias and sensitivity respectively. The skew matrix  $\mathbf{S}$  accounts for non-orthogonality and coupling between the accelerometer axes. The gain  $s_{mn}$  defines the proportion of axis  $n$  affecting axis  $m$ .

$$\begin{aligned} \boldsymbol{\alpha} &= \mathbf{SK}\mathbf{u} - \mathbf{b} \\ &= \begin{bmatrix} 1 & s_{xy} & s_{xz} \\ s_{yx} & 1 & s_{yz} \\ s_{zx} & s_{zy} & 1 \end{bmatrix} \begin{bmatrix} k_x & 0 & 0 \\ 0 & k_y & 0 \\ 0 & 0 & k_z \end{bmatrix} \begin{bmatrix} u_x \\ u_y \\ u_z \end{bmatrix} - \begin{bmatrix} b_x \\ b_y \\ b_z \end{bmatrix} \end{aligned} \quad (4.38)$$

A stationary accelerometer will measure the direction and magnitude of the Earth's gravitational field due to the reaction force on the mass within the MEMS structure. For a calibrated sensor, the measured magnitude will be constant at all orientations of the sensor. That is, Equation 4.39 will be true for all said measurements where  $g$  is the magnitude of gravity and the sensor is stationary. Therefore,  $\mathbf{K}$  and  $\mathbf{b}$  may be found as the solution to Equation 4.40 where  $\mathbf{u}_j$  represents the un-calibrated sensor measurement at the  $j$ 'th orientation within a calibration dataset. Although it is possible to compound  $\mathbf{S}$  and  $\mathbf{K}$  into a single matrix, it was found this would limit the success of an optimisation algorithm as it became sensitive to the initial guesses of the gain parameters.

$$g = \|\mathbf{SK}\mathbf{u} - \mathbf{b}\| \quad (4.39)$$

$$\min_{\mathbf{S}, \mathbf{K}, \mathbf{b} \in \Re} \sum_j (g - \|\mathbf{SK}\mathbf{u}_j - \mathbf{b}\|)^2 \quad (4.40)$$

A calibration dataset was collected using an automated calibration gimbal capable of rotating independently around the pan, roll and tilt axes. The pendulum (with attached accelerometer probes) was fixed to the gimbal so that it could be rotated to any orientation. An automated routine was composed to rotate the gimbal

pitch and roll axes through their full range in  $10^\circ$  steps so that every combination of pitch and roll angle was achieved. At each orientation, the gimbal was allowed 10 seconds to come to rest and then the mean of each accelerometer output was taken for a 15 second period.

A value of  $g$  was obtained as  $9.812 \text{ ms}^{-2}$  for the geographical location of the studies [113]. Equation 4.40 was solved using MATLAB's Optimisation toolbox's `fminunc` function and the Broyden-Fletcher-Goldfarb-Shanno (BFGS) method. Initial values for gains and biases were chosen as the typical values detailed in the AXDL335 datasheet. Calibration was verified by comparing the measured magnitude of gravity with the known magnitude at each orientation within the calibration dataset. Figure 4.11 shows typical calibrated measurements of one accelerometer for the full calibration dataset.

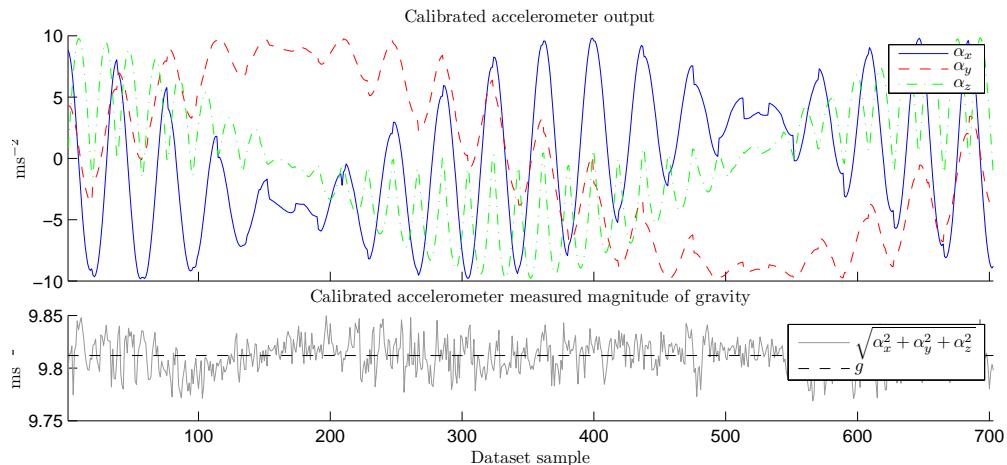


Figure 4.11: Validation the calibrated accelerometer gains and biases

#### 4.6.2.2 Alignment

Hardware installation practicalities mean that misalignment between the accelerometers and the pendulum coordinate frame must be accounted for in each triple-axis accelerometer. Alignment errors as low as  $0.1^\circ$  have been shown to result in significant errors in estimated kinematic states [86]. An aligned triple-axis

accelerometer measurement,  ${}^i\boldsymbol{\alpha}$ , is related to the misaligned measurement  ${}^{i'}\boldsymbol{\alpha}$  by the rotation matrix  ${}_{i'}\mathbf{R}$  as described by Equation 4.1. The alignment may be equivalently represented by the quaternion  ${}_{i'}\mathbf{q}$ . If the array is stationary, Equation 4.41 will be true for sensors at all orientations. Therefore  ${}_{i'}\mathbf{q}$  may be found as the solution to 4.42 where  ${}^i\boldsymbol{\alpha}_j$  and  ${}^B\mathbf{a}_j$  represent the measured acceleration of the  $i$ 'th accelerometer and the acceleration of the body at the  $j$ 'th orientation respectively. As the acceleration of the body is unknown, one fiducial sensor must be chosen to provide a measurement equal to  ${}^B\mathbf{a}$ . An alternative kinematic minimisation approach for alignment calibration was also implemented but performance was found to be no better than simply choosing a convenient sensor to be the fiducial.

$${}^B\mathbf{a} = {}_{i'}\hat{\mathbf{q}}^* \otimes {}^i\boldsymbol{\alpha} \otimes {}_{i'}\hat{\mathbf{q}} \quad (4.41)$$

$$\min_{{}_{i'}\mathbf{q} \in \Re} \sum_j \left[ {}_{i'}\hat{\mathbf{q}}^* \otimes {}^i\boldsymbol{\alpha}_j \otimes {}_{i'}\hat{\mathbf{q}} - {}^B\mathbf{a}_j \right]^2 \quad (4.42)$$

Equation 4.42 was solved using the same calibration dataset and method described in Section 4.6.2.1. The fiducial sensor was chosen as that judged to be best aligned to the pendulum on visual inspection. This gravitational based attitude calibration scheme is similar to previously proposed techniques; however, the use of a quaternion representation avoids problems of non-orthogonality associated with a rotation matrix based approach [114].

### 4.6.3 Results

The pendulum was left to swing with motion in both axes remaining in phase until rest while data from each of the 10 triple-axis accelerometers and 2 joint potentiometers was recorded. The ‘true’ suspended body kinematics were then estimated for the first 30 seconds of the dataset using the potentiometer data and

kinematic model within a moving horizon state estimation algorithm.

Figure 4.12 and Figure 4.13 show the pendulum kinematic states as estimated using the pseudoinverse method and data from all 10 sensors. The angular velocity estimation algorithm used a gain of  $K = 2$  and the orientation estimation algorithm used gains of  $K_P = 5$  and  $K_I = 1.0$ . Each figure consists of 2 plots: the upper plot represents the measured or estimated state and the lower plot represents the corresponding error. Small errors in alignment calibration and the constant acceleration of gravity result in an angular velocity bias error and low velocities (see Figure 4.13). The orientation estimation algorithm described in Section 4.3 compensates for this error in  $p$  and  $q$  but errors  $r$  cannot be compensated for as this axis of rotation eventually aligns with the direction of gravity.

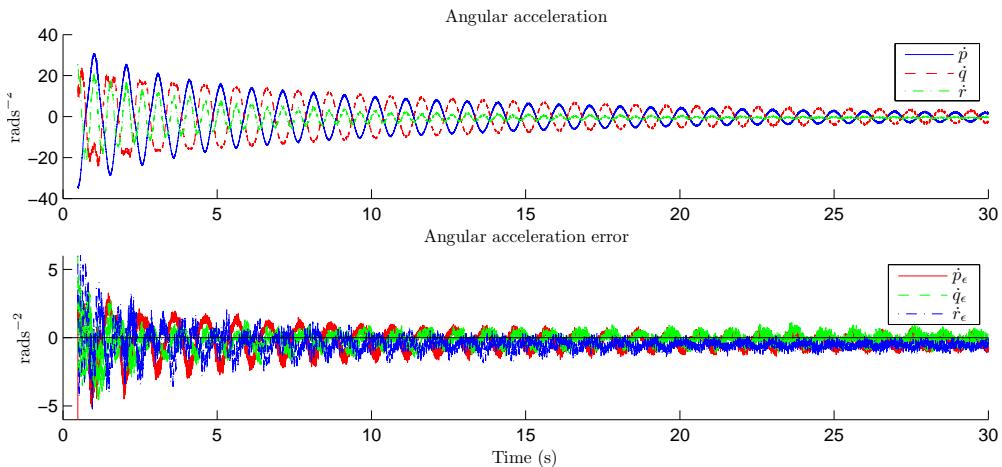


Figure 4.12: Angular accelerations of pendulum measured by accelerometer array (top) and residuals (bottom)

To investigate the relationship between error and redundancy, the suspended body kinematics were computed using the pseudoinverse method for each combination of sensors for the available array of ten sensors, thus results from four to ten sensors were possible. Since the number of combinations for a given number of sensors is defined by  $\binom{10}{n}$ , 210 results were possible for 4 sensors, rising to 252 for 5 sensors and then falling to give only the single result when the fully redundant array of 10 sensors was computed. Performance is quantified as the mean of the

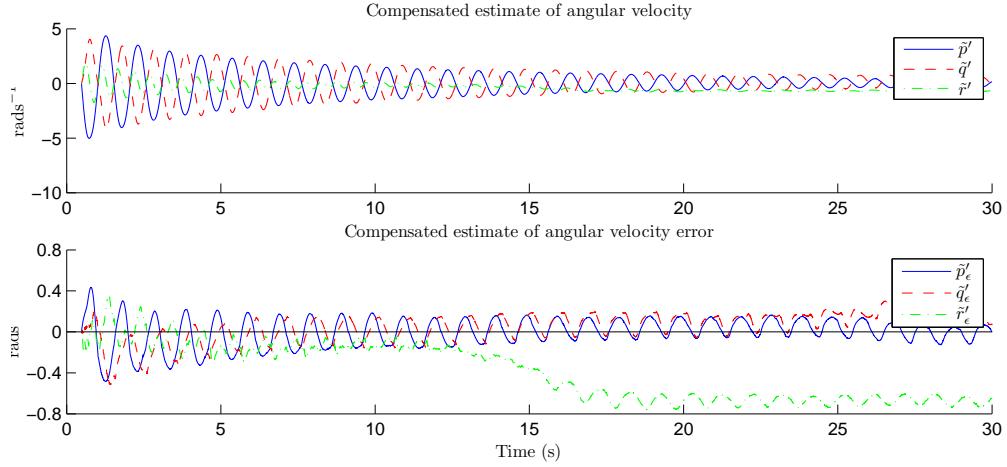


Figure 4.13: Estimated angular velocity compensated by orientation estimation algorithm (top) and residuals (bottom)

RMS error in the  $x$ ,  $y$  and  $z$  DOF for linear acceleration, angular acceleration and angular velocity. These results are shown as box plots in Figure 4.14. The results compare well with the simulation studies given in Section 4.5. The anomalous rise apparent in the third graph of Figure 4.14 when all 10 sensors are used is an artefact of the imperfect alignment of sensors causing the information of one sensor to dominate the calculation. This would suggest a practical system could make a confidence estimate of individual sensors and down grade sensors that might be close to saturation, become mechanically misaligned (or knocked), or operating at a low signal to noise ratio.

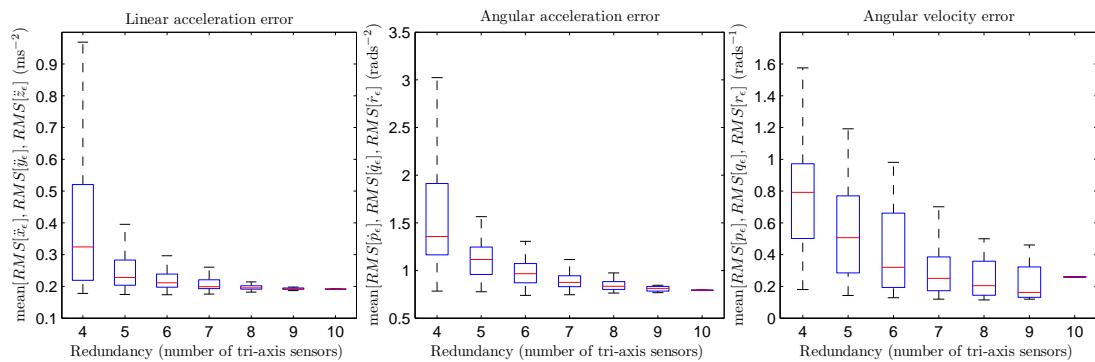


Figure 4.14: Relationship between error and sensor redundancy for linear acceleration (left), angular acceleration (middle), angular velocity (right). Whiskers indicate  $1.5 \times$  interquartile range.

## 4.7 Discussion

This work has shown a clear advantage of higher levels of redundancy. Incorporating a large number of sensors requires a practical consideration of the computational load. The pseudoinverse method has shown not only to provide greater accuracy but to do so at significantly less computational expense. As the matrix inversion is required to be computed only once, the computational load of the pseudoinverse is proportional to the number of sensors in the array.

Due to sensor alignment errors and the constant acceleration of gravity, the estimate of angular velocity  $r$  was subject to a bias error at low velocities ( $< 0.5 \text{ rads}^{-1}$ ). Observations of the direction of gravity enabled the orientation estimation algorithm to compensate for such errors in  $p$  and  $q$ ; however, compensation is not possible when the axis of  $r$  is parallel with the gravity vector. Such compensation may be achieved if the algorithm is extended to incorporate information from a magnetometer [59]. An accelerometer array cannot be used to determine directly the direction of an angular velocity. The angular velocity estimation algorithm presented can potentially assert an incorrect sign when angular velocities and accelerations are both of low magnitude. The velocity estimation method appeared to be robust in dealing with this problem although a sign switch is possible.

An accelerometer array IMU presents a number of benefits to a biomedical application. Unlike a conventional IMU it is able to provide direct measurement of angular acceleration which may provide new insight to movement studies. Redundancy provides increased levels of accuracy, potentially allowing a system to be created where a single (more expensive) high quality sensor may be replaced by many (cheaper) lower quality sensors.

Calibration has been limited to being off-line using a specific calibration dataset

so that use of accelerometer arrays is limited to ridged structures where the orientation and location of each accelerometer is fixed. This is immediately applicable to robotics and avionics applications where the sensor array can be installed to an ridged chassis. If future work can achieve an on-line calibration method capable of estimating changes in individual accelerometer locations and orientations using the natural motion of the subject, then use of accelerometer arrays can be extended to non-rigid bodies. This would be of interest to biomedical applications as it would allow the use of accelerometers woven into a wearable fabric. In the interim, sensors could be embedded within a rigid orthosis such as an angle foot orthosis, or onto rigid clothing inserts. In either case, the resulting sensor will allow continuous, low power and accurate measurements with minimum obstruction to patient movements and lifestyle.

This work has demonstrated how redundancy in an array of accelerometers can improve measurement accuracy. Redundancy of such homogeneous arrays opens avenues for further exploration on object dynamics, in particular considering relaxing the constraints by considering multiple sensors attached to an elastic object, such as skin, or constrained to move in a complex relationship (such as clothing), or subject to vibration modes, (such as an aeroplane wing) or included a segmentation such as biomechanical joint was included as a constraint.

## 4.8 Conclusion

This work analyses the benefits of using large arrays of accelerometers. Results include both simulation studies of up to 100 tri-axial sensors and empirical results of an array of up to 10 MEMS accelerometers. A comparison is given between simulation and empirical data based on a 2 axis pendulum simulating a knee.

Two methods are reported for the reconstruction of an acceleration state vector

to represent the limb. Both methods allow the pre-computation of the necessary matrix inverse. The method based on computing a pseudoinverse (using singular value decomposition) is shown to achieve better accuracy than that based on combinatorial averaging of the sensors, and at less computational expense.

A limitation of accelerometer arrays is that it is not possible to recover the sign of the angular velocities directly so a method is given that tracks velocity transitions through this zero transition. Finally, the work gives a method for assessing the optimal placement of sensors in a large planar array, describes a hardware solution and discusses a method for calibrating large arrays of accelerometer sensors.



# **Chapter 5**

## **Characterisation of low-cost MEMS sensors**

### **5.1 Introduction**

The objective of this project is to demonstrate how new applications can be realised using modern low-cost MEMS gyroscopes, accelerometers and magnetometers. A key part of achieving this is the development of calibration solutions so that each sensor is able to provide precise measurements with a determined accuracy. For example, it is shown in Chapter 6 that calibration of the accelerometer bias and sensitivity may reduce orientation measurement errors from  $8^\circ$  to  $<0.5^\circ$ .

Low-cost MEMS are typically targeted towards consumer electronics that do not demand a high level of accuracy. A competitive market means that many modern devices are supported by the manufacturer for only a few years before being superseded by the next generation. As a result, detailed information about sensor characteristics and performance is often unavailable. It was therefore necessary for this project to determine significant sensor characteristics as a first step towards the development of calibration solutions. This chapter presents the methods and

findings of the sensor characterisations studies that form the basis of the calibration solutions presented in the next chapter.

The investigations presented in this the chapter focus on the sensor performance over an operating temperature range or at a fixed temperature. For gyroscopes and accelerometers, this requires precise controlled motion while simultaneously regulating the device temperature. Ideally, this would be achieved using an industrial three-axis thermal chamber. For example, the Ideal Aerosmith 2003HP shown in Figure 5.1 is able to achieve an accuracy in angular velocities of 0.0001% and orientations of  $\pm 1$  arcsecond ( $\sim 0.00027^\circ$ ) for temperatures between  $-65^\circ\text{C}$  to  $+85^\circ\text{C}$  [115]. Temperature is regulated using electric heating and Liquid Nitrogen ( $\text{LN}_2$ ) or Carbon Dioxide ( $\text{CO}_2$ ) cooling.



Figure 5.1: Ideal Aerosmith 2003HP three-axis thermal chamber of achieve precise velocities and orientations for temperatures between  $-65^\circ\text{C}$  to  $+85^\circ\text{C}$ . Image source: [115]

Although such industrial equipment would facilitate a comprehensive characterisation of gyroscopes and accelerometers, the hire cost made it inaccessible to this project [116]. Similar characterisation studies presented in the subject literature take advantage of such equipment [117, 28, 118]. The investigations presented in this chapter are achieved using improvised solutions.

### 5.1.1 Devices selected for characterisation

Several different gyroscopes, accelerometers and magnetometers from a range of manufacturers were used throughout this project. This chapter presents the characterisation of the two devices that would ultimately be incorporated to the IMU platforms described in Chapter 8. The devices are the InvenSense MPU-6050 combined triple-axis gyroscope ( $\pm 2000^\circ/\text{s}$  range, 16-bit resolution) and accelerometer ( $\pm 16 \text{ g}$  range, 16-bit resolution), and the Honeywell HMC5883L triple-axis magnetometer ( $\pm 8 \text{ Gauss}$  range, 12-bit resolution). Both devices incorporate an Analogue-to-Digital Converter (ADC) and provide measurements in units of least-significant bit (lsb) via Inter-Integrated Circuit ( $\text{I}^2\text{C}$ ) communication bus. The MPU-6050 also incorporates an on-board thermometer for accurate measurement of the sensor temperature.

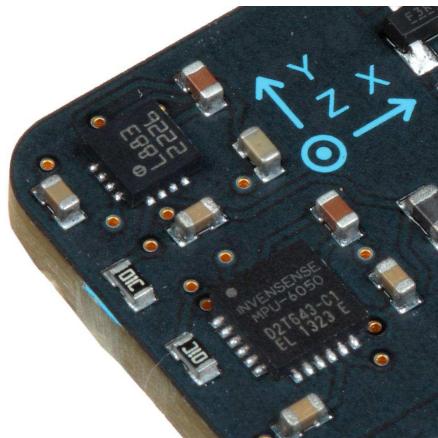


Figure 5.2: MPU-6050 (top IC) and HMC5883L (bottom IC) mounted on the x-IMU PCB. The larger IC is  $4 \times 4 \text{ mm}$  in size.

Investigations were conducted using prototypes of the x-IMU and x-BMIU platforms presented in Chapter 8. Both platform incorporates the MPU-6050 and HMC5883L along side wireless communication or Secure Digital (SD) card logging to facilitate data collection. Figure 5.2 shows the MPU-6050 and HMC5883L mounted on a Printed Circuit Board (PCB).

### 5.1.2 Scope of investigations

Standards such as IEEE 1431-2004 [119] and 1293-1998 [120] offer an extensive list of gyroscope and accelerometer characteristics that may be quantified. The investigations presented in this chapter are limited to the characteristics indicated as being most significant in the device datasheets [121, 122] with a particular focus on the gyroscope bias. The scope of the investigations is summarised by the following.

- Gyroscope bias temperature sensitivity
- Gyroscope bias acceleration sensitivity
- Gyroscope bias random walk
- Gyroscope sensitivity and bias thermal response
- Accelerometer and magnetometer thermal response
- Sensitivity and bias variation between parts for gyroscope, accelerometer and magnetometer.

The following sections present the methods and findings of each investigation in the order above.

## 5.2 Gyroscope bias thermal response

The MPU-6050 datasheet specifies the gyroscope bias “variation over temperature” as  $\pm 20^\circ/\text{s}$  over  $-40^\circ\text{C}$  to  $80^\circ\text{C}$  [121]. Further information about characteristic, such as linearity or repeatability, is not specified in the datasheet or any other available documentation. The absence of such information about this potentially very large measurement error exemplifies the necessity for these characterisation studies. The gyroscope bias thermal response was characterised for eight devices by logging

the gyroscope output when stationary within a thermal chamber to expose the gyroscope to a controlled temperature range.

### 5.2.1 Thermal chamber

A thermal chamber was designed to provide a stable temperature ascent from -20°C to 80°C using a thermal mass and active heating. The chamber was constructed from a polystyrene box originally intended to enclose cold goods during transportation. Figure 5.3 shows the chamber internal components including: a 41 m<sup>3</sup>/hour fan, 1 kg block of copper with aluminium fins, 20 W fan heater and a thermocouple probe. The prototype x-IMU being tested is positioned in the centre and interfaced externally via a wired connection.

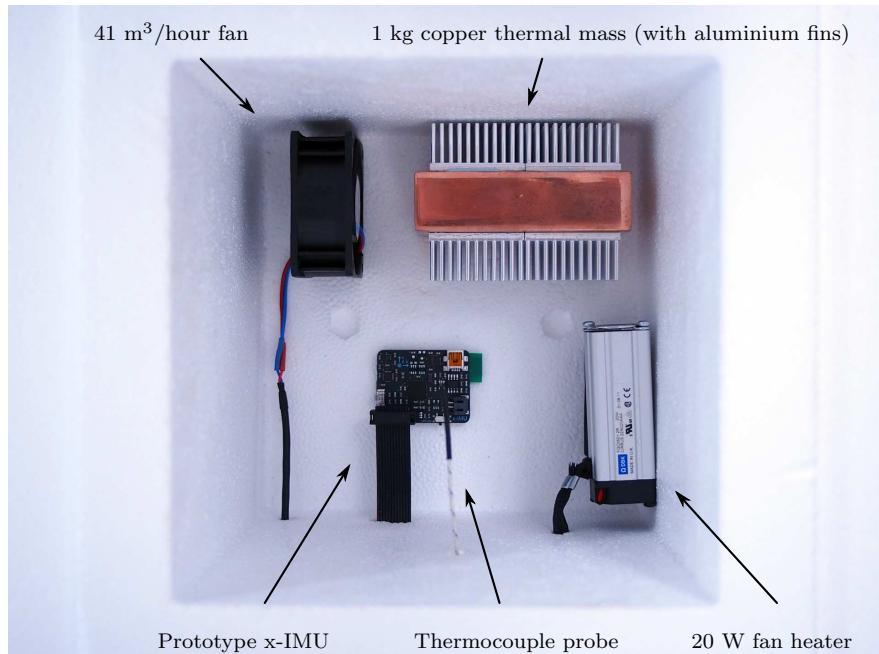


Figure 5.3: Thermal chamber internal components including the prototype x-IMU being tested

During operation, the fan continuously blows on to the thermal mass to circulate air within the chamber and transfer heat produced by the x-IMU to the thermal mass. The fan heater blows directly on to the thermal mass and away from the

x-IMU to prevent rapid thermal dynamics. Separate variable power supplies were used to power the fan and heater and a thermocouple meter used to monitor internal temperature. Software was written to log sensor from the x-IMU and display temperature measurements in real-time to guide manual control of the heater. Figure 5.4 shows the sealed thermal chamber with separate variable power supplies and thermocouple meter.



Figure 5.4: Sealed thermal chamber with thermocouple meter and separate variable power supplies for fan and heater

Eight x-IMUs were tested individually. For each test, the thermal chamber and x-IMU were initially left in an industrial freezer at -20°C for >12 hours. The chamber was then sealed within the freezer before being removed to conduct the experiment. The heat produced by the fans and x-IMU caused the temperature to rise of ~5°C over the first 5 minutes. The heater power supply was then gradually increased to maintain a fixed ascent of ~0.5°C per minute over 3 hours. The temperature ascent for each of the eight devices (named ‘A’ to ‘H’) is shown in Figure 5.5.

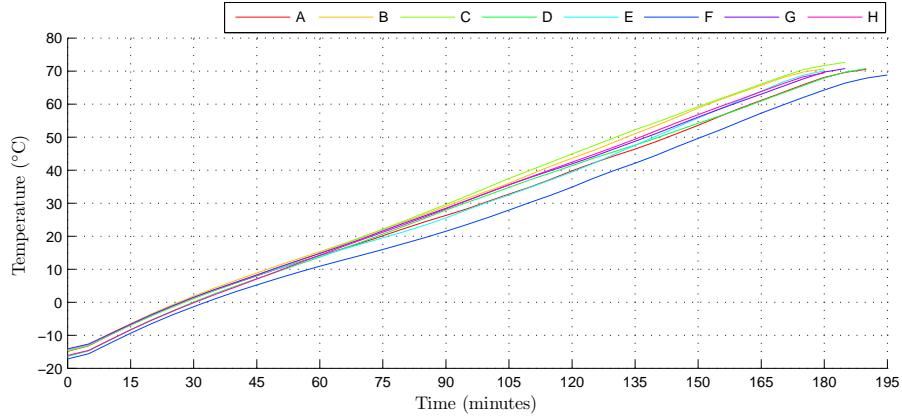


Figure 5.5: Measured gyroscope temperature during experiments for each of the eight devices tested

### 5.2.2 Monotonic thermal response

The measured gyroscope bias for each of the eight devices over a -20°C to 70°C temperature range is shown in Figure 5.6. The three plots indicate the bias for the  $x$ ,  $y$  and  $z$  axes termed  $b_{\omega x}$ ,  $b_{\omega y}$  and  $b_{\omega z}$ . The gyroscope output originally in units of lsb was converted to degrees per second using the typical sensitivity of 16.4 lsb/°/s specified in the datasheet.

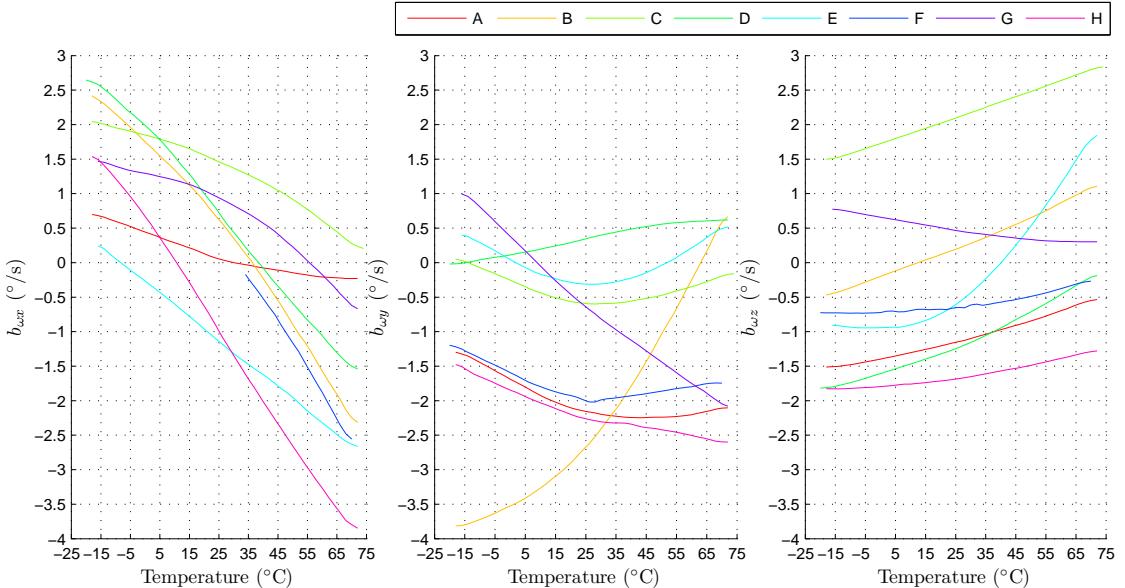


Figure 5.6: Gyroscope bias thermal response of eight devices over a -20°C to 70°C temperature range

The bias variation over temperature is within the  $\pm 20^{\circ}/s$  range specified in the

datasheet though the relationship was revealed to be both non-linear and unique to each device. A bias calibration solution that accounts for this variation would need to incorporate a non-linear model evaluated uniquely for each device. Although some devices may be approximated by a linear model over a limited temperature range, this would not be sufficient for all; notably,  $b_{\omega y}$  of device ‘E’ approximates a parabola centred at 25°C.

A calibration solution may model the thermal response of the bias as an  $n^{th}$  order polynomial function of temperature. To determine the necessary order, a polynomial fit was performed for each device for first to fifth order polynomials and the resultant maximum error of all devices summarised in Table 5.1. Each polynomial was computed using the MATLAB **polyfit** function with data points computed as the mean of each consecutive 2°C bin.

Polynomial order, $n$	Maximum error (°/s)		
	$b_{\omega x}$	$b_{\omega y}$	$b_{\omega z}$
1	0.328	0.400	0.395
2	0.190	0.146	0.144
3	0.041	0.120	0.109
4	0.038	0.053	0.040
5	0.037	0.047	0.035

Table 5.1: Gyroscope bias error assuming an  $n^{th}$  order temperature calibration model

Table 5.1 suggests at least a fourth order polynomial is necessary to account for the non-linear response over the investigated temperature range. At least five data points would be required to approximate such a polynomial. In practice, using the minimum number of data points will result in a less accurate approximation; the above accuracy was achieved using thousands of data points obtained over a three hour temperature ascent. Evaluating calibration parameters in this way for a large number devices would represent a significant practical challenge.

### 5.2.3 Thermal hysteresis

The gyroscope bias thermal response presented in Section 5.2.2 was obtained for a slow, monotonic temperature ascent. These experiments were extended to observe the bias behaviour when exposed to a range of thermal dynamics that could be achieved with the chamber, including: slow cooling, rapid cooling, rapid heating and thermal cycling. These investigations exposed a hysteresis characteristic of the gyroscope that was not described within any of the available manufacturer's documentation. Figure 5.7 shows the measured temperature during one such experiment that exemplifies the observations. The annotated markers 'a', 'b', 'c' and 'd' indicate key events during the experiment.

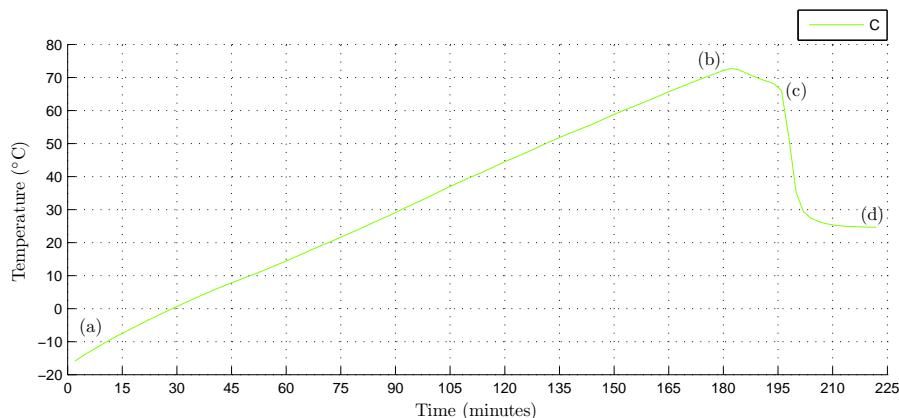


Figure 5.7: Measured gyroscope temperature during an experiment to demonstrate thermal hysteresis

The first  $\sim$ 180 minutes of the experiment, from point 'a' to point 'b', show the temperature ascent to  $\sim$ 70°C presented in Section 5.2.2. The heater was then switched off and the temperature slowly descended for  $\sim$ 15 minutes until point 'c'. The thermal chamber lid was then removed and the temperature descended rapidly until stabilising at  $\sim$ 25°C at point 'd'. The bias response is shown as 5.8. The annotated markers correspond to the events previously described within Figure 5.7. These plots indicate thermal hysteresis in all three axes.

Although this thermal hysteresis is a known characteristic [123, 27], there is little

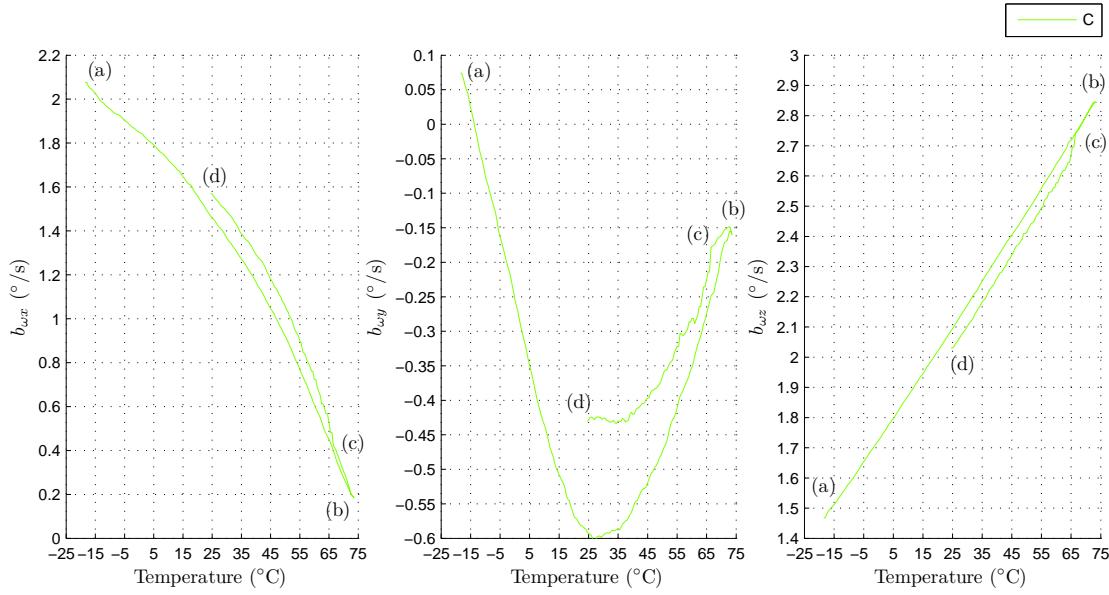


Figure 5.8: Thermal response of the gyroscope bias indicating thermal hysteresis

information available on the specific behaviour that may be expected. InvenSense were unable to offer an explanation as they do not thermal cycle their devices, only test that the devices operate over the specific temperature range [124]. The thermal response seen in Figure 5.8 suggests that the hysteresis is sensitive to both the change in direction of temperature and the rate of change of temperature. The hysteresis of  $b_{\omega y}$  is apparent the moment the device begins to cool from point ‘b’. However,  $b_{\omega z}$  seems to indicate little or no hysteresis for the  $\sim 15$  minutes of slow cooling but immediately shifts when rate of cooling is increased at point ‘c’.

The observed hysteresis indicates that the polynomial approximation of the thermal response previously suggest in Section 5.2.2 may be insufficient. For example, a polynomial fit to the initial monotonic response of  $b_{\omega y}$  would yield an error of  $1.55^{\circ}/s$  after cooling.

### 5.3 Gyroscope bias acceleration sensitivity

The MPU-6050 datasheet indicates a typical gyroscope bias sensitivity to static acceleration of  $0.1^\circ/\text{s/g}$  [121]. A sensitivity to static acceleration suggests that the error is dependent on the orientation of the device relative to gravity and could be compensated for through a calibration model that incorporates accelerometer measurements. The ‘g-sensitivity’ calibration model proposed by Bancroft *et al.* incorporates a  $3 \times 3$  matrix that suggests each axis bias is sensitive to accelerations in all directions. To investigate this characteristic, the device was mounted on a cube structure so that the six orientations of the cube on a flat surface would align each gyroscope axis with +1 g, -1 g and 0 g. The gyroscope bias was calculated as the mean of  $\sim 40,000$  samples per taken over 5 minutes for each orientation. 27 different devices were tested to demonstrate statistical significance.

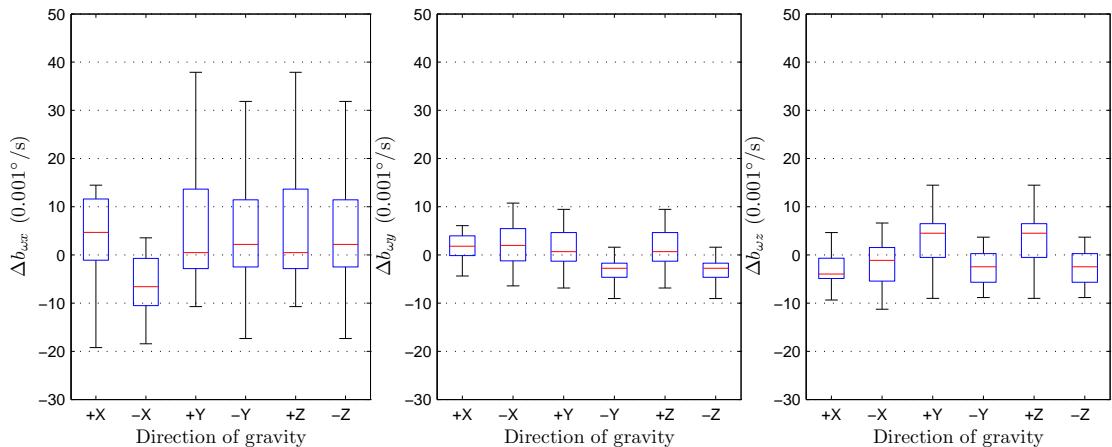


Figure 5.9: Gyroscope bias deviation for six different orientations relative to gravity. Box plots show the distribution for 27 devices (whiskers indicate  $1.5 \times$  interquartile range).

The deviation of the gyroscope bias for each axis,  $\Delta b_{\omega x}$ ,  $\Delta b_{\omega y}$  and  $\Delta b_{\omega z}$ , was calculated as the difference between the gyroscope bias at a given orientation from the mean for all orientations. The box plots in Figure 5.9 shows the distribution of  $\Delta b_{\omega x}$ ,  $\Delta b_{\omega y}$  and  $\Delta b_{\omega z}$  for all six orientations of the gyroscope relative to gravity. The data was scaled to units of degrees per second using

the typical sensitivity specified in the datasheet. Although the specific nature of the acceleration sensitivity is unknown, the box plots were expected to indicate a correlation between the direction of gravity and an offset error of with an order of magnitude of  $\sim 0.1\%$ s. Instead, the box plots indicate a mean deviation of approximately zero and a variance representative of noise. These results suggest that, contrary to the manufacturer's specification, the gyroscope bias does not have a significant sensitivity to static acceleration.

## 5.4 Gyroscope bias random walk

The MPU-6050 datasheet does not provide any information about the gyroscope bias time-variant random walk and so an investigation was conducted to quantify this characteristic. Four x-IMUs were sealed in the thermal chamber, each set up to log the gyroscope output and thermometer measurements to the on-board SD card at a rate of 512 samples per second. The thermal chamber was intended only to isolate the internal temperature from the ambient environment; the internal fan and heater were not used. Data was logged over a 48 hour period and then cropped to select a period with minimal temperature variations.

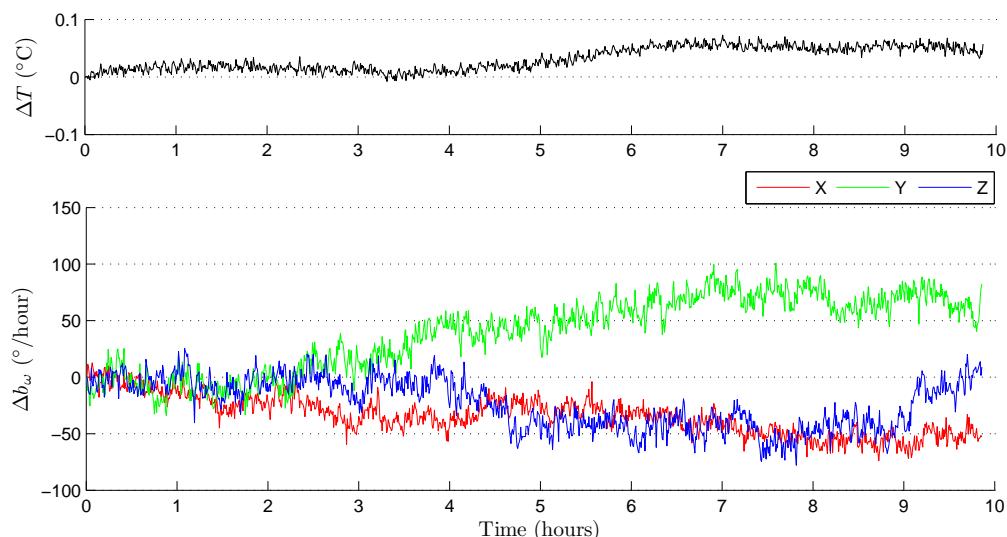


Figure 5.10: Gyroscope bias random walk over 10 hours

Figure 5.10 shows the measured temperature variation,  $\Delta T$ , and gyroscope bias variation,  $\Delta b_\omega$ , for one device over a 10 hour period with a temperature variation of  $\pm 0.025^\circ\text{C}$ . The measurement were average for each 30 second window to attenuate the high-frequency noise. Bias measurements were scaled to units of degrees per hour using the typical sensitivity specified in the datasheet. The results shown above are representative of the observations made for all devices and indicate a bias variation that does not correlate to temperature. This random walk resulted in a peak error of  $190^\circ/\text{hour}$  ( $\sim 0.053^\circ/\text{s}$ ) for the four devices over the 10 hour period.

### 5.4.1 Allan variance

Plotting the gyroscope bias variation over time clearly demonstrates random walk but does not provide an effective means for quantifying this characteristic. Allan variance is the accepted standard for evaluating gyroscope bias stability [125]. Stockwell provides a concise introduction to the implementation of the method in a Crossbow white paper [126].

The Allan variance, AVAR, is calculated by splitting the dataset into multiple consecutive windows of time and evaluating the variance in the gyroscope bias averaged for each window of time. The AVAR is computed for a range of different window sizes, termed  $\tau$ . The maximum value of  $\tau$  must not split the dataset into more than nine windows to maintain statistical significance [125]. Equation 5.1 describes the calculation where  $\text{AVR}^2(\tau)$  is the squared Allan variance for period  $\tau$ ,  $n$  is the number of windows created by the given value of  $\tau$ , and  $y(\tau)_i$  is the averaged gyroscope bias calculated for the  $i^{th}$  window.

$$\text{AVR}^2(\tau) = \frac{1}{2(n-1)} \sum_i (y(\tau)_{i+1} - y(\tau)_i)^2 \quad (5.1)$$

Figure 5.11 shows the AVAR for each axis calculated using values of  $\tau$  between 0.01 and 1000 seconds. The linear descent created on a logarithmic plot for small values of  $\tau$  indicates a reduction in variance due to the averaging of mean-zero random noise. The AVAR reaches a minimum at which point the length of  $\tau$  is sufficiently long to incorporate the changes due to the random walk and the variance begins to increase again. The stability of the gyroscope bias is quantified as the minimum Allan variance and corresponding value of  $\tau$  [126]. In this case the gyroscope bias stability is quantified as  $\sim 4^\circ/\text{hour}$  for  $\tau = 100$  seconds.

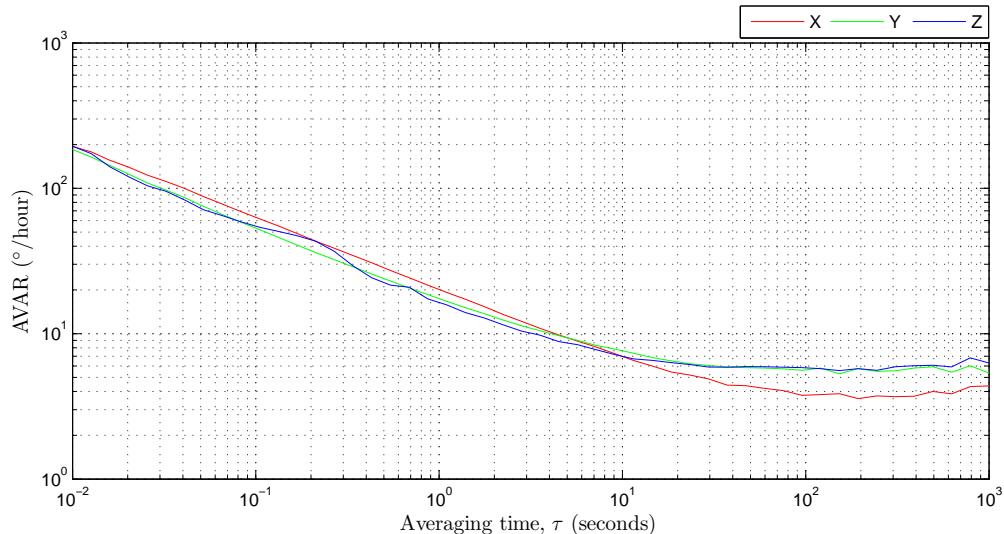


Figure 5.11: Allan variance plot of the gyroscope bias random walk

## 5.5 Gyroscope, accelerometer and magnetometer thermal response

The MPU-6050 and HMC5883L datasheets indicate that the gyroscope, accelerometer and magnetometer all have a temperature sensitivity that will cause each sensor's sensitivity and bias to vary over the operating temperature range. Table 5.2 summarises the temperature sensitivities as specified in the device datasheets [121, 122]. Although units such as  $^\circ/\text{C}$  suggest a linear relationship,

most of the specified quantities are ambiguous. The gyroscope bias investigations in Section 5.2 have already revealed the potential significance of such ambiguities. This provides a motivation to investigate the thermal response of the bias and sensitivity for all three types of sensor.

Parameter	Temperature sensitivity	Conditions
Gyroscope sensitivity	$\pm 2\%$	-
Gyroscope bias	$\pm 20^\circ/\text{s}$	-40°C to 85°C
Accelerometer sensitivity	$\pm 0.02\%/\text{°C}$	-40°C to 85°C
Accelerometer bias ( $x$ and $y$ axis)	$\pm 35 \text{ mg}$	0°C to 70°C
Accelerometer bias ( $z$ axis)	$\pm 60 \text{ mg}$	0°C to 70°C
Magnetometer bias	-	-
Magnetometer sensitivity	-0.3%/°C	-

Table 5.2: Temperature sensitivity of sensor parameters as specified in device datasheets. “-” indicates that the information is not provided.

The sensitivity and bias and may be evaluated for each sensor by exposing each axis to both a positive and negative reference. For example, this could be a  $\pm 1 \text{ g}$  reference for each accelerometer axis. The axis bias may then be calculated as Equation 5.2 where  $b$  is the axis bias in lsb,  $u_{+r}$  is the axis output when exposed to the positive reference and  $u_{-r}$  is the axis output when exposed to the negative reference.

$$b = \frac{u_{+r} + u_{-r}}{2} \quad (5.2)$$

The axis sensitivity may be calculated as Equation 5.3 where  $s$  is the axis bias in lsb per units of the reference signal,  $u_{+r}$  is the axis output when exposed to the positive reference and  $u_{-r}$  is the axis output when exposed to the negative reference, and  $r$  is the magnitude of the reference signal.

$$s = \frac{|u_{+r}| + |u_{-r}|}{2r} \quad (5.3)$$

The bias and sensitivity may be determined at a given temperature using Equations 5.2 and 5.3. To evaluate variation over temperature, each sensor must

be exposed to a corresponding positive and negative reference while simultaneously regulating the device temperature. The gyroscope and accelerometer both require physical reference quantities that can only be achieved through controlled motion. A three-axis thermal chamber was created to facilitate these investigations.

### 5.5.1 Three-axis thermal chamber

The three-axis thermal chamber was created from a polystyrene box and a cube structure designed to slot within the internal volume to mount four battery powered prototype x-IMUs in alignment with the outer dimensions of the chamber. A 1 kg copper plate with aluminium fins and battery powered fan were mounted internally to blow air directly on to the x-IMUs. The chamber can be orientated at six orientations on a flat surface to expose each accelerometer axis to +1 g and -1 g; and at six orientations relative to the rotational axis of a rate table to expose each gyroscope axis to a positive and negative reference of angular velocity. A Technics SL-1200MK2 turntable was used to provide a repeatable reference of 200°/s (33<sup>1</sup>/<sub>3</sub> Rotations Per Minute (RPM)). Details on the evaluation of this turntable are discussed in Chapter 6. Figure 5.12 shows the components that make up the three-axis thermal chamber.

Each x-IMUs was set up to log sensor data to on-board SD cards at a rate of 256 samples per second while simultaneously streaming the data via Bluetooth to be viewed in real-time. The HMC5883L was mounted only a few millimetres from the MPU-6050 and so assumed to be at the same temperature indicated by the MPU-6050 internal thermometer. The thermal chamber was stored in a -30°C industrial freezer for >12 hours prior to the investigation. The x-IMUs and fan were switched on and thermal chamber sealed before being removed from the freezer. The thermal chamber was then continuously cycled through six static orientations relative to gravity and six orientations on the turntable

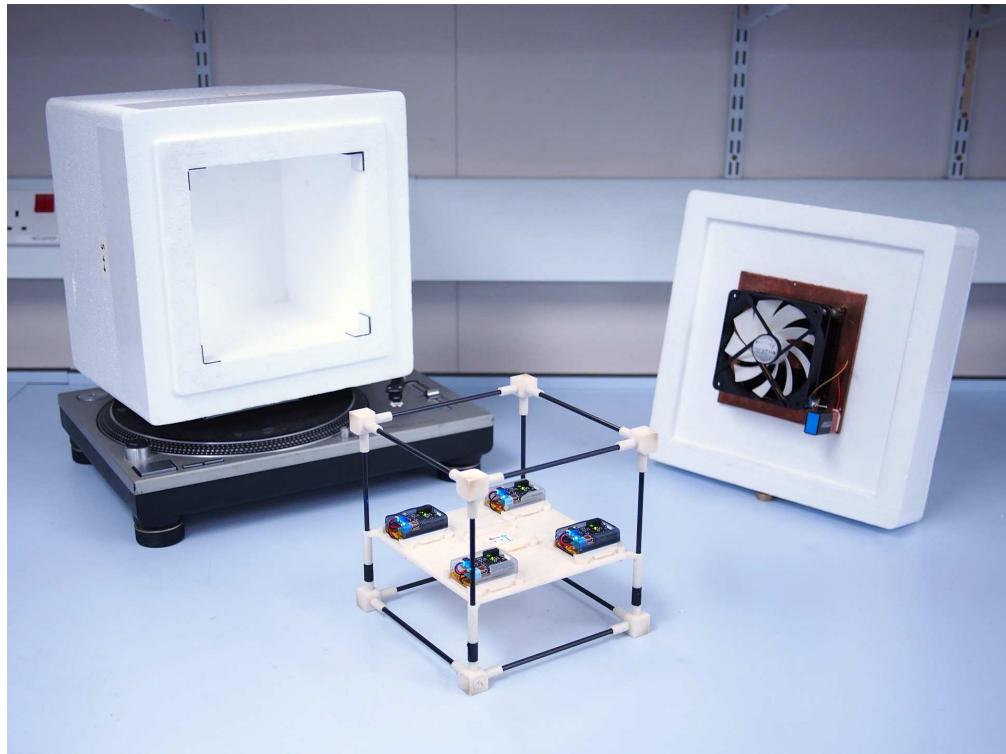


Figure 5.12: Three-axis thermal chamber components including: polystyrene box with internal thermal mass and fan, cube structure to mount four prototype x-IMUs, and SL-1200MK2 turntable

while the internal temperature increased with the heat generated by the internal electronics from approximately -25°C to 40°C over 2.5 hours. Figure 5.13 shows this temperature ascent for each of the four devices: ‘A’, ‘B’, ‘C’ and ‘D’. The necessity for the chamber to be wireless prevented the use of active heating which both limited the maximum temperature temperature and prevented control of the rate of ascent. Nevertheless, the temperature range achieved was sufficient to investigate the thermal response of each of the sensors.

### 5.5.2 Gyroscope sensitivity and bias thermal response

A MATLAB script was written to process the logged data and extract the periods at which the each gyroscope axis was detected as being at the reference speeds of -200°/s and +200°/s. The average gyroscope output was computed for discrete

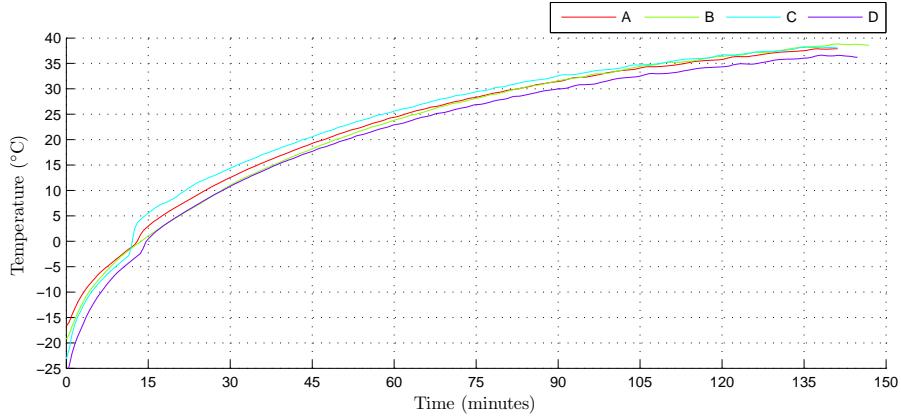


Figure 5.13: Measured temperature ascent of sensors within the three-axis thermal chamber over 2.5 hours

temperature intervals of  $2^{\circ}\text{C}$  to yield a value of  $u_{+\text{r}}$  and  $u_{-\text{r}}$  for each axis at each temperature interval. The corresponding sensitivities,  $s_{\omega x}$ ,  $s_{\omega y}$  and  $s_{\omega z}$ ; and biases,  $b_{\omega x}$ ,  $b_{\omega y}$  and  $b_{\omega z}$ , were calculated using Equations 5.3 and 5.2 to yield the results shown in Figure 5.14. The bias was scaled to units of degrees per second using the typical sensitivity specified in the datasheet.

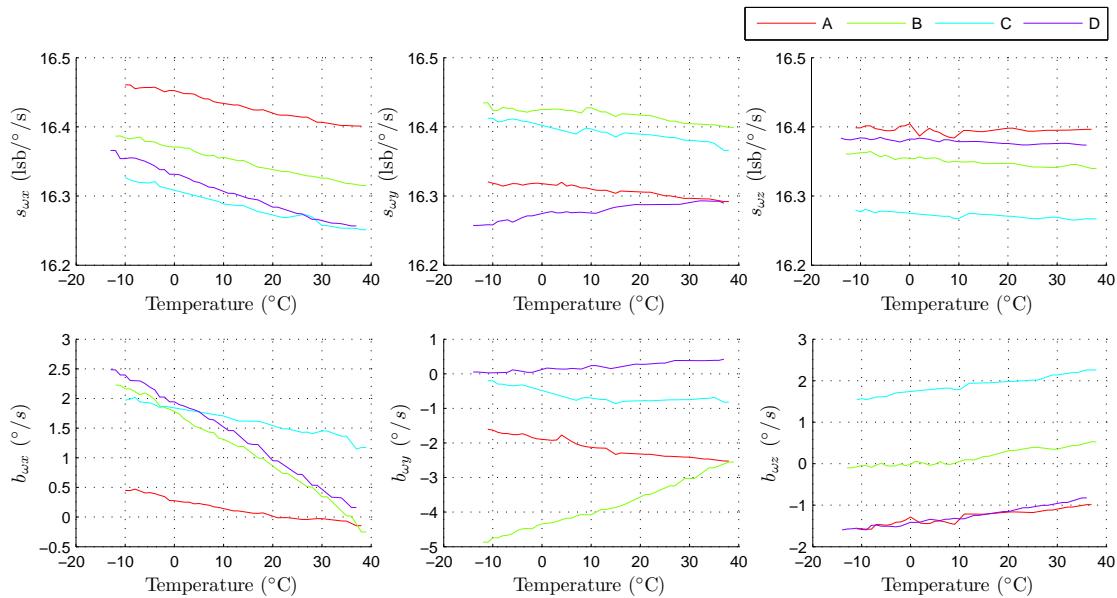


Figure 5.14: Thermal response of gyroscope sensitivity and bias from  $-15^{\circ}\text{C}$  to  $40^{\circ}\text{C}$

The gyroscope biases  $b_{\omega x}$ ,  $b_{\omega y}$  and  $b_{\omega z}$  for the four devices has already been investigated in greater detail in Section 5.2. The thermal responses shown in Figure

5.14 match those previously demonstrated for devices ‘A’, ‘B’, ‘C’ and ‘D’. The gyroscope sensitivity temperature coefficient was calculated for the four devices as the ratio of the best-fit gradient to the mean sensitivity over the temperature range. The results are summarised in Table 5.3.

Device	Temperature coefficients (%/°C)		
	$s_{\omega x}$	$s_{\omega y}$	$s_{\omega z}$
A	-0.0083	-0.0035	-0.0004
B	-0.0091	-0.0035	-0.0027
C	-0.0097	-0.0051	-0.0016
D	-0.0138	0.0043	-0.0012

Table 5.3: Gyroscope sensitivity temperature coefficients

The datasheet specifies a maximum gyroscope sensitivity variation of  $\pm 2\%$  over an unspecified temperature range. This is confirmed by the worst-case coefficient of -0.0138%/°C which corresponds to a 1.73% variation over the operating range of -40°C to 80°C. A calibration model may account for this temperature coefficient as linear approximation. However, the magnitude may already be sufficiently low for most applications; particular those operating within a limited temperature range.

### 5.5.3 Accelerometer sensitivity and bias thermal response

As with the gyroscope analysis, accelerometer data was extracted to yield averaged measurements of  $u_{+r}$  and  $u_{-r}$  for each axis at each temperature interval. The corresponding sensitivities  $s_{ax}$ ,  $s_{ay}$  and  $s_{az}$ , and biases  $b_{ax}$ ,  $b_{ay}$  and  $b_{az}$ , were calculated using Equations 5.3 and 5.2 to yield the results shown in Figure 5.15. The bias was scaled to units of mg using the typical sensitivity specified in the datasheet.

The accelerometer sensitivity temperature coefficient was calculated for the four devices as the ratio of the best-fit gradient to the mean magnitude of the measured sensitivity over the temperature range. The results are summarised in Table 5.4.

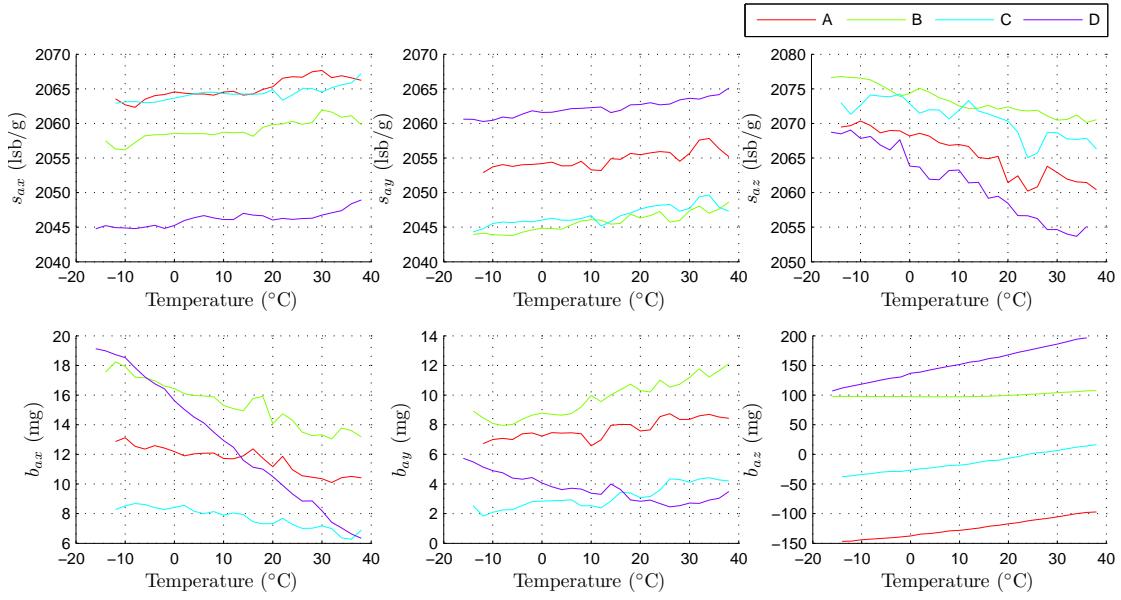


Figure 5.15: Thermal response of accelerometer sensitivity and bias from  $-15^{\circ}\text{C}$  to  $40^{\circ}\text{C}$

The accelerometer sensitivity is indicated as having a temperature coefficient of  $\pm 0.02\%/\text{ }^{\circ}\text{C}$  in the datasheet. The worst-case measured temperature coefficient of  $-0.0155\%/\text{ }^{\circ}\text{C}$  is within this range.

Device	Temperature coefficients ( $\%/\text{ }^{\circ}\text{C}$ )		
	$s_{ax}$	$s_{ay}$	$s_{az}$
A	0.0042	0.0032	-0.0097
B	0.0040	0.0040	-0.0060
C	0.0027	0.0035	-0.0068
D	0.0027	0.0034	-0.0155

Table 5.4: Accelerometer sensitivity temperature coefficients

The accelerometer bias temperature sensitivity was calculated as the best-fit gradient and the results summarised in Table 5.5. The datasheet specifies the temperature variation of  $\pm 35$  mg for the  $x$  and  $y$  axis and  $\pm 60$  mg in the  $z$  axis for a temperature range of  $0^{\circ}\text{C}$  to  $70^{\circ}\text{C}$ . The lower plots of 5.15 confirm a linear relationship. The worst-case measured temperature sensitivities in Table 5.5 indicate a temperature variation of 17.6 mg in the  $x$  axis but 118.6 mg in the  $z$  axis. While the variation in the  $x$  and  $y$  axis is within the manufacturer's specified tolerances, the variation  $z$  in the axis for three of the four devices are not.

Device	Temperature coefficients (mg/°C)		
	$b_{ax}$	$b_{ay}$	$b_{az}$
A	-0.054	0.037	1.001
B	-0.096	0.076	0.168
C	-0.043	0.046	1.045
D	-0.252	-0.050	1.694

Table 5.5: Accelerometer bias temperature coefficients

The accelerometer temperature sensitivities may be compensated for in calibration though a linear approximation though the sensitivities may be sufficiently low that thermal calibration is not necessary for most applications. The  $z$  axis bias represents a possible exception as its relatively large  $\sim 100$  mg variation observed over the 50°C temperature range is beyond the manufacturer's specified tolerances and significantly large relative to the 1 g typical measurement expected by most applications.

### 5.5.4 Magnetometer sensitivity and bias thermal response

The magnetometer incorporates an internal field generator that functions within an automated sequence to sample each axis when the field generator is disabled and then again when it is enabled. The difference between these two samples is provided as a measurement of the reference field strength independent of motion and the ambient magnetic field. The prototype x-IMUs were configured to continuously log this data for positive and negative fields. Averaged measurements of  $u_{+r}$  and  $u_{-r}$  were calculated for each axis at each 2°C temperature interval. The corresponding sensitivities  $s_{mx}$ ,  $s_{my}$  and  $s_{mz}$ , and biases  $b_{mx}$ ,  $b_{my}$  and  $b_{mz}$ , were calculated using Equations 5.3 and 5.2 for each temperature interval to yield the results shown in Figure 5.16.

The magnetometer sensitivity temperature coefficient was calculated for the four devices as the ratio of the best-fit gradient to the mean magnitude of the measured

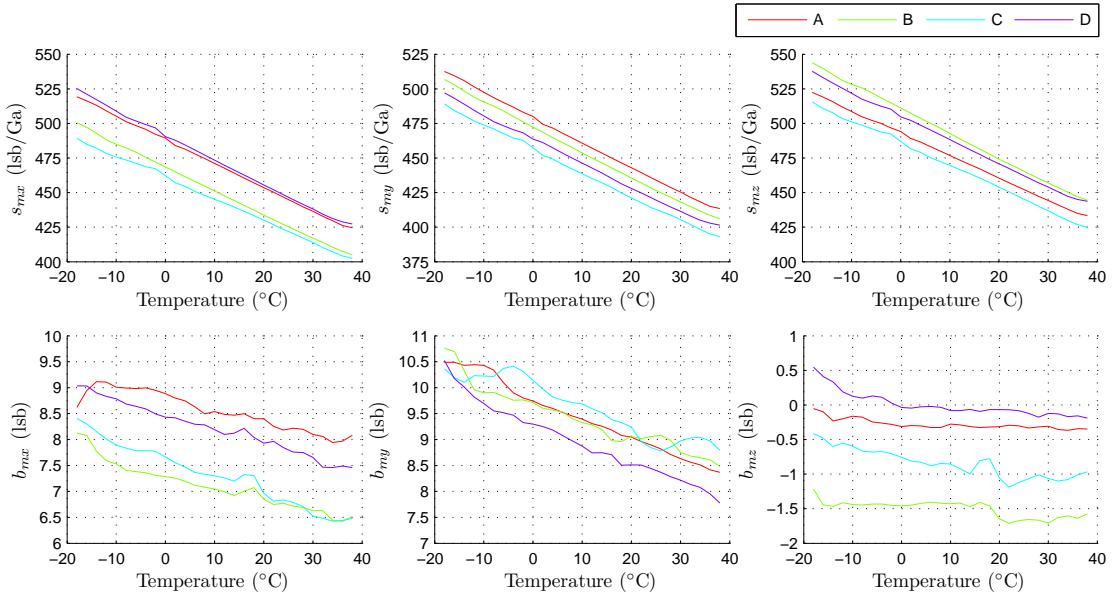


Figure 5.16: Thermal response of magnetometer sensitivity and bias from  $-15^{\circ}\text{C}$  to  $40^{\circ}\text{C}$

sensitivity over the temperature range. The magnetometer bias temperature coefficient was calculated in the same way. The results are summarised in Table 5.6.

Device	Temperature coefficients ( $\%/\text{ }^{\circ}\text{C}$ )					
	$s_{mx}$	$s_{my}$	$s_{mz}$	$b_{mx}$	$b_{my}$	$b_{mz}$
A	-0.37	-0.39	-0.34	-0.24	-0.43	1.28
B	-0.38	-0.40	-0.36	-0.36	-0.36	0.37
C	-0.35	-0.39	-0.35	-0.47	-0.33	1.31
D	-0.37	-0.39	-0.35	-0.34	-0.46	-190.26

Table 5.6: Magnetometer sensitivity and bias temperature coefficients

The datasheet specifies a sensitivity temperature coefficient of  $-0.3\%/\text{ }^{\circ}\text{C}$  which is confirmed by the results in Table 5.6. The datasheet does not document any bias variation over temperature. Although the lower plots of Figure 5.16 suggest that the bias does vary over temperature, representing this variation as  $\%/\text{ }^{\circ}\text{C}$  provides an alternative explanation. The bias temperature coefficients calculated in Table 5.6 approximate the  $-0.3\%/\text{ }^{\circ}\text{C}$  expected in the sensitivity which suggests that the observed bias variation is the result of scaling by the sensitivity and that the bias itself is subject to little or no variation over temperature. The low magnitudes of

$b_{mx}$ ,  $b_{my}$  and  $b_{mz}$  result in the increased variance of the bias temperature coefficient relative to the sensitivity temperature coefficients. In the case of  $b_{mz}$ , calculations approach divisions by zero and so cannot be expected to provide meaningful results.

A magnetometer sensitivity error is of particular significance to AHRS applications as hard-iron distortions may shift a small operating range of a magnetometer axis away from zero. For example, if such distortions shifted an operating range from  $\pm 100$  to a range of 800 to 1000, an offset of 900 can be subtracted to yield a ‘distortion-free’ measurement. However, if a sensitivity error of -10% is also in effect then the compounded operating range would become 720 to 900 and subtracting the assumed 900 offset would yield a range of -180 to 0 corresponding to a maximum error of 180%. This numerical example represents a hard-iron distortion with a magnitude nine times that of the Earth’s magnetic field, such a distortion would be extreme but is not unrealistic.

The absence of a bias variation means that the magnetometer thermal response can be accounted for by scaling all measurements by approximately  $-0.3\%/\text{ }^{\circ}\text{C}$ , though the variation in above results indicate that the specific coefficient would need to be evaluated for each device. An alternative solution would be to use the magnetometer internal field generators to calculate the sensitivity at discrete time or temperature intervals; this technique is suggested in the magnetometer datasheet.

## 5.6 Part variation

The manufacture of the IMU platforms described in Chapter 8 provided an opportunity to investigate the variation of gyroscope, accelerometer and magnetometer characteristics within a population of over 150 x-IMUs. The MPU-6050 and HMC5883L were each sourced from a single production reel and

so are assumed to represent common batch and silicon revision. It is common for electronic components within a given batch to share common characteristics which may not be representative of the part in general. The methods described in Section 5.5 were repeated without the thermal chamber to measure the bias and sensitivity of each sensor once the device had reached stable temperature of  $\sim 35^{\circ}\text{C}$ . The following sections summarise the distribution of these parameters for each sensor.

### 5.6.1 Gyroscope part variation

Figure 5.17 shows the normalised distribution of the gyroscope axis sensitivities  $s_{\omega x}$ ,  $s_{\omega y}$  and  $s_{\omega z}$  and biases  $b_{\omega x}$ ,  $b_{\omega y}$  and  $b_{\omega z}$  measured for the 150 MPU-6050 parts. The biases have been scaled using the typical sensitivity specified in the datasheet.

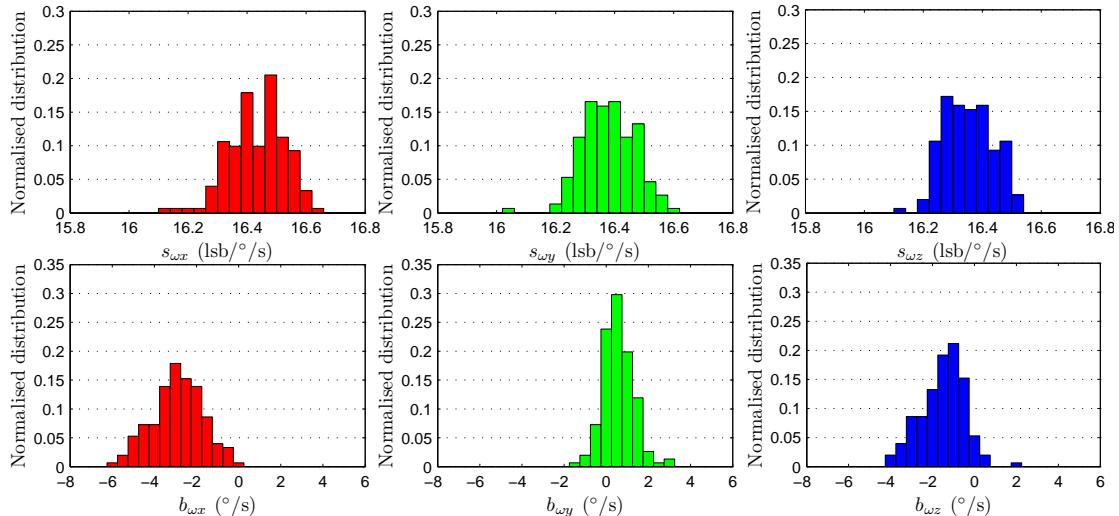


Figure 5.17: Gyroscope sensitivity and bias normalised distribution

The datasheet specifies that the gyroscope sensitivity of  $16.4 \text{ lsb}/^{\circ}\text{s} \pm 3\%$  at  $25^{\circ}\text{C}$ ; corresponding to a range of  $15.9 \text{ lsb}/^{\circ}\text{s}$  to  $16.9 \text{ lsb}/^{\circ}\text{s}$ . The histograms confirm a mean of  $\sim 16.4 \text{ lsb}/^{\circ}\text{s}$  with no outliers exceeding these tolerances. The bias is specified as being within the range  $\pm 20^{\circ}/\text{s}$  at  $25^{\circ}\text{C}$ . The lower histograms of Figure 5.17 reveal that the true distribution is more limited. Whereas the sensitivity

distributions are centred on the specified nominal value, the mean bias for each axis is not centred on zero. For example, the mean value of  $b_{\omega x}$  is approximately  $-3^\circ/\text{s}$ . Although this mean suggests a more accurate nominal value, this may be specific to the batch.

### 5.6.2 Accelerometer part variation

Figure 5.18 shows the normalised distribution of the accelerometer axis sensitivities  $s_{ax}$ ,  $s_{ay}$  and  $s_{az}$  and biases  $b_{ax}$ ,  $b_{ay}$  and  $b_{az}$  measured for the 150 MPU-6050 parts. The biases have been scaled using the typical sensitivity specified in the datasheet.

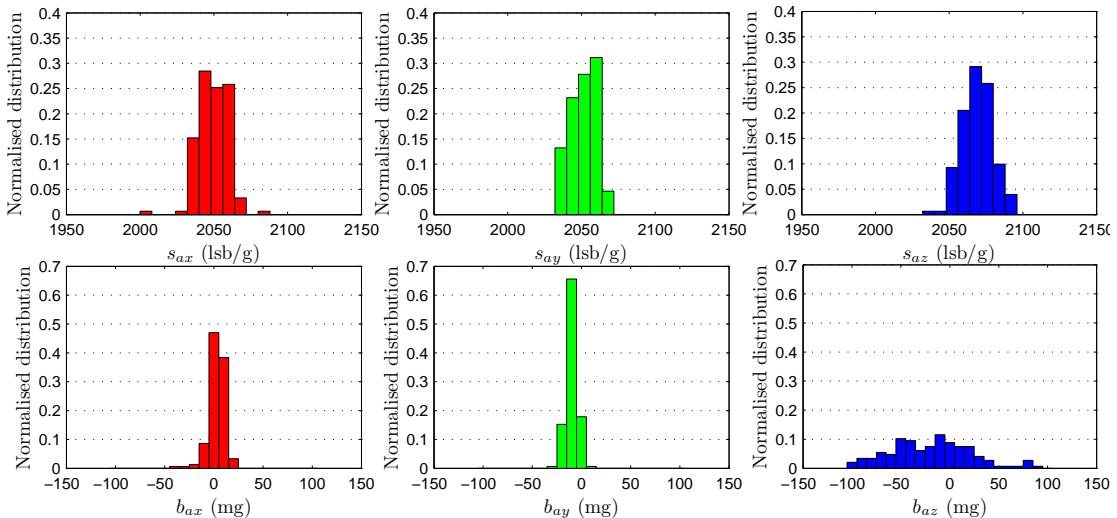


Figure 5.18: Accelerometer sensitivity and bias normalised distribution

The datasheet specifies a sensitivity of 2048 lsb/g  $\pm 3\%$  at  $25^\circ\text{C}$ ; corresponding to a range of 1987 lsb/g to 2109 lsb/g. The histograms confirm a mean of  $\sim 2048$  lsb/g with no outliers exceeding these tolerances. The bias is specified as being within the range  $\pm 50$  mg for  $x$  and  $y$  axis and  $\pm 80$  mg for  $z$  axis at  $25^\circ\text{C}$ . As with the gyroscope, the lower histograms of Figure 5.18 reveal that the true distribution is more limited for the  $x$  and  $y$  axis. However, the distribution for the  $z$  axis exceeds the specified range. This is consistent with the thermal response of the  $z$  axis which was also found to be outside of the manufacturer's tolerances.

### 5.6.3 Magnetometer

Figure 5.18 shows the normalised distribution of the magnetometer axis sensitivities  $s_{mx}$ ,  $s_{my}$  and  $s_{mz}$  and biases  $b_{mx}$ ,  $b_{my}$  and  $b_{mz}$  measured for the 150 HMC5883L parts.

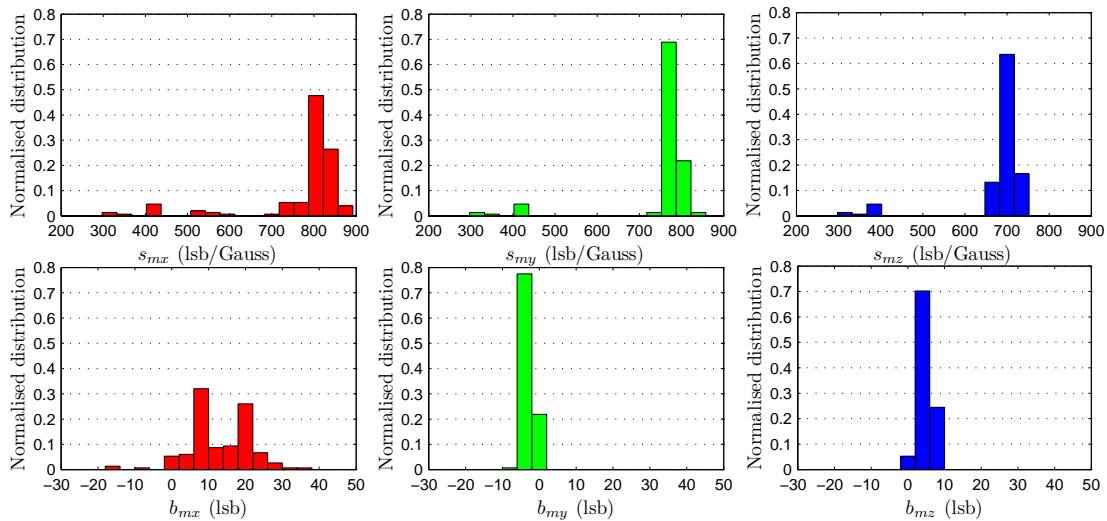


Figure 5.19: Magnetometer sensitivity and bias normalised distribution

The x-IMUs were configured for a magnetometer measurement range of  $\pm 4.0$  Gauss with a corresponding sensitivity of 440 lsb/Gauss specified in the device datasheet. However, the mean sensitivity indicated in Figure 5.19 is closer to the 820 lsb/Gauss sensitivity associated with the magnetometer's default full-scale range of  $\pm 1.9$  Gauss. There are also second peaks around 440 lsb/Gauss which suggest that a malfunction in the HMC5883L or prototype x-IMU firmware has caused an incorrect full-scale range to be selected for a majority of the devices tested. The tests conducted during the collection of this data focussed on the static heading error achieved by the magnetometer and so were insensitive to this scaling error. It was not possible to repeat the data collection for the 150 devices. It is therefore not possible to comment on the distributions of 5.19.

## 5.7 Conclusions

The purpose of this chapter is to determine the significant sensor characteristics that must be accounted for within a calibration solution. Investigations revealed undocumented characteristics as well as discrepancies in the datasheets. The gyroscope bias was found to exhibit thermal hysteresis. The accelerometer  $z$  axis performance was found to be outside of the specified tolerances in both its thermal response and the part variation survey of over 150 parts. Although the gyroscope datasheet indicates a bias sensitivity to static acceleration, a correlation could not be confirmed.

Thermal response investigations confirmed low temperature coefficients of most characteristics. The gyroscope bias and magnetometer sensitivity are exceptions that would need to be accounted for in calibration. Magnetometer temperature calibration may be achieved using the manufacturer's recommended solution. Although the gyroscope incorporates a thermometer to facilitate temperature calibration, the non-linear response and hysteresis may limit success in a practical implementation. It may therefore be more effective to avoid gyroscope bias temperature calibration and instead rely on a bias compensation built-in to the AHRS algorithm. The findings presented in this chapter were used to determine the scope of the calibration solutions developed in the next chapter.



# **Chapter 6**

## **Calibration of low-cost MEMS sensors for AHRS applications**

### **6.1 Introduction**

This chapter describes calibration solutions for gyroscopes, accelerometers and magnetometers for use in AHRS. The proposed solutions are intended to be practical in the sense that they can be implemented without the need for specialist equipment and so be a viable solution for production scenarios. The solutions and methods for IMU calibration are well documented [119, 120, 127, 118]. The solutions presented in this chapter were developed to meet the specific requirements of the MPU-6050 and HMC-5883L as presented in the previous chapter. Furthermore, the calibrated performance for each sensor is evaluated and benchmarked against selected commercial IMUs.

### 6.1.1 Scope of calibration

The calibration solutions presented in this chapter account for the following characteristics.

- Individual axis bias and sensitivity
- Sensor misalignment
- Hard-iron and soft-iron distortions

Notable characteristics not accounted for by the calibration models include the gyroscope bias acceleration sensitivity and the thermal response of the sensors. Previous investigations indicated could not demonstrate a correlation between the gyroscope bias and static acceleration. Thermal response investigations reveal a significant sensitivity in the magnetometer that may be accounted for using the manufactures recommended technique; and of the gyroscope bias which can be accounted for by the AHRS algorithm.

### 6.1.2 Chapter summary

The next section summarises the commercial IMUs chosen as benchmarks. Section 6.4 describes the calibration solution for the gyroscope. Section 6.5 describes the calibration solution for the accelerometer along side an empirical equation to contrast uncalibrated and calibrated AHRS performance. Section 6.6 describes three different magnetometer calibration solutions and demonstrates the AHRS performance of each solution for both ‘typical’ and ‘extreme’ magnetic distortion scenarios. Section 6.7 presents an optimisation that can be made to simplify the sensor alignment arithmetic, and significantly reduce the computation load in application. The chapter concludes with a summary of proposed calibration solutions.

## 6.2 Commercial IMUs chosen as benchmarks

Four commercial IMUs were chosen to benchmark the performance of the proposed calibration solutions. Each product appeared on the market within the last 5 years and is priced less than 3000 USD. The chosen devices, shown Figure 6.1, were the MicroStrain 3DM-GX3-25, Xsens MTw, VectorNav VN-100 and CH Robotics UM6. MicroStrain and Xsens are recognised as long-established and leading manufacturers whereas VectorNav and CH Robotics represent new manufacturers whom have taken advantage of recent advances in MEMS technology to offer lower-end devices at a significantly lower price.



Figure 6.1: Commercial IMUs chosen to benchmark performance. From left to right: MicroStrain 3DM-GX3-25, Xsens MTw, VectorNav VN-100 and CH Robotics UM6

Table 6.1 summarises the calibrated performance of each IMU as specified in the manufacturer's documentation [108, 128, 109, 52, 129]. Prices and release dates were either obtained from the manufacturer's website or confirmed by contacting the manufacturer directly. The MPU-6050 [121] and HMC-5883L [122] have also been included in the table for comparison, these devices function only as an uncalibrated gyroscope, accelerometer and magnetometer and do not have an associated AHRS performance.

There is a clear correlation between the AHRS performance and price with the more expensive devices providing greater accuracy. Both MicroStrain and Xsens specify a dynamic error. However, the meaning of this quantity is ambiguous and it cannot be used for the purposes of benchmarking; Xsens specify the dynamic error along side the disclaimer: "May depend on type of motion" [128]. The proposed

	3DM-GX3-25	MTw	VN-100	UM6	MPU-6050 / HMC5883L
Price	\$2550	\$2990	\$500	\$199	<\$10
Year of release (circa)	2009	2011	2008	2009	2010
<b>AHRS performance</b>					
Static error (pitch/roll)	$\pm 0.5^\circ$	$<0.5^\circ$	$0.5^\circ$	$<2^\circ$	N/A
Static error (heading)	$\pm 0.5^\circ$	$1^\circ$	$2^\circ$	$<5^\circ$	N/A
Dynamic error (pitch/roll)	$\pm 2^\circ$	$2^\circ$	-	-	N/A
Dynamic error (heading)	$\pm 2^\circ$	$2^\circ$	-	-	N/A
Operating temperature	-40 to 70°C	0 to 55°C	25°C	-	N/A
<b>Gyroscope</b>					
Range ( $^\circ/\text{s}$ )	300	1200	2000	2000	2000
Linearity (%)	0.03	0.1	0.1	0.2	0.2
Noise ( $^\circ/\text{s}/\sqrt{\text{Hz}}$ )	0.03	0.05	0.005	0.03	0.005
<b>Accelerometer</b>					
Range (g)	5	16	16	2	16
Linearity (%)	0.1	0.2	0.5	-	0.5
Noise ( $\mu\text{g}/\sqrt{\text{Hz}}$ )	80	0.0003	400	-	400
<b>Magnetometer</b>					
Range (Gauss)	2.5	1.5	2.5	2.5	8.1
Linearity (%)	0.4	0.2	0.1	-	0.1
Noise ( $\mu\text{Gauss}/\sqrt{\text{Hz}}$ )	100	150	140	-	-

Table 6.1: Specified performance of each IMU as specified in the manufacturer's documentation. “-” indicates that a parameter is not specified.

calibration solutions were therefore benchmarked against the static errors only.

The specified heading accuracy of each device also represents a potential ambiguity. It can be shown (see Appendix C) that a calibration offset error corresponding to a  $0.5^\circ$  heading error at the equator would result in a  $1.5^\circ$  heading error in the UK due to a varying inclination of the Earth's magnetic field. This should be considered when benchmarking the heading performance.

## 6.3 Calibration cube

The calibration cube is a precision tool used to calibrate triple-axis gyroscopes, accelerometers and magnetometers. The structure allows sensors to be rotated through orthogonal orientations relative to a physical reference signal. For

example, each accelerometer axis may be exposed to a +1 g or -1 g reference by placing the cube on a level surface so that the axis is either pointing directly up or down. The specific use of the cube varies with each type of sensor. Figure 6.2 shows the calibration cube with nine prototype x-BIMUs held in alignment with the cube axes.

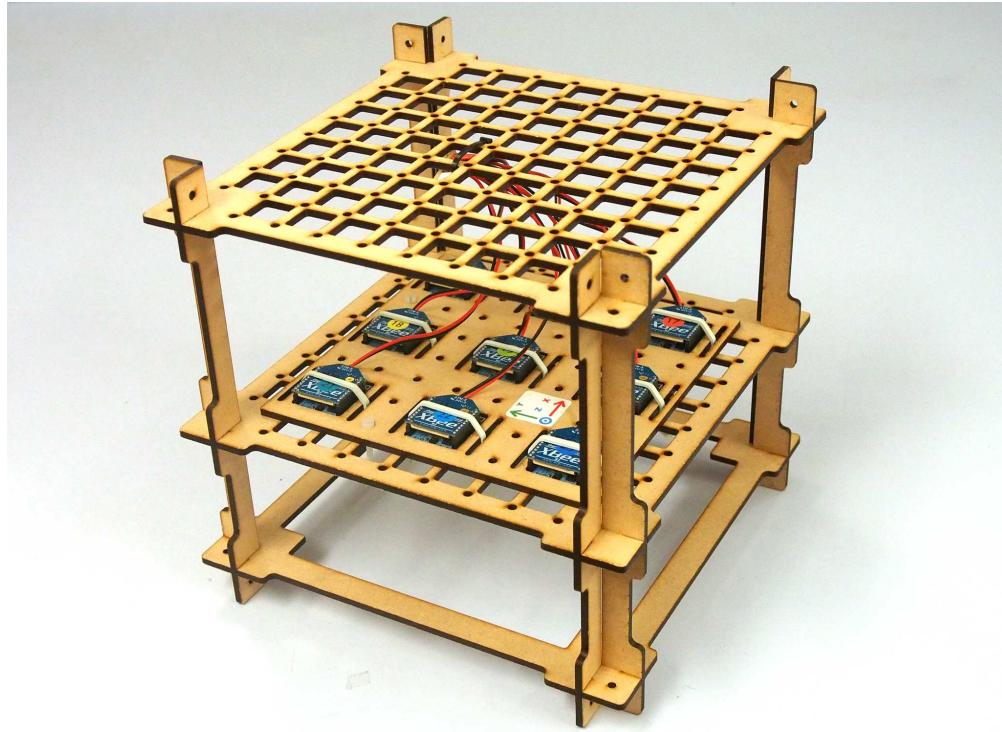


Figure 6.2: Calibration cube with nine prototype x-BIMUs held in alignment with the cube axes

The calibration cube was laser cut from 3 mm Medium-Density Fibreboard (MDF) and assembled through press-fit joints to avoid metal components that may interfere with magnetic calibration. The structure is comprised of three horizontal levels separated by vertical spacers. The middle and top levels feature grids of mounting holes and cut-outs for securing the sensors and additional equipment such as batteries. The bottom level is an empty frame to provide rigidity to the structure. Sensors mounted on the middle level are held in alignment by a laser cut MDF plate with cut-outs matching the dimensions of the x-BIMU. Each corner of the cube functions as a socket to ensure alignment when the cube is mounted on the

protractor described in Section 6.3.2. The calibration cube design is open-source<sup>1</sup>.

### 6.3.1 Calibration cube alignment errors

In practise, sensors mounted on the calibration cube will incorporate a small misalignment and this will cause an attenuation in the measured reference signal. For example, Figure 6.3 shows a misalignment between the calibration cube axes,  $x_c$ ,  $y_c$  and  $z_c$ ; and the sensor axes,  $x_s$ ,  $y_s$  and  $z_s$ .  $x_c$  has been aligned with the reference signal  $r$  but a misalignment between  $x_s$  and  $x_c$  of angle  $\gamma$  prevents the sensor axis from being exposed to the full magnitude of the reference signal; it is instead exposed to the attenuated reference signal  $r_a$ .

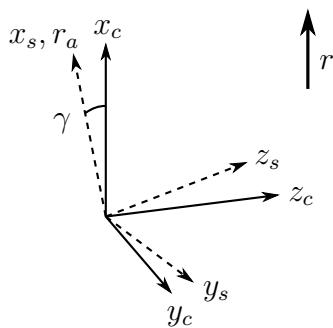


Figure 6.3: A misalignment between the calibration cube and sensor  $x$  axes of angle  $\gamma$  prevents the sensor axis from being exposed to the full magnitude of the reference signal  $r$ .

The relationship between the reference signal,  $r$ , and the attenuated reference signal,  $r_a$ , is defined by Equation 6.1 where  $\gamma$  is the angular misalignment between the sensor axis and the direction of the reference signal.  $r_a$  can be approximated as a truncated Taylor series.

$$r_a = r \cos \gamma \approx r \left( 1 - \frac{\gamma^2}{2} \right) \quad (6.1)$$

The cosine relationship means that the error in the reference signal has greater tolerance to misalignment errors of a small magnitude. The attenuation of the

<sup>1</sup><https://github.com/xioTechnologies/Calibration-Cube>

reference signal for misalignments between  $-10^\circ$  and  $+10^\circ$  is shown in Figure 6.4. A sensor calibrated against an attenuated reference would incorporate a sensitivity error equal to this attenuation if the misalignment is not accounted for within the calibration model.

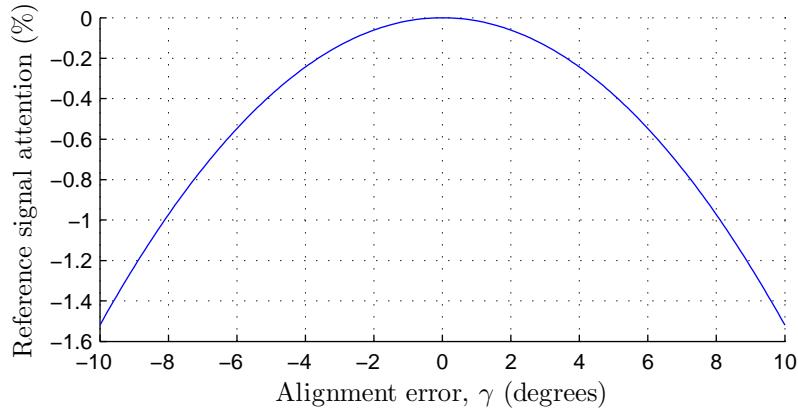


Figure 6.4: Relationship between calibration cube alignment error and attenuation of reference signal

The calibration of accelerometer alignment method in Section 6.5.3 indicated a mean alignment error of  $<0.7^\circ$  for the nine prototype x-BIMUs mounted on the calibration cube.

### 6.3.2 Calibration cube protractor

The calibration cube protractor is used in conjunction with the calibration cube to evaluate the static pitch/roll and heading error achieved by an accelerometer and magnetometer. The protractor is not required for the calibration. Figure 6.5 shows the calibration cube mounted on the protractor.

The protractor is comprised of a 290 mm diameter cog with 72 teeth, and a base plate incorporating a socket for the cog. The cog can be mounted in the socket at 72 different angular positions corresponding to a  $360^\circ$  rotation in  $5^\circ$  steps. The cog has four sockets that mate with any four corners of the calibration cube. The protractor may therefore be used to rotate the cube around all three axes.

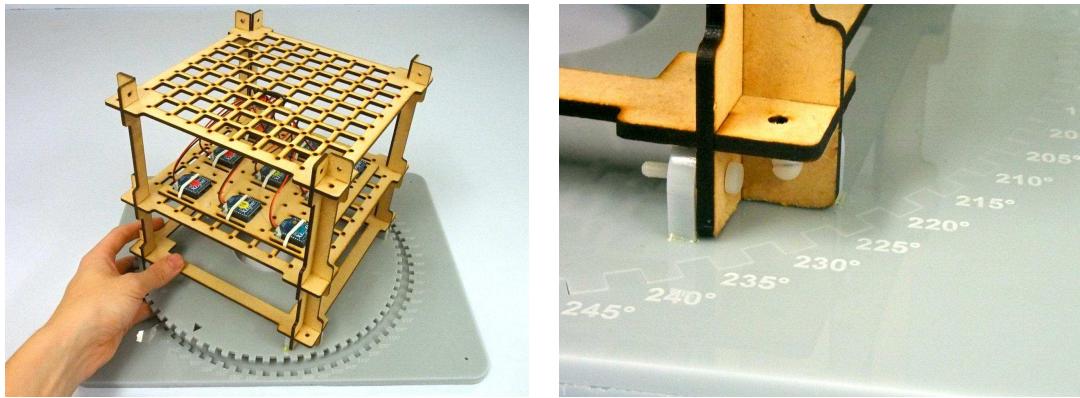


Figure 6.5: Calibration cube mounted on the protractor (left) and close up of cog teeth and engraved angular increments (right)

The manufacture process was essential in achieving the accuracy of the protractor. The cog and socket were laser cut from a single sheet of 5 mm acrylic plastic and the cog flipped within the socket so that the natural chamfers created by the laser cutter formed parallel surfaces. Theoretically, the protractor accuracy would be determined by the backlash between the cog and the socket. In practise, the finite accuracy of the laser cutter results in an elliptical cog and socket that achieve either a flush fit or a ‘press fit’ as the cog is rotated. This potentially improves accuracy by distributing geometric errors between the 72 teeth. An accelerometer was demonstrated as achieving 0.1° static pitch/roll accuracy through a 360° rotation using the protractor in Section 6.5.4; it can therefore be inferred that the protractor accuracy is <0.1°.

## 6.4 Gyroscope calibration

The gyroscope calibration model is described by Equations 6.2 and 6.3 where  $\omega$  is the calibrated gyroscope measurement in degrees per second,  ${}^C_\Omega \mathbf{R}$  is a rotation matrix describing the alignment of the gyroscope relative to the calibrated frame,  $\mathbf{u}_\omega$  is the uncalibrated gyroscope output in lsb,  $\mathbf{S}_\omega$  is the gyroscope sensitivity in lsb per degrees per second, and  $\mathbf{b}_\omega$  is gyroscope bias in lsb.  $\mathbf{S}_\omega$  is a diagonal

matrix as the calibration model does not account for cross-axis coupling or axis non-orthogonality.

$$\boldsymbol{\omega} = {}^C_{\Omega} \mathbf{R} \mathbf{S}_{\omega}^{-1} (\mathbf{u}_{\omega} - \mathbf{b}_{\omega}) \quad (6.2)$$

$$\begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix} = \begin{bmatrix} r_{\omega 11} & r_{\omega 12} & r_{\omega 13} \\ r_{\omega 21} & r_{\omega 22} & r_{\omega 23} \\ r_{\omega 31} & r_{\omega 32} & r_{\omega 33} \end{bmatrix} \begin{bmatrix} s_{\omega x} & 0 & 0 \\ 0 & s_{\omega y} & 0 \\ 0 & 0 & s_{\omega z} \end{bmatrix}^{-1} \left( \begin{bmatrix} u_{\omega x} \\ u_{\omega y} \\ u_{\omega z} \end{bmatrix} - \begin{bmatrix} b_{\omega x} \\ b_{\omega y} \\ b_{\omega z} \end{bmatrix} \right) \quad (6.3)$$

The above calibration model provides a calibrated measurement of angular velocity,  $\boldsymbol{\omega}$ , given the uncalibrated gyroscope output,  $\mathbf{u}_{\omega}$ , and known values of the fixed calibration parameters  ${}^C_{\Omega} \mathbf{R}$ ,  $\mathbf{S}_{\omega}$  and  $\mathbf{b}_{\omega}$ . These parameters are evaluated through a one-off calibration process summarised by the following steps.

1. Calculate the bias,  $\mathbf{b}_{\omega}$ , as the mean gyroscope output,  $\mathbf{u}_{\omega}$ , while the gyroscope is stationary.
2. Collect a calibration dataset representing the mean gyroscope output,  $\mathbf{u}_{\omega}$ , for the six orientations of the calibration cube relative to the rotational axis of a precision rate table to expose each axis to a known positive and negative reference of angular velocity  $\omega$ .
3. Calculate each axis sensitivity,  $s_{\omega x}$ ,  $s_{\omega y}$  and  $s_{\omega z}$ , as the average magnitude of the axis output when exposed the  $+\omega$  and  $-\omega$  reference.
4. Calculate the misalignment,  ${}^C_{\Omega} \mathbf{R}$ , using the previously determined values  $\mathbf{b}_{\omega}$  and  $\mathbf{S}_{\omega}$  and three orthogonal quantities of  $\mathbf{u}_{\omega}$  within the dataset that represent each axis begin exposed to a  $+\omega$  reference.

The following sections describe the complete gyroscope calibration process in detail.

### 6.4.1 Gyroscope bias calibration

The gyroscope bias parameters  $b_{\omega x}$ ,  $b_{\omega y}$  and  $b_{\omega z}$  are calculated as the mean uncalibrated gyroscope output over a period of several seconds while the device is stationary. The gyroscope bias is known to vary over time and with temperature though this is not accounted for in the calibration model. It is assumed that an AHRS algorithm will provide a means of updating  $\mathbf{b}_\omega$  during operation.

### 6.4.2 Gyroscope calibration dataset

The gyroscope calibration dataset requires a rate table to provide a known reference of angular velocity. Industrial rate tables such as the the Ideal Aerosmith 1270VS provide an accuracy of 0.1% over a range of angular velocities [130]. However, such equipment is not commonly available and the dataset requires only a single velocity which can be achieved using widely available record turntables. The Technics SL-1200MK2 turntable was chosen as it features a direct drive motor with quartz-phase-locked control to provide an accurate and repeatable angular velocity [131]. A turntable speed of 33<sup>1</sup>/<sub>3</sub> RPM was chosen for a reference angular velocity of  $\omega = 200^\circ/\text{s}$ . The accuracy was verified using a magnet and a Hall effect sensor to generate a square wave with a period equal to the angular velocity; a TTi TF930 frequency counter was used to measure the period with an accuracy of  $\pm 0.2$  parts per million (ppm) [132]. The turntable speed was measured as  $200.0023^\circ/\text{s}$ ; corresponding to an error of 12 ppm. This accuracy was maintained with a 5 kg load which is far greater than the expected load during calibration.

The dataset is collected by orientating the calibration cube on the turntable so that so that each axis is aligned with the vertical rotational axis and pointing either up or down; exposing each axis to a reference of  $+\omega$  and  $-\omega$ . This is represented by the six orientations shown in Figure 6.6. The mean uncalibrated gyroscope output,

$\mathbf{u}_\omega$ , is taken over a period of several seconds for each orientation to yield a dataset of six vector quantities. Each pair of measurements representing  $+\omega$  and  $-\omega$  for a given axis should be made contiguously with the calibration cube rotated around only a single axis relative to the turntable between the two orientations. This ensures that unexpected time-variant or temperature deviations are minimised and that a misalignment between the plater and rotational axis results in an equal attenuation of the  $+\omega$  and  $-\omega$  reference.

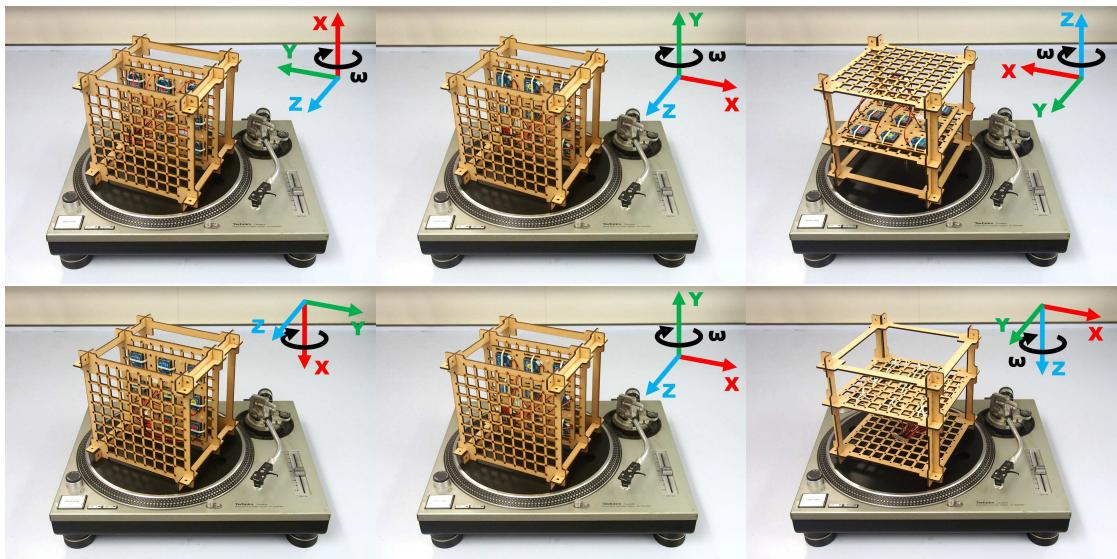


Figure 6.6: The gyroscope calibration dataset is represented by the six orientations of the calibration cube on the turntable to expose each axis to a  $+\omega$  and  $-\omega$  reference

### 6.4.3 Gyroscope sensitivity calibration

The gyroscope axis sensitivities  $s_{\omega_x}$ ,  $s_{\omega_y}$  and  $s_{\omega_z}$  are calculated as the average magnitude of the axis output when exposed to the  $+\omega$  and  $-\omega$  reference. This is described by Equation 6.4 where  $s_\omega$  is the axis sensitivity in lsb per degrees per second,  $u_{+\omega}$  and  $u_{-\omega}$  are the axis output in lsb when exposed to the  $+\omega$  and  $-\omega$  reference respectively, and  $\omega$  is the magnitude of the reference angular velocity

known to be 200°/s.

$$s_\omega = \frac{|u_{+\omega}| + |u_{-\omega}|}{2\omega} \quad (6.4)$$

#### 6.4.4 Gyroscope alignment calibration

The alignment of the gyroscope relative to the calibrated frame,  ${}^C_\Omega \mathbf{R}$ , is calculated using the values of the  $\mathbf{S}_\omega$  and  $\mathbf{b}_\omega$  previously determined in Sections 6.4.1 and 6.4.3 and the three orthogonal vector quantities of  $\mathbf{u}_\omega$  within the calibration dataset that represent each axis being exposed to a positive reference of angular velocity.  ${}^C_\Omega \mathbf{R}$  is initially approximated as  ${}^C_G \tilde{\mathbf{R}}$  as described by Equation 6.5 where  $\mathbf{u}_{\omega x}$ ,  $\mathbf{u}_{\omega y}$  and  $\mathbf{u}_{\omega z}$  are the gyroscope outputs when the positive reference is applied to the  $x$ ,  $y$  and  $z$  axes respectively. The three columns of  ${}^C_G \tilde{\mathbf{R}}$  are constructed as each measured direction of rotation for three orthogonal orientations of the calibration cube on the turntable.

$${}^C_\Omega \tilde{\mathbf{R}} = \begin{bmatrix} \|\mathbf{S}_\omega^{-1} (\mathbf{u}_{\omega x} - \mathbf{b}_\omega)\| \\ \|\mathbf{S}_\omega^{-1} (\mathbf{u}_{\omega y} - \mathbf{b}_\omega)\| \\ \|\mathbf{S}_\omega^{-1} (\mathbf{u}_{\omega z} - \mathbf{b}_\omega)\| \end{bmatrix}^T \quad (6.5)$$

${}^C_G \tilde{\mathbf{R}}$  can only be assumed to be an approximation of a rotation matrix because measurement errors may prevent it from being orthogonal.  ${}^C_\Omega \mathbf{R}$  is obtained from  ${}^C_G \tilde{\mathbf{R}}$  by computing the best-fit rotation matrix as described in Chapter 2.

### 6.5 Accelerometer calibration

The accelerometer calibration model is described by Equations 6.6 and 6.7 where  $\mathbf{a}$  is the calibrated accelerometer measurement in g,  ${}^A_\Omega \mathbf{R}$  is a rotation matrix describing the alignment of the accelerometer relative to the calibrated frame,  $\mathbf{u}_a$  is the uncalibrated accelerometer output in lsb,  $\mathbf{S}_a$  is the accelerometer sensitivity in lsb per g, and  $\mathbf{b}_a$  is accelerometer bias in lsb.  $\mathbf{S}_a$  is a diagonal

matrix as the calibration model does not account for cross-axis coupling or axis non-orthogonality.

$$\mathbf{a} = {}^C_A \mathbf{R} \mathbf{S}_a^{-1} (\mathbf{u}_a - \mathbf{b}_a) \quad (6.6)$$

$$\begin{bmatrix} a_x \\ a_y \\ a_z \end{bmatrix} = \begin{bmatrix} r_{a11} & r_{a12} & r_{a13} \\ r_{a21} & r_{a22} & r_{a23} \\ r_{a31} & r_{a32} & r_{a33} \end{bmatrix} \begin{bmatrix} s_{ax} & 0 & 0 \\ 0 & s_{ay} & 0 \\ 0 & 0 & s_{az} \end{bmatrix}^{-1} \left( \begin{bmatrix} u_{ax} \\ u_{ay} \\ u_{az} \end{bmatrix} - \begin{bmatrix} b_{ax} \\ b_{ay} \\ b_{az} \end{bmatrix} \right) \quad (6.7)$$

The above calibration model provides a calibrated measurement of acceleration,  $\mathbf{a}$ , given the uncalibrated accelerometer output,  $\mathbf{u}$ , and known values of the fixed calibration parameters  ${}^C_A \mathbf{R}$ ,  $\mathbf{S}_a$  and  $\mathbf{b}_a$ . These parameters are evaluated through a one-off calibration process summarised by the following steps.

1. Collect a calibration dataset representing the mean accelerometer output,  $\mathbf{u}_a$ , for the six orientations of the calibration cube on a level surface to expose each axis to a +1 g and -1 g reference.
2. Calculate each axis bias,  $b_{ax}$ ,  $b_{ay}$  and  $b_{az}$ , as the average of the axis output when exposed the +1 g and -1 g reference.
3. Calculate each axis sensitivity,  $s_{ax}$ ,  $s_{ay}$  and  $s_{az}$ , as the average magnitude of the axis output when exposed the +1 g and -1 g reference.
4. Calculate the misalignment,  ${}^C_A \mathbf{R}$ , using the previously determined values  $\mathbf{S}_a$  and  $\mathbf{b}_a$  and the three orthogonal quantities of  $\mathbf{u}_a$  within the dataset representing each axis begin exposed to a +1 g.

The following sections describe the accelerometer calibration process in detail and present a practical evaluation of the calibrated accelerometer performance. The calibrated performance is contrasted with the uncalibrated performance and that of the commercially available IMUs.

### 6.5.1 Accelerometer calibration dataset

The accelerometer calibration dataset is collected by orientating the calibration cube on a flat, level surface so that each axis is aligned with gravity and pointing either vertically up or down, exposing each axis to a reference of +1 g and -1 g. This dataset is represented by the six orientations of the calibration cube shown in Figure 6.7. The mean accelerometer output,  $\mathbf{u}_a$ , is taken over a period of several seconds for each orientation to yield a dataset of six vector quantities. Each pair of measurements representing +1 g and -1 g for a given axis should be made contiguously with the calibration cube rotated around only a single axis relative to the level surface between the two orientations. This ensures that unexpected time-variant or temperature deviations are minimised and that a misalignment between the level surface and direction of gravity results in an equal attenuation of the +1 g and -1 g reference.

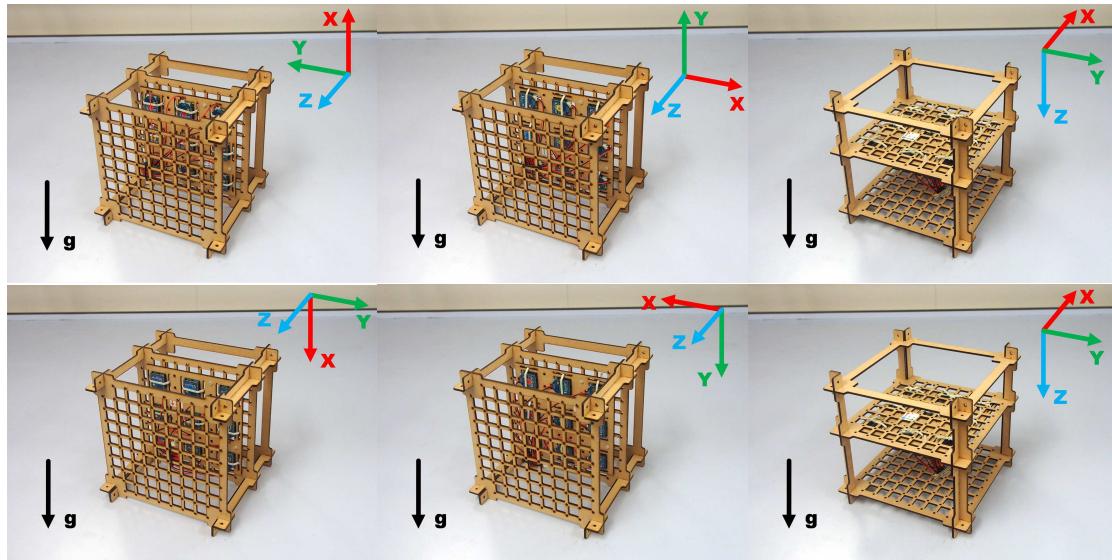


Figure 6.7: The accelerometer calibration dataset is represented by the six orientations of the calibration cube on a level surface to expose each axis to a +1 g and -1 g reference

### 6.5.2 Accelerometer sensitivity and bias calibration

The accelerometer axis sensitivities  $s_{ax}$ ,  $s_{ay}$  and  $s_{az}$  and biases  $b_{ax}$ ,  $b_{ay}$  and  $b_{az}$  are calibrated using the accelerometer calibration dataset and an industry-standard 6-point tumble test [120]. The bias of each axis is calculated as the average of the axis output when exposed to the +1 g and -1 g reference. This is described by Equation 6.8 where  $b_a$  is the axis bias in lsb, and  $u_{+g}$  and  $u_{-g}$  are the axis output in lsb when exposed to the +1 g and -1 g reference respectively.

$$b_a = \frac{u_{+g} + u_{-g}}{2} \quad (6.8)$$

The sensitivity of each axis is calculated as the average magnitude of the axis output when exposed to the +1 g and -1 g reference. This is described by Equation 6.9 where  $s_a$  is the axis sensitivity in lsb per g; and  $u_{+g}$  and  $u_{-g}$  are the axis output in lsb when exposed to the +1 g and -1 g reference respectively.

$$s_a = \frac{|u_{+g}| + |u_{-g}|}{2g} \quad (6.9)$$

### 6.5.3 Accelerometer alignment calibration

The alignment of the calibrated frame relative to the physical accelerometer,  ${}_A^C\mathbf{R}$ , is calculated using the values of the  $\mathbf{S}_a$  and  $\mathbf{b}_a$  previously determined in Section 6.5.2 and the three orthogonal vector quantities of  $\mathbf{u}_a$  within the calibration dataset that represent each axis when exposed to a +1 g reference.  ${}_A^C\mathbf{R}$  is initially approximated as Equation 6.10 where  ${}_A^C\tilde{\mathbf{R}}$  is an approximation of  ${}_A^C\mathbf{R}$ ; and  $\mathbf{u}_{ax}$ ,  $\mathbf{u}_{ay}$  and  $\mathbf{u}_{az}$  are the accelerometer outputs when the +1 g reference is applied to the  $x$ ,  $y$  and  $z$  axes respectively. In this equation, the three columns of  ${}_A^C\tilde{\mathbf{R}}$  are constructed as the directions of gravity measured by the accelerometer when gravity is aligned to

the each orthogonal axis of the calibration cube.

$${}^C_A \tilde{\mathbf{R}} = \begin{bmatrix} \|\mathbf{S}_a^{-1}(\mathbf{u}_{ax} - \mathbf{b}_a)\| \\ \|\mathbf{S}_a^{-1}(\mathbf{u}_{ay} - \mathbf{b}_a)\| \\ \|\mathbf{S}_a^{-1}(\mathbf{u}_{az} - \mathbf{b}_a)\| \end{bmatrix}^T \quad (6.10)$$

${}^C_A \tilde{\mathbf{R}}$  can only be assumed to be an approximation of a rotation matrix because measurement errors may prevent it from being orthogonal.  ${}^C_A \mathbf{R}$  is obtained from  ${}^C_A \tilde{\mathbf{R}}$  by computing the best-fit rotation matrix as described in Chapter 2.

#### 6.5.4 Calibrated accelerometer performance

The accelerometer performance was evaluated as the error in the angular measurement of inclination provided by the accelerometer when rotated around an axis perpendicular to gravity. Both uncalibrated and calibrated performance were evaluated for comparison. The angle of inclination for a rotation around a given axis can be calculated using Equations 6.11 to 6.14 where  $n_x$ ,  $n_y$  and  $n_z$  represent either the normalise uncalibrated or calibrated accelerometer measurement; and  $\theta_x$ ,  $\theta_y$  and  $\theta_z$  represent the angle of inclination for rotations around the accelerometer  $x$ ,  $y$  and  $z$  axis respectively.

$$\begin{bmatrix} n_x & n_y & n_z \end{bmatrix}^T = \begin{cases} \hat{\mathbf{u}} \\ \hat{\mathbf{a}} \end{cases}, \text{ where } \hat{\mathbf{u}} = \frac{\mathbf{u}}{\|\mathbf{u}\|} \hat{\mathbf{a}} = \frac{\mathbf{a}}{\|\mathbf{a}\|} \end{cases} \quad (6.11)$$

$$\theta_x = \text{atan2}(n_y, n_z) \quad (6.12)$$

$$\theta_y = \text{atan2}(n_z, n_x) \quad (6.13)$$

$$\theta_z = \text{atan2}(n_x, n_y) \quad (6.14)$$

The calibration cube protractor was used to rotate the cube 360° around each axis in 5° steps. At each step, the mean accelerometer output was obtained for approximately 2000 samples over 30 seconds to yield both uncalibrated and calibrated measurements for each accelerometer. The corresponding values of  $\theta_x$ ,  $\theta_y$  or  $\theta_z$  for each step would then be computed and compared with the angle indicated by the protractor to yield the error in the angular measurement of inclination provided by an accelerometer. Figure 6.8 shows the calibration cube protractor being used to collect the accelerometer evaluation dataset.

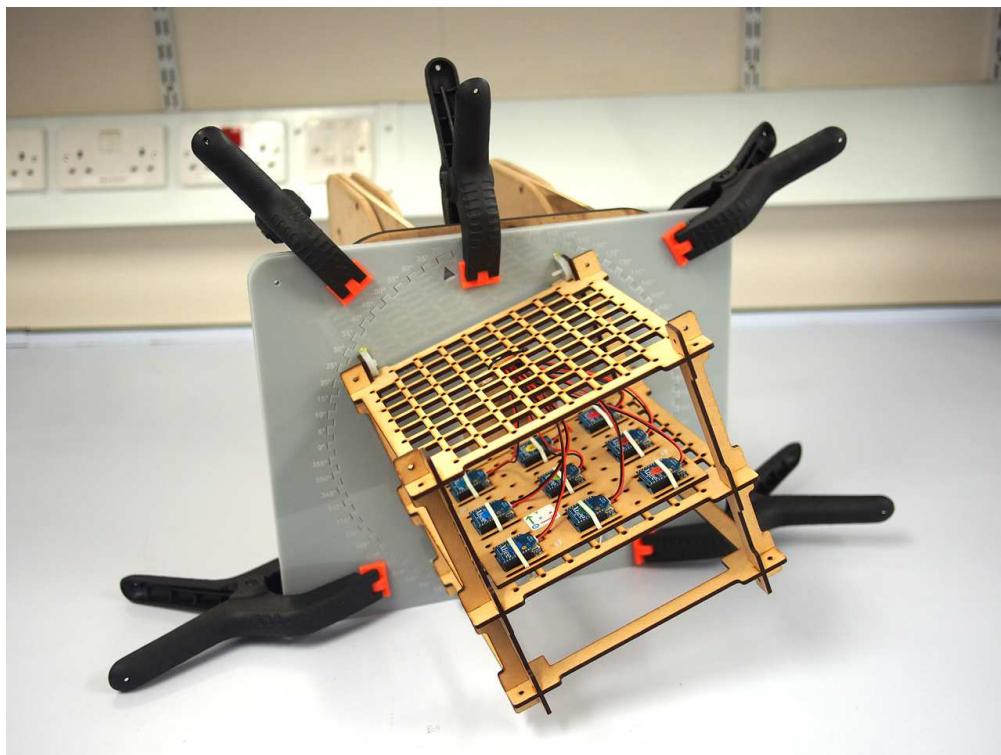


Figure 6.8: Calibrated accelerometer performance was evaluated by using the calibration cube protractor to rotate through precise inclinations

A misalignment between the zero position of the protractor and the direction of gravity introduced an offset to each accelerometer's measurement of inclination. This offset was removed by subtracting the mean angular error of all nine

accelerometers from each. Figure 6.9 shows the performance of the uncalibrated accelerometers. The top three plots indicate the error in each accelerometer's measurement of inclination for a  $360^\circ$  rotation around each of accelerometer axes. The distribution of these errors are shown for each accelerometer as bottom three box plots. Rotations around the  $z$  axis can be seen to result in the lowest error. This is expected because rotations around the  $x$  axis or  $y$  axis incorporate measurements from the accelerometer  $z$  axis which is indicated as providing a greater initial offset error in the MPU-6050 data sheet [121]. This is due to the physical structure of the MEM sensor. The worst case inclination measurement errors are  $\pm 6.5^\circ$ ,  $6.9^\circ$  and  $\pm 2.6^\circ$  for rotations around the  $x$ ,  $y$  and  $z$  axes respectively. These errors are three times greater than that of the lowest performing commercial IMU and demonstrate the need for calibration.

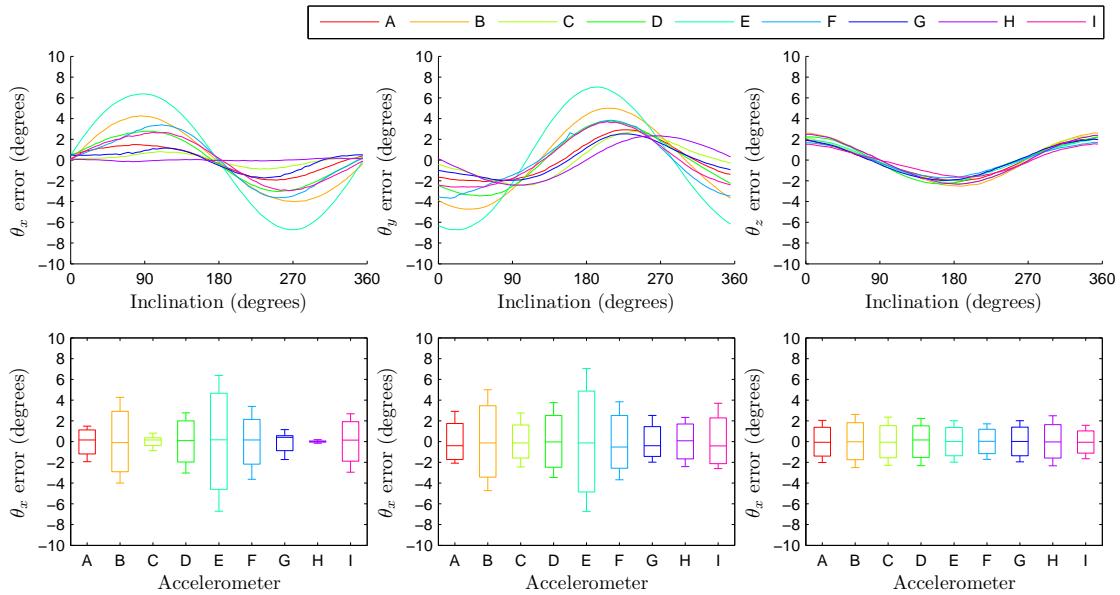


Figure 6.9: Error of each uncalibrated accelerometer's measurement of inclination for rotations around each axis. The bottom three box plots show the distribution of the error in the top three plots (whiskers indicate  $1.5 \times$  interquartile range).

Figure 6.10 shows the performance of the calibrated accelerometers. The worst case inclination measurement errors are  $\pm 0.35^\circ$ ,  $\pm 0.44^\circ$  and  $\pm 0.15^\circ$  for rotations around the  $x$ ,  $y$  and  $z$  axes respectively. This performance matches that of the highest performing commercial IMU; the 3DM-GX3-25 which is indicated as achieving an

accuracy of  $\pm 0.5^\circ$ .

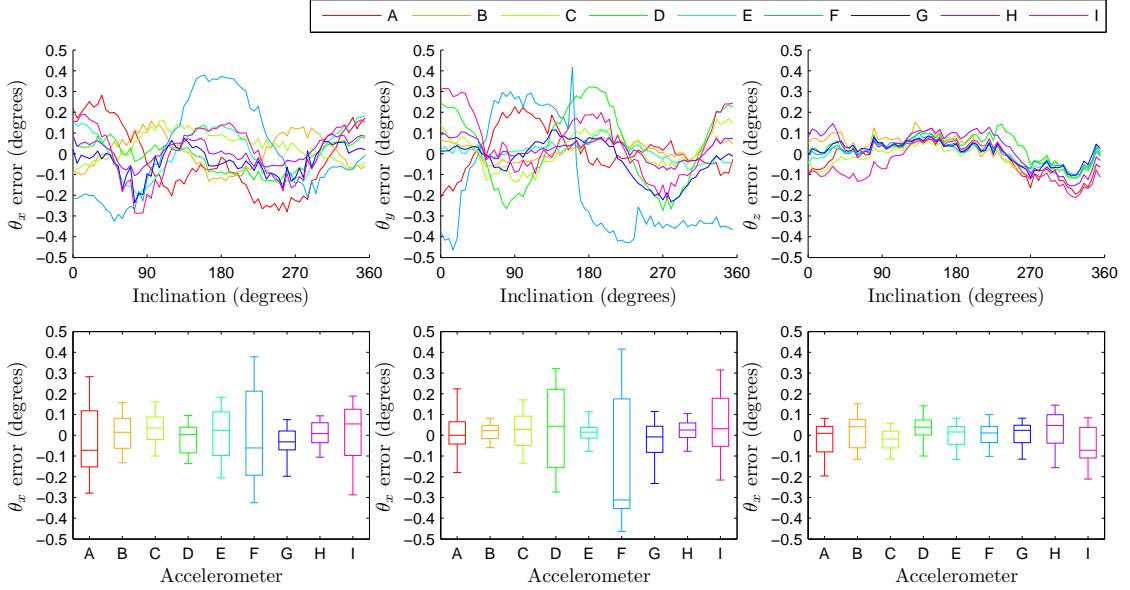


Figure 6.10: Error of each calibrated accelerometer's measurement of inclination for rotations around each axis. The bottom three box plots show the distribution of the error in the top three plots (whiskers indicate  $1.5 \times$  interquartile range).

## 6.6 Magnetometer calibration

The magnetometer calibration model is described by Equations 6.15 and 6.16 where  $\mathbf{m}$  is the calibrated magnetometer measurement in Guass,  $\mathbf{S}$  is the soft-iron matrix incorporating the individual axis sensitivities and cross-axis coupling,  $\mathbf{u}$  the calibrated magnetometer output in lsb, and  $\mathbf{h}$  is magnetometer bias in Gauss.

$$\mathbf{m} = \mathbf{S}\mathbf{u} - \mathbf{h} \quad (6.15)$$

$$\begin{bmatrix} m_x \\ m_y \\ m_z \end{bmatrix} = \begin{bmatrix} s_{xx} & s_{xy} & s_{xz} \\ s_{yx} & s_{yy} & s_{yz} \\ s_{zx} & s_{zy} & s_{zz} \end{bmatrix} \begin{bmatrix} u_x \\ u_y \\ u_z \end{bmatrix} - \begin{bmatrix} h_x \\ h_y \\ h_z \end{bmatrix} \quad (6.16)$$

Three separate calibration solutions to determine the soft-iron matrix,  $\mathbf{S}$ , and

hard-iron biases,  $\mathbf{h}$ , are proposed in the following sections. Each solution represents a different level of complexity with associated practical implications.

### 6.6.1 Magnetometer calibration datasets

Magnetometer calibration was evaluated for two scenarios to demonstrate ‘typical’ and ‘worst case’ magnetic distortions. The ‘typical’ distortions are represented by the magnetometer mounted on a PCB designed according to guidelines [122]; for example, positioning the magnetometer away from ferro-magnetic elements and high-current traces. The ‘worst case’ distortions are obtained from the same devices with an additional metal plate positioned directly above the magnetometer. The plate is the shielding on a radio module and represents a scenario relevant to any wireless application incorporating a magnetometer. Figure 6.11 shows the magnetometer in the both the ‘typical’ and ‘worst case’ magnetic distortion arrangements.

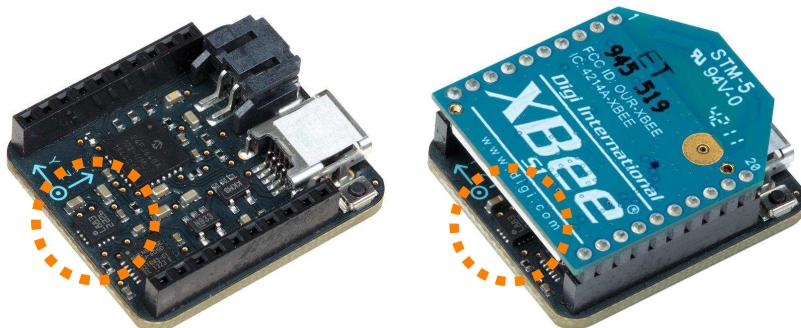


Figure 6.11: Magnetometer in the ‘typical’ (left) and ‘worst case’ (right) magnetic distortion arrangements. The magnetometer chip is indicated by the orange circle.

An immediate characterisation of the calibrated magnetometer can be achieved through a 3D plot of the measurements obtained for a set of Two Dimensional (2D) and 3D rotations within the ambient uniform magnetic field. Magnetometer measurements for an arbitrary 3D rotation through many different orientations would ideally form the surface of a sphere centred on the origin and measurements

for 2D rotations around each calibrated axis would form three circles on the surface of the sphere, each concentric with the axis of rotations. However, soft-iron distortions will distort the sphere into an ellipsoid and hard-iron biases will shift the centre away from the origin. This visual method of evaluation is demonstrated in some magnetometer manufacturer application notes [133].

Figure 6.12 shows ellipsoid plots of the uncalibrated measurements provided by one of the nine magnetometers mounted on the calibration cube for the ‘typical’ and ‘worst case’ distortions. The measurements for the arbitrary 3D rotation are shown as the grey line. The red, green and blue points are measurements for 2D rotations around the  $x$ ,  $y$  and  $z$  axis respectively. The  $\sim 70^\circ$  inclination of the Earth’s magnetic field in the UK results in a significant equatorial displacement of each of the coloured circles. Both plots indicate soft-iron and hard-iron distortions but the ‘worst case’ magnetic distortions causes significantly greater distortions of the ellipsoid and offset from the origin. The ‘worst case’ distortions are so great the origin is outside of the ellipsoid manifold, any calculation of heading as a direction around the origin would be meaningless.

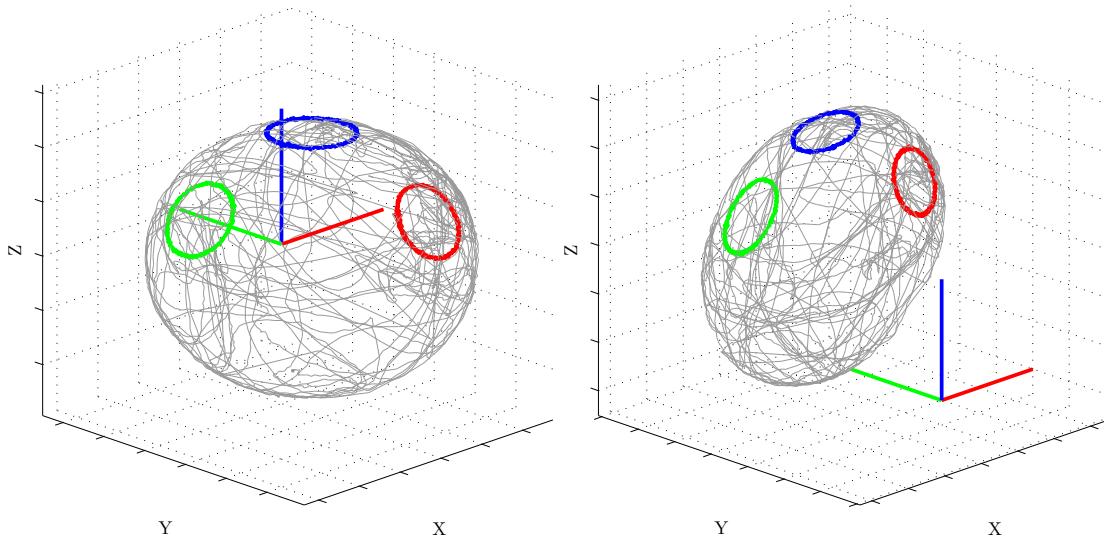


Figure 6.12: Example 3D plot of uncalibrated magnetometer measurements for ‘typical’ (left) and ‘worst case’ (right) magnetic distortion datasets

### 6.6.2 Calibrated magnetometer performance

The magnetometer performance was obtained as the error in the angular measurement of heading provided by the magnetometer when rotated around an axis perpendicular with the Earth's surface. The heading angle for a rotation around a given axis can be calculated using Equations 6.18 to 6.20 where  $v_x$ ,  $v_y$  and  $v_z$  describe a vector orthogonal to both the measured direction of the magnetic field provided by the magnetometer,  $\mathbf{m}$ , and the measured direction of gravity provided by the accelerometer,  $\mathbf{a}$ ; and  $\theta_x$ ,  $\theta_y$  and  $\theta_z$  are the heading angles for rotations around the  $x$ ,  $y$  and  $z$  axis respectively. It is necessary to incorporate the calibrated accelerometer measurements to eliminate sensitivity to the tilt of the magnetometer relative to the horizontal.

$$\begin{bmatrix} v_x & v_y & v_z \end{bmatrix}^T = \frac{\mathbf{a}}{\|\mathbf{a}\|} \times \frac{\mathbf{m}}{\|\mathbf{m}\|} \quad (6.17)$$

$$\theta_x = \text{atan2}(v_y, v_z) \quad (6.18)$$

$$\theta_y = \text{atan2}(v_z, v_x) \quad (6.19)$$

$$\theta_z = \text{atan2}(v_x, v_y) \quad (6.20)$$

The calibration cube protractor was used to rotate the cube  $360^\circ$  around each axis in  $5^\circ$  steps and the mean accelerometer and magnetometer outputs obtained for approximately 2000 samples over 30 seconds at each step. The corresponding values of  $\theta_x$ ,  $\theta_y$  or  $\theta_z$  for each step were then be computed and compared with the angle indicated by the protractor to yield the error in the angular measurement of heading provided by an magnetometer. A misalignment

between the protractor and magnetic North introduced an offset error to each magnetometer's measurement of heading. This offset was removed by subtracting the mean error of the nine measurements of heading for a given axis of rotation. Figure 6.13 shows the calibration cube protractor being used to collect this magnetometer evaluation data.

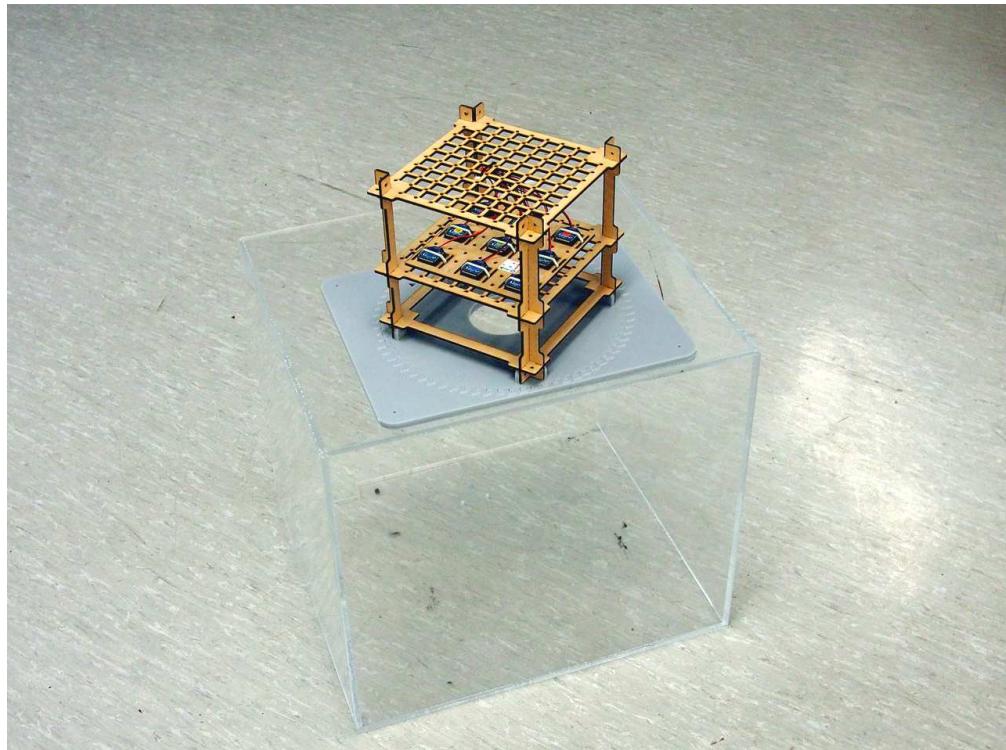


Figure 6.13: Calibrated magnetometer performance was evaluated by using the calibration cube protractor to rotate through precise heading angles

The top three plots of 6.14 show the error in each uncalibrated magnetometer's measurement of heading for a  $360^\circ$  rotation around each of axes for the 'typical' magnetic distortion scenario. The distribution of these errors are shown for each magnetometer as bottom three box plots. The distribution of the errors between  $\pm 180$  seen in Figure 6.14 demonstrates that magnetometer calibration is essential for the calculation of heading. The error in uncalibrated measurements of heading obtained for the 'worst case' dataset were all uniformly distributed between  $\pm 180$ .

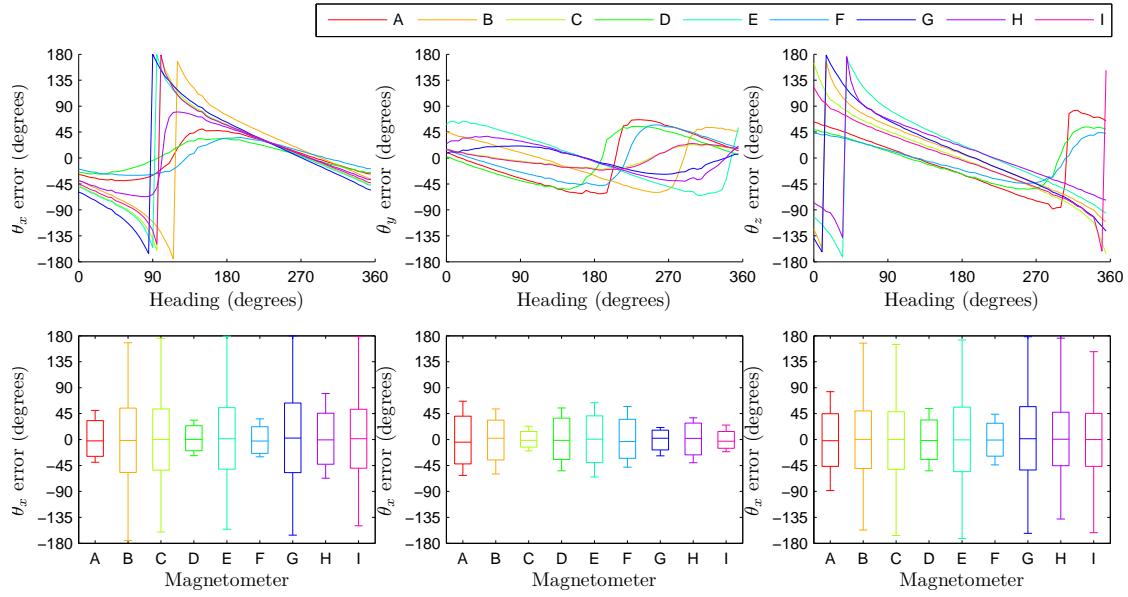


Figure 6.14: Error of each uncalibrated magnetometer’s measurement of heading for rotations around each axis for the ‘typical’ magnetic distortion dataset. The bottom three plots show the distribution of the error in the top three plots.

### 6.6.3 Offset calibration

Magnetometer offset calibration accounts for only the hard-iron distortions. The soft-iron matrix,  $\mathbf{S}$  is assumed to be an identity so that the magnetometer calibration model described by Equation 6.16 simplifies to Equation 6.21. The calibrated magnetometer measurement and hard-iron bias are both represented in units of lsb for this model.

$$\begin{bmatrix} m_x \\ m_y \\ m_z \end{bmatrix} = \begin{bmatrix} u_x \\ u_y \\ u_z \end{bmatrix} - \begin{bmatrix} h_x \\ h_y \\ h_z \end{bmatrix} \quad (6.21)$$

The hard-iron bias parameters  $h_x$ ,  $h_y$  and  $h_z$  are determined through a least-squares calculation based on a dataset of the uncalibrated magnetometer measurements when rotated through many different orientations within a uniform magnetic field. A calibrated magnetometer would measure a constant magnitude for all orientations so that all measurements would satisfy Equation 6.22 where  $v$

is the magnitude of the magnetic field in lsb. This can be expressed in vector form as shown by Equation 6.23.

$$(m_x - h_x)^2 + (m_y - h_y)^2 + (m_z - h_z)^2 = v^2 \quad (6.22)$$

$$\begin{bmatrix} -2m_x & 1 & -2m_y & 1 & -2m_z \end{bmatrix} \begin{bmatrix} h_x \\ h_x^2 \\ h_y \\ h_y^2 \\ h_z \\ h_z^2 \end{bmatrix} = v^2 - m_x^2 - m_y^2 - m_z^2 \quad (6.23)$$

The dimensions of Equation 6.23 can be extended to accommodate a dataset of  $n$  measurements as shown by Equation 6.24 which can be equivalently represented by Equation 6.25.

$$\begin{bmatrix} -2m_{x1} & 1 & -2m_{y1} & 1 & -2m_{z1} \\ -2m_{x2} & 1 & -2m_{y2} & 1 & -2m_{z2} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ -2m_{xn} & 1 & -2m_{yn} & 1 & -2m_{zn} \end{bmatrix} \begin{bmatrix} h_x \\ h_x^2 \\ h_y \\ h_y^2 \\ h_z \\ h_z^2 \end{bmatrix} = \begin{bmatrix} v^2 - m_{x1}^2 - m_{y1}^2 - m_{z1}^2 \\ v^2 - m_{x2}^2 - m_{y2}^2 - m_{z2}^2 \\ \vdots \\ v^2 - m_{xn}^2 - m_{yn}^2 - m_{zn}^2 \end{bmatrix} \quad (6.24)$$

$$\mathbf{M}\mathbf{h} = \mathbf{v} \quad (6.25)$$

The hard-iron biases can therefore be calculated as the solution to Equation 6.25 for a specified value of  $v$  and numerical dataset of  $n$  measurements. The value of  $\mathbf{h}$  may be obtained using the pseudoinverse of  $\mathbf{M}$  as described by Equation 6.31 for a calibration dataset of  $n > 6$ . Elements  $h_x$ ,  $h_y$  and  $h_z$  of vector  $\mathbf{h}$

are invariant to the value of  $v$ ; a value of 0 can be chosen for convenience.

$$\begin{aligned}\mathbf{h} &= \mathbf{M}^+ \mathbf{v} \\ &= (\mathbf{M}^T \mathbf{M})^{-1} \mathbf{M}^T \mathbf{v}\end{aligned}\tag{6.26}$$

### 6.6.3.1 Offset calibration performance

The nine magnetometers mounted on the calibration cube were calibrated as described above for both the ‘typical’ and ‘worst case’ scenarios. Figure 6.15 shows the ellipsoid plots for each dataset. The ellipsoid is centred on the origin which indicates that the hard-iron biases have been removed. The absence of the soft-iron sensitivity matrix,  $\mathbf{S}$ , means that the deformation from a sphere cannot be accounted for with this limited calibration model.

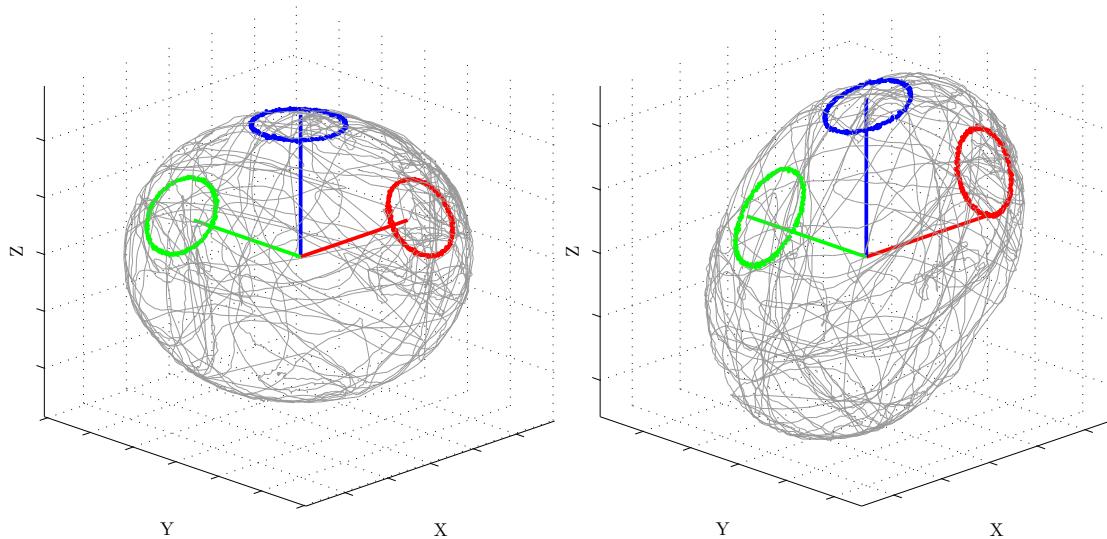


Figure 6.15: Example 3D plot of magnetometer measurements after offset calibration for ‘typical’ (left) and ‘worst case’ (right) magnetic distortion datasets

Figure 6.16 summarises the heading measurement error provided by the nine magnetometers for the ‘typical’ magnetic distortion scenario after offset calibration. The heading error archived for each axis is approximately  $\pm 6^\circ$ . This is greater than that of the worst performing commercial IMU, the UM6. However, this accuracy may be sufficient for some applications and offers the benefit of a

simple calibration model.

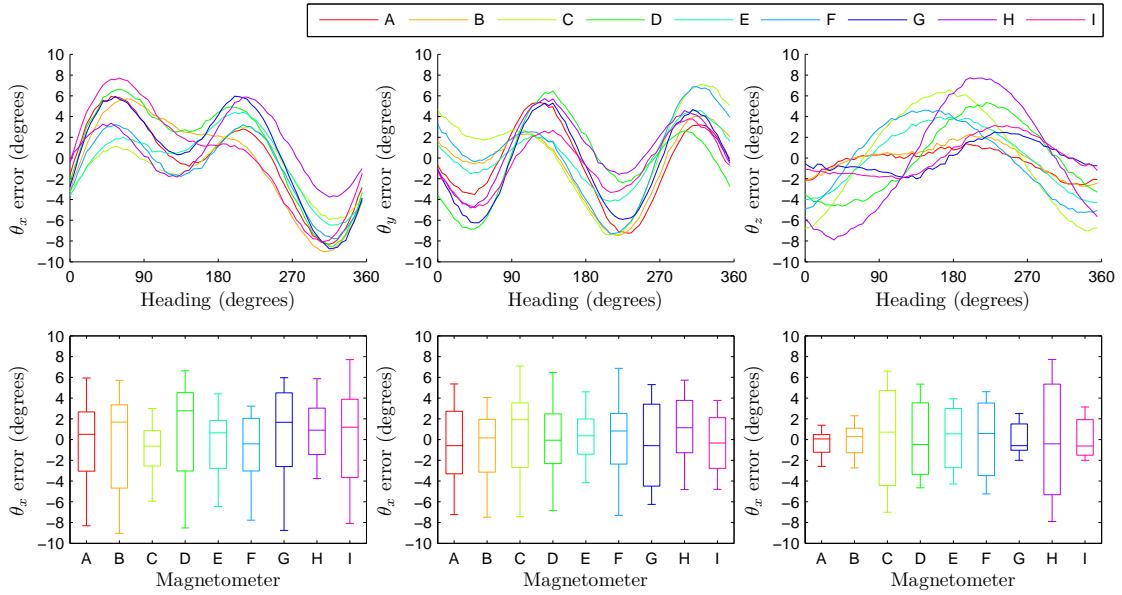


Figure 6.16: Heading measurement error provided by the nine magnetometers for the ‘typical’ magnetic distortion scenario after offset calibration.

Figure 6.17 summarises the heading measurement error provided by the nine magnetometers for the ‘worst case’ magnetic distortion scenario after offset calibration. The simplified offset calibration model is clearly insufficient for this scenario. The distortion of a sphere into the ellipsoid seen in Figure 6.15 causes the  $x$  axis to fall outside of the circle of data points describing  $\theta_x$ ; consequently it is not possible to resolve a heading and  $\theta_x$  is distributed between  $\pm 180^\circ$ .

#### 6.6.4 Ellipsoid calibration

Ellipsoid calibration requires the fitting an ellipsoid to the data points provided by the uncalibrated magnetometer when rotated through many different orientation in a homogeneous magnetic field. The soft-iron matrix,  $\mathbf{S}$ , and hard-iron vector,  $\mathbf{h}$ , may then be obtained as the geometric solution that equated the ellipsoid with a sphere. Ellipsoid fitting is the basis for many existing magnetometer calibration solutions [134]. The magnetometer calibration solution presented here builds on

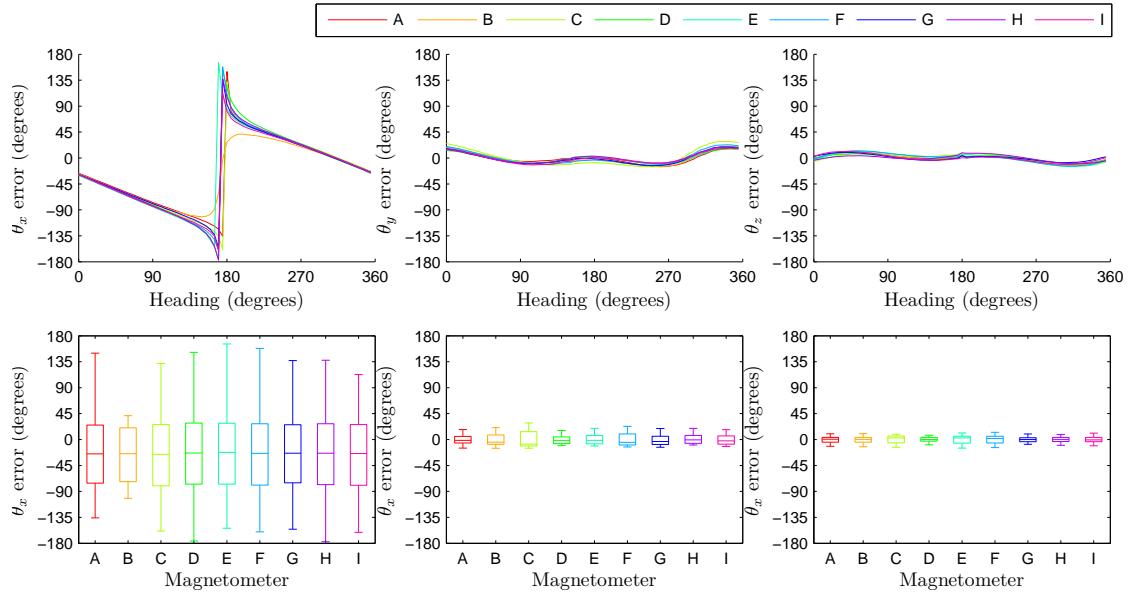


Figure 6.17: Heading measurement error provided by the nine magnetometers for the ‘worst case’ magnetic distortion scenario after offset calibration.

an ellipsoid fitting algorithm shared by Yury Petrov on the MATLAB file exchange [135] that evaluates the ellipsoid origin, radii and principle axes for a given dataset. The calibration parameters  $\mathbf{S}$  and  $\mathbf{h}$  are derived from these parameters. The ellipsoid fitting algorithms describes an ellipsoid using with the polynomial shown as Equations 6.27 and 6.28.

$$v_1x^2 + v_2y^2 + v_3z^2 + 2v_4xy + 2v_5xz + 2v_6yz + 2v_7x + 2v_8y + 2v_9z = 1 \quad (6.27)$$

$$\begin{bmatrix} x^2 & y^2 & z^2 & 2xy & 2xz & 2yz & 2x & 2y & 2z \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \\ v_3 \\ v_4 \\ v_5 \\ v_6 \\ v_7 \\ v_8 \\ v_9 \end{bmatrix} = 1 \quad (6.28)$$

The dimensions of Equation 6.28 can be extended to accommodate a dataset of  $n$  measurements as shown by Equation 6.29 which can be equivalently represented by Equation 6.30 where  $x$ ,  $y$ , and  $z$  correspond to calibrated magnetometer measurements  $m_x$ ,  $m_y$  and  $m_z$  in lsb.

$$\begin{bmatrix} x_1^2 & y_1^2 & z_1^2 & 2x_1y_1 & 2x_1z_1 & 2y_1z_1 & 2x_1 & 2y_1 & 2z_1 \\ x_2^2 & y_2^2 & z_2^2 & 2x_2y_2 & 2x_2z_2 & 2y_2z_2 & 2x_2 & 2y_2 & 2z_2 \\ \vdots & \vdots \\ x_n^2 & y_n^2 & z_n^2 & 2x_ny_n & 2x_nz_n & 2y_nz_n & 2x_n & 2y_n & 2z_n \end{bmatrix} \begin{bmatrix} A \\ B \\ C \\ D \\ E \\ F \\ G \\ H \\ I \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix} \quad (6.29)$$

$$\mathbf{M}\mathbf{h} = \mathbf{v} \quad (6.30)$$

The ellipsoid can therefore be calculated as the solution to Equation 6.30 for a numerical dataset of  $n$  measurements. The value of  $\mathbf{h}$  may be obtained using the

pseudoinverse of  $\mathbf{M}$  as described by Equation 6.31 for a calibration dataset of dataset of  $n > 9$ .

$$\begin{aligned}\mathbf{h} &= \mathbf{M}^+ \mathbf{v} \\ &= (\mathbf{M}^T \mathbf{M})^{-1} \mathbf{M}^T \mathbf{v}\end{aligned}\tag{6.31}$$

The polynomial parameters can be arranged in a matrix to describe the algebraic form of the ellipsoid shown Equation 6.32. The origin of the ellipsoid,  $\mathbf{c}$ , found as Equation 6.33.

$$\mathbf{T} = \begin{bmatrix} v_1 & v_4 & v_5 & v_7 \\ v_4 & v_2 & v_6 & v_8 \\ v_5 & v_6 & v_3 & v_9 \\ v_7 & v_8 & v_9 & -1 \end{bmatrix} \tag{6.32}$$

$$\mathbf{c} = \begin{bmatrix} v_7 & v_8 & v_9 \end{bmatrix} \begin{bmatrix} v_1 & v_4 & v_5 \\ v_4 & v_2 & v_6 \\ v_5 & v_6 & v_3 \end{bmatrix}^{-1} \tag{6.33}$$

The radii of the ellipsoid can be determined from the Eigenvalues of  $\mathbf{R}_c$  as described by Equation 6.35.  $\mathbf{T}_c$  describes the  $\mathbf{T}$  having been translated to the origin.  $\lambda_1$  to  $\lambda_2$  are the first three Eigenvalue of  $\mathbf{R}_c$ .

$$\mathbf{T}_c = \begin{bmatrix} t_{11} & t_{12} & t_{13} & t_{14} \\ t_{21} & t_{22} & t_{23} & t_{24} \\ t_{31} & t_{32} & t_{33} & t_{34} \\ t_{41} & t_{42} & t_{43} & t_{44} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix}^T \mathbf{T} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix} \tag{6.34}$$

$$\mathbf{R}_c = \begin{bmatrix} t_{11} & t_{12} & t_{13} \\ t_{21} & t_{22} & t_{23} \\ t_{31} & t_{32} & t_{33} \end{bmatrix} \frac{1}{t_{44}} \tag{6.35}$$

A rotation matrix describing the orientation of ellipsoid is created as a matrix where the Eigenvectors of  $\mathbf{R}_c$  make up the columns.

$$\mathbf{R} = \begin{bmatrix} \boldsymbol{\lambda}_1 & \boldsymbol{\lambda}_2 & \boldsymbol{\lambda}_3 \end{bmatrix} \quad (6.36)$$

The above ellipsoid fitting algorithm describes the ellipsoid as an origin,  $\mathbf{c}$ , radii,  $\mathbf{r}$  and a rotation  $\mathbf{R}$ . The soft-iron matrix,  $\mathbf{S}$ , and hard-iron vector,  $\mathbf{h}$ , can be obtained using Equations 6.37 and 6.38.  $r$  is the intensity of the Earth's field in Gauss within the calibration dataset, known to be approximately 0.5 Gauss in the UK [30].

$$\mathbf{S} = \mathbf{R} \begin{bmatrix} \frac{r}{r_x} & 0 & 0 \\ 0 & \frac{r}{r_x} & 0 \\ 0 & 0 & \frac{r}{r_x} \end{bmatrix} \mathbf{R}^T \quad (6.37)$$

$$\mathbf{h} = \mathbf{Sc} \quad (6.38)$$

#### 6.6.4.1 Ellipsoid calibration performance

The nine magnetometers mounted on the calibration cube were calibrated as described above for both the ‘typical’ and ‘worst case’ scenarios, Figure 6.18 shows the ellipsoid plots for each dataset. The calibrated measurements for both datasets now form the surface of a sphere centred on the origin. This confirms successful calibration of the relative axis sensitivities and the hard-iron bias. The red, green and blue circles describing the rotations around each axis are now concentric with each corresponding axis for the ‘typical’ dataset. However, these circles indicate that the magnetometers calibrated for the ‘worst case’ datasets incorporate a misalignment approximated by a  $-15^\circ$  rotation around the  $y$  axis.

Figure 6.19 summarises the heading measurement error provided by the nine magnetometers for the ‘typical’ magnetic distortion scenario after ellipsoid

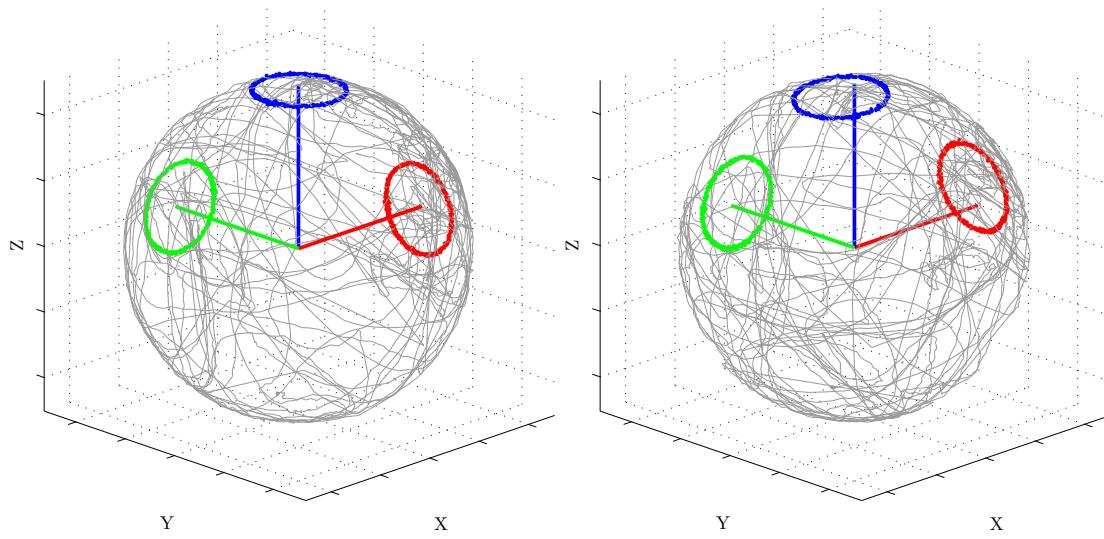


Figure 6.18: Example 3D plot of magnetometer measurements after ellipsoid calibration for ‘typical’ (left) and ‘worst case’ (right) magnetic distortion datasets

calibration. The error for rotations around each axis is approximately  $\pm 5^\circ$ , matching the performance of the worst performing commercial IMU, the UM6.

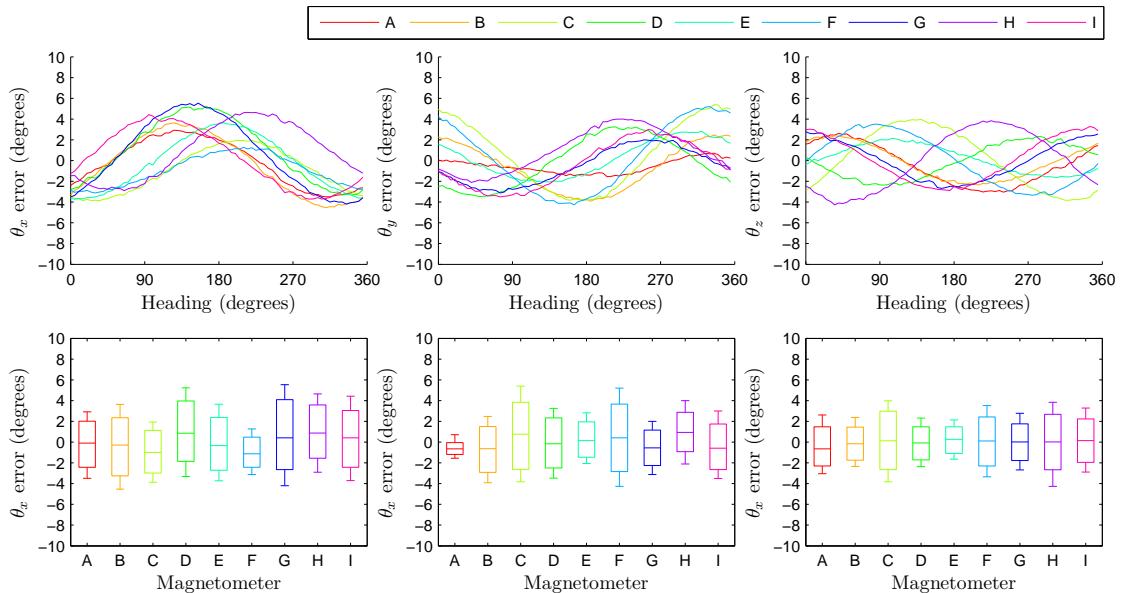


Figure 6.19: Heading measurement error provided by the nine magnetometers for the ‘typical’ magnetic distortion scenario after ellipsoid calibration.

The heading measurement errors for the ‘worst case’ magnetic distortion scenario are summarised in Figure 6.20. The misalignment around the  $y$  axis observed in Figure 6.18 indicates that the variance of errors in  $\theta_x$  and  $\theta_z$  will be greater than

that in  $\theta_y$ . This is confirmed by Figure 6.20. It is evident that ellipsoid calibration alone is not sufficient for the the ‘worst case’ magnetic distortion scenario.

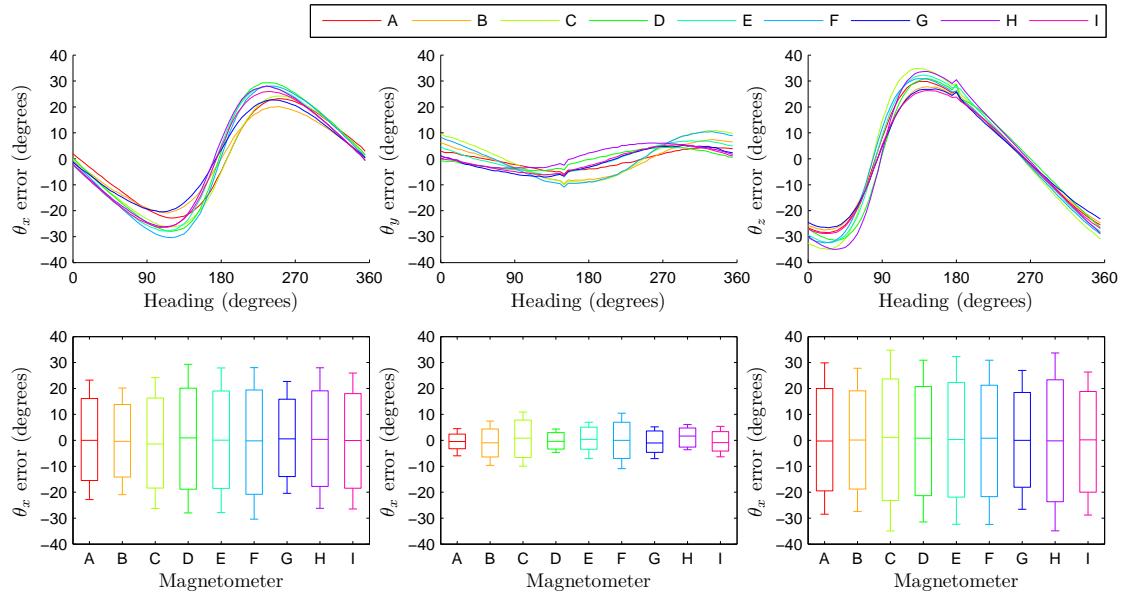


Figure 6.20: Heading measurement error provided by the nine magnetometers for the ‘worst case’ magnetic distortion scenario after ellipsoid calibration.

Ellipsoid calibration is able to successfully evaluate each axis sensitivity, cross-axis sensitivities and the hard-iron biases but it is not able to compensate for rotations around the origin. Consequently, the magnetometers calibrated for the ‘worst case’ magnetic distortions provide heading errors of up to  $30^\circ$ . Although performance for ‘typical’ magnetic distortions is better, it is limited by a failure to incorporate a misalignment between the magnetometer and accelerometer. The shortcomings of the ellipsoid calibration method can be overcome if the calibrated magnetometer axes can be brought into alignment with the calibrated frame.

### 6.6.5 Alignment calibration

The proposed alignment calibration method requires that the magnetometer has already been calibrated using the ellipsoid method to yield an initial soft-iron matrix,  $\hat{\mathbf{S}}$ , and hard-iron vector,  $\hat{\mathbf{h}}$ . The calibrated datasets describing rotations

around the calibration cube  $x$ ,  $y$  and  $z$  axis form the red, green and blue circles seen Figure 6.18 are used to calibrated for misalignment. The data points of each circle lie on a plane orthogonal to the axis of rotation. A rotation matrix describing the calibration cube relative to the ellipsoid-calibrated magnetometer can therefore be found as the solutions to these planes.

Equations 6.39 and 6.40 describe a plane where  $\alpha$ ,  $\beta$ ,  $\gamma$  and  $\delta$  are fixed coefficients and  $m_x$ ,  $m_y$  and  $m_z$  represents an ellipsoid-calibrated magnetometer measurement that sits on the plane.

$$\alpha m_x + \beta m_y + \gamma m_z + \delta = 0 \quad (6.39)$$

$$\begin{bmatrix} m_x & m_y & m_z & 1 \end{bmatrix} \begin{bmatrix} \alpha \\ \beta \\ \gamma \\ \delta \end{bmatrix} = 0 \quad (6.40)$$

The dimensions of this equation can be extended to incorporate the  $n$  measurements as described by Equations 6.41 and 6.42 where  $\mathbf{v}$  describes a vector that is orthogonal to the plane and of arbitrary magnitude.

$$\begin{bmatrix} m_{x1} & m_{y1} & m_{z1} \\ m_{x2} & m_{y2} & m_{z2} \\ \vdots & \vdots & \vdots \\ m_{xn} & m_{yn} & m_{zn} \end{bmatrix} \begin{bmatrix} \frac{\alpha}{-\delta} \\ \frac{\beta}{-\delta} \\ \frac{\gamma}{-\delta} \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix} \quad (6.41)$$

$$\mathbf{M}\mathbf{v} = \begin{bmatrix} 1 & 1 & \dots & 1 \end{bmatrix}^T \quad (6.42)$$

The axis orthogonal to the plane,  $\mathbf{v}$ , can be found as the solution to Equation 6.43 for a dataset of  $n$  measurements where  $n > 3$  where  $\mathbf{M}^+$  is the pseudoinverse of

$\mathbf{M}$ .

$$\begin{aligned}\mathbf{v} &= \mathbf{M}^+ \begin{bmatrix} 1 & 1 & \dots & 1 \end{bmatrix}^T \\ &= (\mathbf{M}^T \mathbf{M})^{-1} \mathbf{M}^T \begin{bmatrix} 1 & 1 & \dots & 1 \end{bmatrix}^T\end{aligned}\tag{6.43}$$

The vectors orthogonal to each planes are found as  $\mathbf{v}_x$ ,  $\mathbf{v}_y$  and  $\mathbf{v}_z$ . These vectors can be normalised and combined to approximate a rotation matrix as shown by Equation 6.44. A true rotation matrix,  $\mathbf{R}$ , can be obtained from  $\tilde{\mathbf{R}}$  by computing the best-fit rotation matrix as described in chapter 2.

$$\tilde{\mathbf{R}} = \begin{bmatrix} \frac{\mathbf{v}_x}{\|\mathbf{v}_x\|} & \frac{\mathbf{v}_y}{\|\mathbf{v}_y\|} & \frac{\mathbf{v}_z}{\|\mathbf{v}_z\|} \end{bmatrix}^T\tag{6.44}$$

The aligned soft-iron matrix,  $\mathbf{S}$ , and hard-iron vector,  $\mathbf{h}$ , can be obtained from the initial ellipsoid-calibrated parameters,  $\hat{\mathbf{S}}$  and  $\hat{\mathbf{h}}$  by correcting each using the calculated misalignment,  $\mathbf{R}$ , as described by Equations 6.45 and 6.46 .

$$\mathbf{S} = \mathbf{R}^T \hat{\mathbf{S}}\tag{6.45}$$

$$\mathbf{h} = \mathbf{R}^T \hat{\mathbf{h}}\tag{6.46}$$

### 6.6.5.1 Alignment calibration performance

The nine magnetometers mounted on the calibration cube were calibrated as described above for both the ‘typical’ and ‘worst case’ scenarios. Figure 6.21 shows the ellipsoid plots for each dataset. The  $-15^\circ$  misalignment around the  $y$  axis previously seen in Figure 6.18 for the ‘worst case’ scenario has been corrected for and the red, green and blue circles now lie concentric with each axis for both datasets.

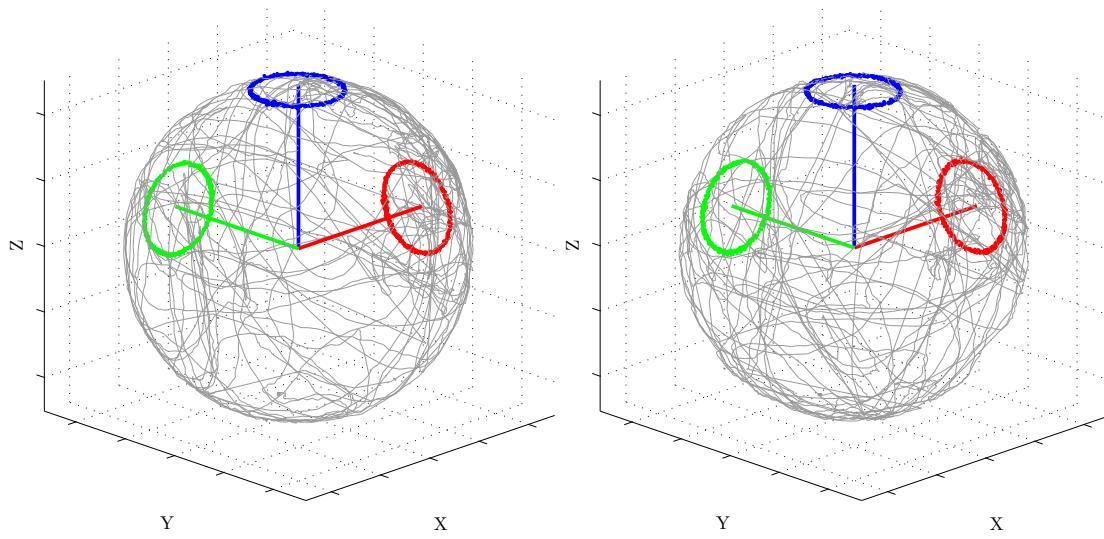


Figure 6.21: Example 3D plot of magnetometer measurements after alignment calibration for ‘typical’ (left) and ‘worst case’ (right) magnetic distortion datasets

Figure 6.22 summarises the heading measurement error provided by the nine magnetometers for the ‘typical’ magnetic distortion scenario after alignment calibration. The error remains appropriately equal for each axis of rotation but is the peak-to-peak error through a  $360^\circ$  is now half that of the ellipsoid-calibrated performance. The performance is now approximates that of the  $2^\circ$  accuracy achieved by the VN-100 commercial IMU. Geographic location is known to affect the heading accuracy achieved by a magnetometer; for example, a  $\pm 0.5^\circ$  at the equator would result in a  $\pm 1.5^\circ$  heading error in the UK (see Appendix C). This suggests that the approximate  $\pm 2^\circ$  heading accuracy demonstrated here may be closer to that of the high-end MTw and 3DM-GX3-25 if the test were to be repeated at the equator. However, it was not possible to verify this empirically, nor was it possible to confirm the test conditions of the heading accuracy specified by the commercial IMU.

The heading measurement errors for the ‘worst case’ magnetic distortion scenario are summarised in Figure 6.23. The error for rotations around each axis are now appropriately equal though the errors remain greater than those for the ‘typical’ magnetic distortions. Although some devices achieve the  $2^\circ$  accuracy achieved by

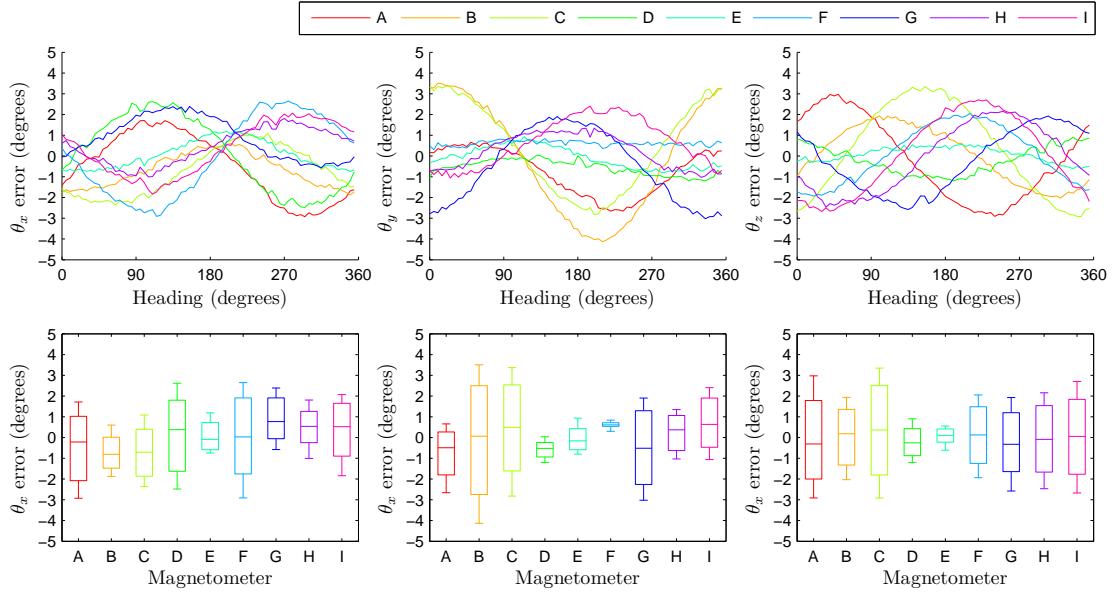


Figure 6.22: Heading measurement error provided by the nine magnetometers for the ‘typical’ magnetic distortion scenario after alignment calibration.

the VN-100, the performance is generally closer to that of the UM6 which may be sufficient for many applications. The contrast between the ‘typical’ and ‘worst case’ performance demonstrates the need to minimise magnetic distortions within the physical design of a device.

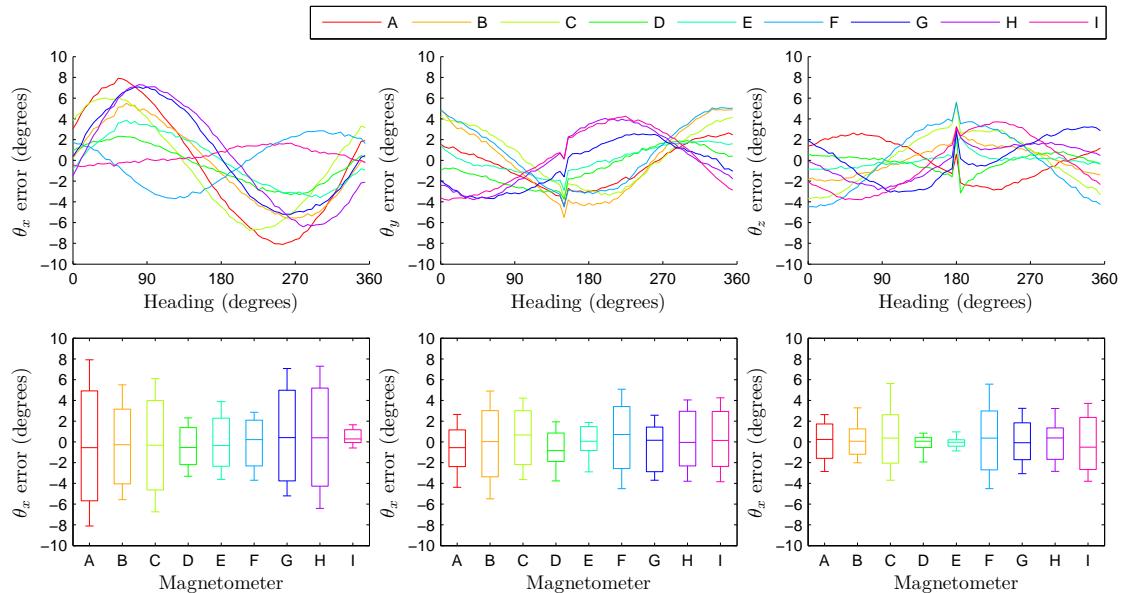


Figure 6.23: Heading measurement error provided by the nine magnetometers for the ‘worst case’ magnetic distortion scenario after alignment calibration.

### 6.6.6 Summary of magnetometer calibration solutions

Three magnetometer calibration solutions have proposed, each achieving a different level of accuracy and having different practical implications for the end application. Table 6.2 summarises the heading error for each of the solutions, calculated as the mean of the maximum of each device tested.

Calibration solution	Heading error	
	'Typical'	'Worst case'
Hard-iron	$\pm 5.5^\circ$	$\pm 55.5^\circ$
Ellipsoid	$\pm 3.3^\circ$	$\pm 21.1^\circ$
Ellipsoid + alignment	$\pm 1.9^\circ$	$\pm 3.9^\circ$

Table 6.2: Heading error for each magnetometer calibration solution achieved for 'typical' and 'worst case' magnetic distortion scenarios

The hard-iron calibration solution uses a simplified calibration model requiring only three parameters and implemented through three scalar operations. This may appeal to application with limited processing resources. The  $\pm 5.5^\circ$  accuracy achieved for 'typical' distortions is sufficient for many applications but hard-iron calibration alone is unable to compensate for the 'worst case' distortions. The  $\pm 3.3^\circ$  accuracy achieved through ellipsoid calibration is greater than that of the UM6 and approaches the accuracy of the VN-100. However, ellipsoid calibration alone is also insufficient for 'worst case' distortion scenarios.

Where as hard-iron and ellipsoid calibration require only the arbitrary rotation dataset, alignment calibration requires the additional three single-axis rotation datasets. This increases the complexity the calibration process but enables significantly improved performance. In the case of 'typical' distortions, the  $\pm 1.9^\circ$  accuracy achieved matches that of the VN-100. The  $\pm 3.9^\circ$  accuracy achieved for 'worst case' distortions is less accurate but still sufficient for many applications.

None of the calibration solutions were able to match the  $\pm 0.5^\circ$  heading accuracy of the 3DM-GX3-25. Although it may be assumed that the greatest accuracy

achieved,  $\pm 1.9^\circ$ , is limited by the geographical location of the tests, there was not an opportunity to demonstrate this empirically or to clarify the test conditions of the 3DM-GX3-25.

## 6.7 Simplifying sensor alignment arithmetic

The gyroscope, accelerometer and magnetometer calibration models account for a misalignment of each sensor to provide measurements aligned to a common calibrated frame. The associated rotation matrix multiplication more than triples the number of scalar arithmetic operations required by the gyroscope and accelerometer calibration models which may represent a significant burden for low-power or high sample rate applications. If the gyroscope and accelerometer can be assumed to be physically aligned then it is possible to eliminate the rotation matrix multiplications while maintaining a calibrated alignment between all three sensors.

In this simplified calibration model, the common calibrated frame becomes that of the physical accelerometer and the magnetometer calibration parameters are adjusted accordingly. The gyroscope is assumed to be physically aligned with the accelerometer so that the calibration models for the three sensors may be redefined as Equations 6.47 to 6.49.

$$\boldsymbol{\omega} = \boldsymbol{S}_{\omega}^{-1} (\boldsymbol{u}_{\omega} - \boldsymbol{b}_{\omega}) \quad (6.47)$$

$$\boldsymbol{a} = \boldsymbol{S}_a^{-1} (\boldsymbol{u}_a - \boldsymbol{b}_a) \quad (6.48)$$

$$\boldsymbol{m} = \tilde{\boldsymbol{S}} \boldsymbol{u}_m - \tilde{\boldsymbol{b}} \quad (6.49)$$

The adjusted soft-iron and hard-iron parameters  $\tilde{\mathbf{S}}$  and  $\tilde{\mathbf{h}}$  in Equation 6.49 are calculated as Equations 6.50 and 6.51 where  $\mathbf{R}$  is the rotation matrix describing the calibrated frame relative to the physical accelerometer, previously determined in Section 6.5.3.

$$\tilde{\mathbf{S}} = {}^C_A \mathbf{R}^T \mathbf{S} \quad (6.50)$$

$$\tilde{\mathbf{h}} = {}^C_A \mathbf{R}^T \mathbf{h} \quad (6.51)$$

This simplification significantly reduces both the computational load of the calibration models as well as the number of parameters needed to be stored in the memory of a calibrated device. Monolithic packages such as the MPU-6050 combine both the gyroscope and accelerometer on the same silicon die [136] to minimise misalignment between these two sensors and justify the assumption behind this simplification.

## 6.8 Conclusion

This chapter has presented a complete calibration solution for IMU incorporating a gyroscope, accelerometer and magnetometer. Practical performance was evaluated for the low-cost (<10 USD) MPU-6050 and HM-5883L sensors and benchmarked against four commercial IMUs price between 199 USD and 3000 USD. The results indicate a static pitch and roll accuracy matching that to the highest performing commercial IMU and a static heading accuracy matching that of a mid-range commercial IMU.

# Chapter 7

## Revised AHRS algorithm

### 7.1 Introduction

The original AHRS algorithm presented in Chapter 3 represents some of the first work of this project. The algorithm was derived analytically and tested using a fully-calibrated commercial IMU. The subsequent characterisation studies presented in Chapter 5 and calibration solutions in Chapter 6 demonstrated how modern low-cost MEMS sensors can achieve equivalent performance of such commercial IMUs. The revised AHRS algorithm presented in this chapter was designed to operate with these sensors based on the findings of Chapters 5 and 6. The algorithm design continued to evolve through the development of the IMU platforms and applications presented in Chapter 8; building upon the experiences gained from a diverse range of practical applications.

The revised AHRS algorithm builds on the quaternion implementation of the complementary filter proposed by Mahony *et al.* [55, 62]. As with other complementary filters, Mahony’s algorithm is dependent on a predefined fixed inclination of the magnetic field in the operating environment [63, 64]. This limitation is incompatible with many applications that must accommodate

magnetic variations [100, 101]. It is common for complementary filters to compensate for a gyroscope bias through integral feedback [62, 35]. However, this is arguably an inappropriate solution as integral wind-up dynamics will likely exceed those of the gyroscope bias and so risk worse performance. The revised algorithm addresses these shortcomings and introduces several new features, including:

- Fast initialisation behaviour
- Gyroscope bias compensation
- Magnetic distortion rejection
- Linear acceleration rejection
- Calculation of ‘zero-g’ and global accelerations

Although the calculation of ‘zero-g’ and global accelerations are not required for the computation of attitude or heading, these accelerations can only be obtained in conjunction with an AHRS algorithm and are often the ultimate quantities required by an application.

### 7.1.1 Algorithm overview

The algorithm provides a measurement of the IMU orientation relative to the Earth. The Earth coordinate system is defined according to the East, North, Up (ENU) convention with the  $y$  axis pointing in the direction of magnetic north and the  $z$  pointing directly up. Sensors must be calibrated to provide gyroscope measurements in units of radians per second and accelerometer measurements in units of g. The calibrated magnetometer units may be arbitrary. The algorithm can also operate without a magnetometer as described in the discussion at the end of the chapter. The complete algorithm is shown as the block diagram in Section 7.5. The following sections describe the individual processes that make up the revised AHRS algorithm.

## 7.2 Complementary filter estimator

The unnormalised quaternion  ${}^I_E \mathbf{q}$  describes the orientation of the Earth,  $E$ , relative to the IMU,  $I$ , and is obtained through the integration of the rate of change of the quaternion,  ${}^I_E \dot{\mathbf{q}}$ , as described by Equation 7.1.  ${}^I_E \dot{\mathbf{q}}$  is computed as Equation 7.2 [98] and is a function of the normalised quaternion,  ${}^I_E \hat{\mathbf{q}}$ ; the gyroscope measurement,  $\boldsymbol{\omega}$ ; and an error term,  $\mathbf{e}$ , scaled by the algorithm gain  $K$ .

$${}^I_E \mathbf{q} = \int {}^I_E \dot{\mathbf{q}} dt \quad (7.1)$$

$${}^I_E \dot{\mathbf{q}} = \frac{1}{2} {}^I_E \hat{\mathbf{q}} \otimes \begin{bmatrix} 0 & (\boldsymbol{\omega} - K\mathbf{e})^T \end{bmatrix} \quad (7.2)$$

The error term  $\mathbf{e}$  is determined from the accelerometer and magnetometer measurements with each sensor providing a separate error component. The accelerometer error component,  $\mathbf{e}_a$ , is calculated as the cross product of the normalised accelerometer measurement,  $\hat{\mathbf{a}}$ , and the direction of gravity assumed by  ${}^I_E \hat{\mathbf{q}}$  as shown by Equation 7.3 where  $q_w$ ,  $q_x$ ,  $q_y$  and  $q_z$  are the elements of  ${}^I_E \hat{\mathbf{q}}$ . The second operand of Equation 7.12 is equal to the third column of the rotation matrix obtained from  ${}^I_E \hat{\mathbf{q}}$ . Equations 7.1 to 7.3 alone describe an algorithm identical to [62]; subsistent algorithm processes represent novel development.

$$\mathbf{e}_a = \hat{\mathbf{a}} \times \begin{bmatrix} 2q_x q_z - 2q_w q_y \\ 2q_y q_z + 2q_w q_x \\ 2q_w^2 - 1 + 2q_z^2 \end{bmatrix}, \text{ where } \hat{\mathbf{a}} = \frac{\mathbf{a}}{\|\mathbf{a}\|} \quad (7.3)$$

The magnetometer error component,  $\mathbf{e}_m$ , is calculated as the cross product of the normalised magnetometer measurement,  $\hat{\mathbf{m}}$ , and the direction of East (perpendicular to magnetic north) assumed by  ${}^I_E \mathbf{q}$  as shown by Equation 7.4. The last operand of Equation 7.4 is equal to the negated second column of the rotation

matrix obtained from  ${}_E^I \hat{\mathbf{q}}$ . The cross product of  $\hat{\mathbf{a}}$  and  $\hat{\mathbf{m}}$  yields a vector that is orthogonal to both gravity as assumed by the accelerometer, and the measured direction of the magnetic field thus providing a vector measurement on a plane parallel with the Earth's surface that is insensitive to the orientation of the IMU and the inclination of the magnetic field, as in [137]. This is a feature of Digital Magnetic Compasses (DMCs) sometimes referred to as *tilt compensation*.

$$\mathbf{e}_m = \hat{\mathbf{a}} \times \hat{\mathbf{m}} \times \begin{bmatrix} -2q_w^2 + 1 - 2q_x^2 \\ -2q_xq_y + 2q_wq_z \\ -2q_xq_z - 2q_wq_y \end{bmatrix}, \text{ where } \hat{\mathbf{m}} = \frac{\mathbf{m}}{\|\mathbf{m}\|} \quad (7.4)$$

In normal operation,  $\mathbf{e}$  is calculated as the sum of  $\mathbf{e}_a$  and  $\mathbf{e}_m$ . However, these individual components are derived from the normalised measurements and so risk the numerical process being corrupted by a divided by zero.  $\mathbf{e}$  is therefore defined by Equation 7.5 where invalid sensor measurements are omitted from the algorithm update. If  $\mathbf{e}$  is set to zero then  ${}_E^I \hat{\mathbf{q}}$  is determined from the integrated gyroscope measurements alone. This conditional selection of  $\mathbf{e}$  is a prerequisite of some of the sensor conditioning described in Section 7.3.

$$\mathbf{e} = \begin{cases} \mathbf{e}_a + \mathbf{e}_m & \text{if } \|\mathbf{a}\| > 0 \text{ and } \|\mathbf{m}\| > 0 \\ \mathbf{e}_a & \text{else if } \|\mathbf{a}\| > 0 \\ \begin{bmatrix} 0 & 0 & 0 \end{bmatrix}^T & \text{else} \end{cases} \quad (7.5)$$

### 7.2.1 Algorithm gain and initialisation

The initial value of  ${}_E^I \hat{\mathbf{q}}$ , assumed to be an identity, will likely be incorrect upon initialisation of the algorithm. The low value of  $K$  and corresponding slow convergence of  ${}_E^I \hat{\mathbf{q}}$  desired for normal operation will cause this error to be sustained for a period of several seconds. This initialisation time can be reduced by ramping

down  $K$  from a high value to the value intended during normal operation. This process is described by Equation 7.6 where  $t_{init}$  is the initialisation period in seconds,  $K_{init}$  is the initial large value of  $K$  and  $K_{normal}$  is the value intended during normal operation where  $K_{normal} > K_{init}$ .

$$K = \begin{cases} K_{normal} + \frac{t_{init}-t}{t_{init}}(K_{init} - K_{normal}) & \text{if } t < t_{init} \\ K_{normal} & \text{else} \end{cases} \quad (7.6)$$

The algorithm output will be unreliable during initialisation. An application may determine whether the algorithm initialisation has completed as the boolean result of the condition  $t \geq t_{init}$ .

## 7.3 Sensor conditioning

The sensor conditioning processes described in this section are performed prior to the main complementary filter described in Section 7.2.

### 7.3.1 Gyroscope bias compensation

Gyroscope bias compensation is achieved by substituting  $\omega'$  in place of  $\omega$  in Equation 7.2 where  $\omega'$  is equal to the original gyroscope measurement with an estimated bias,  $\omega_{bias}$ , subtracted as described by Equation 7.7.

$$\omega' = \omega - \omega_{bias} \quad (7.7)$$

$\omega_{bias}$  is computed as the low-pass filtered result of  $\omega$  as described in Equation 7.8 where  $f_c$  is the corner frequency in Hz and  $p$  is a value dynamically determined as

either 1 or 0 to enable the integrator only when the IMU is stationary

$$\boldsymbol{\omega}_{bias} = 2\pi f_c \int p\boldsymbol{\omega}.dt \quad (7.8)$$

$p$  is determined as Equation 7.9 where the function  $f_b(\boldsymbol{\omega}, \omega_{min})$  that computes the time that the magnitude of each element of  $\boldsymbol{\omega}$  has been below  $\omega_{min}$ .  $\omega_{min}$  is the minimum angular velocity below which the bias estimator can be enabled.  $t_b$  is the minimum stationary period after which the bias estimator will be enabled.

$$p = \begin{cases} 1 & \text{if } f_b(\boldsymbol{\omega}, \omega_{min}) > t_b \\ 0 & \text{else} \end{cases} \quad (7.9)$$

### 7.3.2 Magnetic distortion rejection

Magnetic measurements that deviate from the expected intensity of the Earth's magnetic field are rejected to prevent magnetic distortions from corrupting the algorithm output. This is achieved by substituting  $\mathbf{m}'$  in place of  $\mathbf{m}$  in Equations 7.4 and 7.5.  $\mathbf{m}'$  is defined by Equation 7.10 where  $m_{min}$  and  $m_{max}$  are the specified minimum and maximum magnitudes of the Earth's magnetic field in units of the magnetometer. The selection of  $\mathbf{m}'$  as zero works in conjunction with the conditional selection of  $\mathbf{e}$  in Equation 7.5 to omit the magnetometer from the algorithm update.

$$\mathbf{m}' = \begin{cases} \mathbf{m} & \text{if } m_{min} < \|\mathbf{m}\| < m_{max} \\ \begin{bmatrix} 0 & 0 & 0 \end{bmatrix}^T & \text{else} \end{cases} \quad (7.10)$$

### 7.3.3 Linear acceleration rejection

Linear acceleration rejection is achieved in a similar way to [56]. The accelerometer is only omitted if the measured magnitude of gravity exceeds limits for a specified

period of time. The process is described by 7.11 where the function  $f(\mathbf{a}, g_{min}, g_{max})$  calculates the period of time that the condition of  $-g_d < (\|\mathbf{a}\| - 1) < g_d$  has been false.

$$\mathbf{a}' = \begin{cases} \begin{bmatrix} 0 & 0 & 0 \end{bmatrix}^T & \text{if } f(\mathbf{a}, g_d) > t_a \\ \mathbf{a} & \text{else} \end{cases} \quad (7.11)$$

## 7.4 ‘Zero-g’ and global acceleration calculation

The ‘zero-g’ acceleration,  $\mathbf{a}_{zero}$ , is the accelerometer measurement with gravity removed. It is calculated as the accelerometer measurement,  $\mathbf{a}$ , minus the 1 g offset dependent on the orientation of the IMU as shown by Equation 7.12 where  $q_w$ ,  $q_x$ ,  $q_y$  and  $q_z$  are the elements of  ${}^I_E\hat{\mathbf{q}}$ . The second operand of Equation 7.12 is equal to the third column of the rotation matrix obtained from  ${}^I_E\hat{\mathbf{q}}$ .

$$\mathbf{a}_{zero} = \mathbf{a} - \begin{bmatrix} 2q_x q_z - 2q_w q_y \\ 2q_y q_z + 2q_w q_x \\ 2q_w^2 - 1 + 2q_z^2 \end{bmatrix} \quad (7.12)$$

The global acceleration,  $\mathbf{a}_{global}$ , is the acceleration of the IMU in the Earth coordinate system as defined by the ENU convention. It is calculated as Equation 7.13.

$$\begin{bmatrix} 0 & \mathbf{a}_{global}^T \end{bmatrix} = {}^I_E\hat{\mathbf{q}} \otimes \begin{bmatrix} 0 & \mathbf{a}_{zero}^T \end{bmatrix} \otimes {}^I_E\hat{\mathbf{q}}^* \quad (7.13)$$

## 7.5 Algorithm block diagram

Figure 7.1 shows a block diagram of the complete AHRS algorithm and indicates the connection between the separate processes described in Sections 7.2 to 7.4.

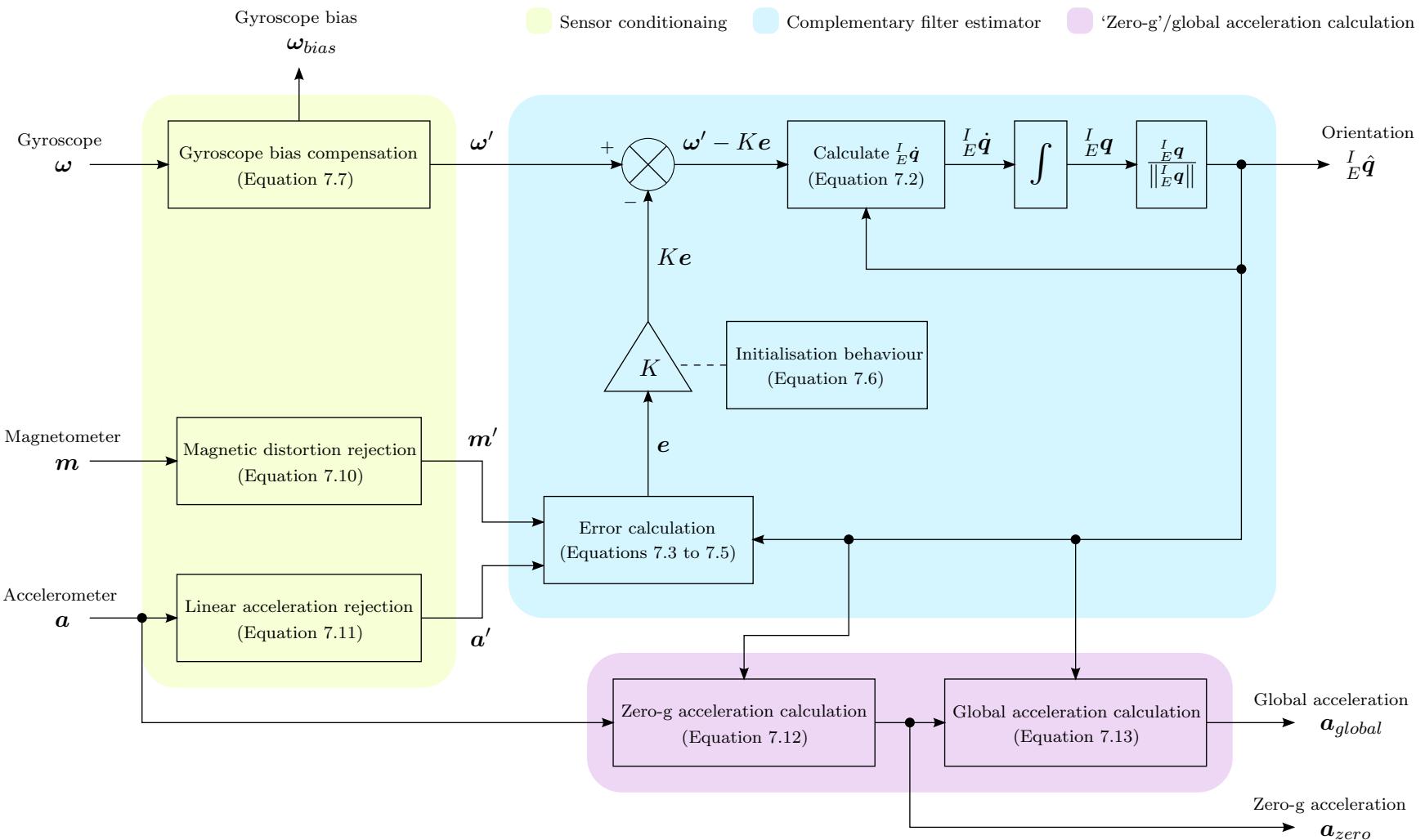


Figure 7.1: Complete AHRS block diagram

## 7.6 Discussion

The algorithm computes the error in  $\boldsymbol{\omega}'$  from the accelerometer and magnetometer through a cross product operation, the reason for this may not be immediately obvious. In the case of the accelerometer, the error is determined as the difference between the direction of gravity measured by the accelerometer,  $\hat{\mathbf{a}}$ , and the direction of gravity assumed by  ${}^I_E \hat{\mathbf{q}}$ . The cross product yields  $\mathbf{e}_a$  as a vector with a magnitude equal to the sine of the angular error in  ${}^I_E \hat{\mathbf{q}}$  and a direction orthogonal to both  $\hat{\mathbf{a}}$  and the direction of gravity assumed by  ${}^I_E \hat{\mathbf{q}}$ .  $\mathbf{e}_a$  therefore describes the necessary ‘adjustment’ to  ${}^I_E \hat{\mathbf{q}}$  applicable to the dimensions of  $\boldsymbol{\omega}'$ . In the case of the magnetometer,  $\mathbf{e}_m$  represents an equivalent adjustment though its direction is always aligned with gravity to prevent the magnetometer from influencing the pitch and roll components of orientation.

The sine relationship defining  $\|\mathbf{e}\|$  suggests a potential flaw in the algorithm. Angular errors exceeding  $90^\circ$  would yield an increasingly small feedback term. An error of  $180^\circ$  would result in no feedback and the algorithm would never converge. In practice, it is assumed that angular errors in  ${}^I_E \hat{\mathbf{q}}$  will always be  $<90^\circ$  after initialisation and an appropriate selection of  $K_{init}$  will avoid the hazard associated with an initial error of  $180^\circ$ .

### 7.6.1 Algorithm gain

The algorithm gain,  $K$ , controls the rate at which  ${}^I_E \hat{\mathbf{q}}$  converges to an orientation assumed by the accelerometer and magnetometer. A large value of  $K$  will mean an increased rate of convergence but also an greater sensitivity to linear accelerations and temporal magnetic distortions. A small value of  $K$  will mean that the  ${}^I_E \hat{\mathbf{q}}$  is predominately determined by the gyroscope but will risk an accumulating error if  $K$  is not sufficiently large enough to compensate for gyroscope measurement

errors. A minimum value  $K$  is therefore limited by the accuracy of the gyroscope, if  $K = 0$  then  ${}^I_E\hat{\mathbf{q}}$  is determined from the integrated gyroscope measurements alone. An optimal value  $K$  is that found to provide sufficient attenuation of the errors imposed by the accelerometer and magnetometer whilst remaining large enough that gyroscope drift is compensated for.

The algorithm process described by Equations 7.1 to 7.5 is a first-order proportional feedback system where  $\|\mathbf{e}\| \approx \frac{1}{\pi}$  radians for small angular errors in  ${}^I_E\hat{\mathbf{q}}$ . It is therefore possible to approximate performance in terms of  $K$  analytically. For example, it can be shown that  $K < \pi$  is necessary for stability and that the algorithm will converge within 2% of the final output in  $\frac{4\pi}{K}$  seconds [138]. The dynamics may be expressed in the frequency domain to show that  ${}^I_E\hat{\mathbf{q}}$  incorporates the accelerometer and magnetometer data through a low-pass filter with a corner frequency of  $\frac{K}{2\pi^2}$  Hz. This may provide a convenient means of choosing  $K$  if the frequency characteristics of an application are known, as in [57, 61, 56]. In practice, an appropriate value of  $K$  is typically found through manual adjustments.  $K_{normal} = 0.5$  was found to provide suitable performance for most applications.  $K_{init} = 10$  and  $t_{init} = 3$  were found to provide reliable initialisation behaviour. The large value of  $K_{init}$  deliberately causes a momentary instability to escape an initial angular error in  ${}^I_E\hat{\mathbf{q}}$  of  $\sim 180^\circ$ .

### 7.6.2 Gyroscope bias compensation

The gyroscope bias is estimated each time the IMU is detected as being stationary. Some manufacturers suggest similar mechanisms where measurements below a threshold are zeroed [139] or steady-state signals are sampled as the bias [140]. Such solutions would risk corrupting small amplitude signals and result in undesirable step changes of the gyroscope bias. The bias compensation proposed in this chapter is not subject to these shortcomings. Some applications may involve no

natural stationary periods. In such scenarios, bias compensation may still effective if an initial stationary period can be imposed to allow the system to ‘warm up’ prior to use.

The gyroscope bias compensation parameters include the filter corner frequency,  $f_c$ , the minimum angular velocity,  $\omega_{min}$ , below which the bias estimator can be enabled and the minimum stationary period,  $t_b$ , after which the bias estimator will be enabled. A corner frequency of  $f_c = 0.05$  Hz achieves convergence (within 2% of final value) within 13 seconds yet is relatively low to reduce the risk of attenuating low amplitude motion that may falls below  $\omega_{min}$ . A value of  $\omega_{min}$  may be determined as the expected variation in the gyroscope bias during operation. For example, the thermal response and random walk investigations presented in Chapter 5 indicate that a threshold of  $\omega_{min} = 4^\circ/\text{s}$  would be sufficient. The value should include a margin to account unexpected noise within the application; for example, mechanical vibrations. A period threshold of  $t_b = 2$  seconds would be sufficient to prevent the bias compensation of activating during zero-crossing in most applications.

### 7.6.3 Magnetic distortion rejection

The magnetic distortion rejection parameters,  $m_{min}$  and  $m_{max}$  define the minimum and maximum valid intensity for the ambient magnetic field. Default values of  $m_{min} = 0.22$  Gauss and  $m_{max} = 0.67$  Gauss may be assumed as these represent the natural limits of the Earth’s magnetic field [30]. More robust performance can be achieved if a reduced range can be specified representative of a specific operating environment.

### 7.6.4 Linear acceleration rejection

The linear acceleration rejection parameters include the period threshold,  $t_a$ , and gravity deviation,  $g_d$ . Values of  $t_a = 100\text{ms}$  and  $g_d = 0.1 \text{ g}$  were found to compensate for linear accelerations without being too sensitivity that accelerometer measurements are rejected unnecessarily.

### 7.6.5 Algorithm outputs

The algorithm provides  ${}^I_E \hat{\mathbf{q}}$  describing the orientation of the Earth relative to the IMU. Many applications will require a measurement of the IMU relative to the Earth,  ${}^E_I \hat{\mathbf{q}}$ , which is obtained as the conjugate  ${}^I_E \hat{\mathbf{q}}^*$ . An equivalent Euler angle or rotation matrix may be obtained using the methods described in Chapter 2.

The algorithm also provides the estimated gyroscope bias,  $\boldsymbol{\omega}_{bias}$ . This may be useful to processes external to the AHRS algorithm that require a bias-compensated gyroscope measurement. The provision of  $\boldsymbol{\omega}_{bias}$  also enables the gyroscope bias to be saved to non-volatile memory so that the ‘tracked’ value can be maintained even when the system is shut down.

The calculation of ‘zero-g’ accelerations,  $\mathbf{a}_{zero}$ , and global accelerations,  $\mathbf{a}_{global}$ , can only be achieved in conjunction with an AHRS algorithm.  $\mathbf{a}_{global}$  is the second derivative of position in the Earth coordinate system. Integrating this would yield a measurement of velocity; integrating the velocity would yield the displacement of the IMU. The ability to measure displacement is of benefit to many applications. In practice, this can only be successful if an exponentially increasing error inherent to numerical integration process can be compensated for. For example, pedestrian dead-reckoning applications achieve this through zero velocity detection [3].

### 7.6.6 Omitting the magnetometer

In some applications it is desirable to omit the magnetometer; for example, if the magnetic environment is known to be unreliable. This may be achieved by substituting zeros in place of the magnetometer measurement,  $\mathbf{m}$ . Without the magnetometer, the heading component of  ${}^I_E \hat{\mathbf{q}}$  is determined from the gyroscope alone. The heading will initially be relative to the direction of the IMU immediately after initialisation but will drift over time. The pitch and roll components of  ${}^I_E \hat{\mathbf{q}}$  will be unaffected and still provide an absolute measurement relative to the Earth's surface. The calculation of  $\mathbf{a}_{zero}$  and  $\mathbf{a}_{global}$  will also be unaffected.

## 7.7 Conclusion

The revised AHRS algorithm presented in this chapter builds on the findings of the sensor characterisation and calibration work presented in Chapters 5 and 6. It was developed simultaneously with the IMU platforms and applications and applications in the next chapter and incorporates features that address practical requirements, verified through these empirical works.



# **Chapter 8**

## **IMU platforms and applications**

### **8.1 Introduction**

This chapter describes the development of IMU platforms and the broad applications they have facilitated. These platforms take advantage of modern MEMS devices, the calibration solutions presented in Chapter 6, and the AHRS algorithm presented in chapter 7. The following three sections (8.2 to 8.3) summarise the development of three IMU platforms developed through this research project; the: x-IMU, x-BIMU and x-OSC. Each platform fulfils a different design specification and together have facilitated a wide range of applications and research projects. Section 8.5 summarises projects that have utilised the platforms for specific applications in collaboration with other academic researchers. Section 8.6 section presents a selection of user applications that have utilised the IMU platforms within commercial or academic research projects. The chapter concludes with a description of the future developments scheduled for for the IMU platforms and applications.

## 8.2 x-IMU: A versatile data acquisition platform

The x-IMU was developed in 2010 to provide a versatile IMU and data acquisition platform for use in a wide range of potential applications. In the four years since, the hardware and firmware has continued to evolve to take advantage of modern MEMS sensors and introduce new functionality based on user feedback. The original design was intended to meet the following specification.

### x-IMU design specification

- Calibrated gyroscope, accelerometer and magnetometer with on-board AHRS
- Real-time wired and wireless communication
- Autonomous data logging
- Input/Output (I/O) interface for external sensors and output devices

#### 8.2.1 x-IMU overview

The x-IMU is equally applicable to real-time and autonomous data logging applications. Real-time communication is supported through USB for wired applications and Bluetooth for wireless. Autonomous data logging is supported by an on-board SD card and several features including a Real-Time Clock (RTC) and sleep timer with motion trigger wake-up. An eight channel auxiliary port provides a configurable interface for external analogue and digital sensors or output devices. Figure 8.1 shows the x-IMU board alone (left) and enclosed within its plastic housing with a 1000 mAh battery with a total size of  $57 \times 38 \times 21$  mm and weight of 49 g.

Each x-IMU is calibrated during manufacture and individual sensor calibration



Figure 8.1: x-IMU board alone (left) and enclosed in its plastic housing with 1000 mAh battery (right)

parameters saved to on-board memory. During normal operation, the sensors are continuously sampled and the on-board AHRS algorithm updated at 512 Hz. An extensive set of internal settings can be managed using the x-IMU software to configure the device behaviour and the send rate of individual packet types up to 512 Hz.

### 8.2.1.1 x-IMU hardware

The x-IMU board dimensions are  $33 \times 42$  mm and it weights 12 g. Figure 8.2 shows an annotated top and bottom view with key hardware components are described below.

- (a) **dsPIC33FJ128GP804:** 16-bit, 40 Million Instructions Per Second (MIPS) general purpose Digital Signal Controller (DSC) with 16 kB of Random-Access Memory (RAM) and 128 kB of program space [141].
- (b) **Crystal oscillator:** 32.768 kHz crystal provides a time base for the RTC sample timer.
- (c) **USB bridge:** Future Technology Devices International (FTDI) FT232R USB bridge. See Section 8.2.2 for more information.

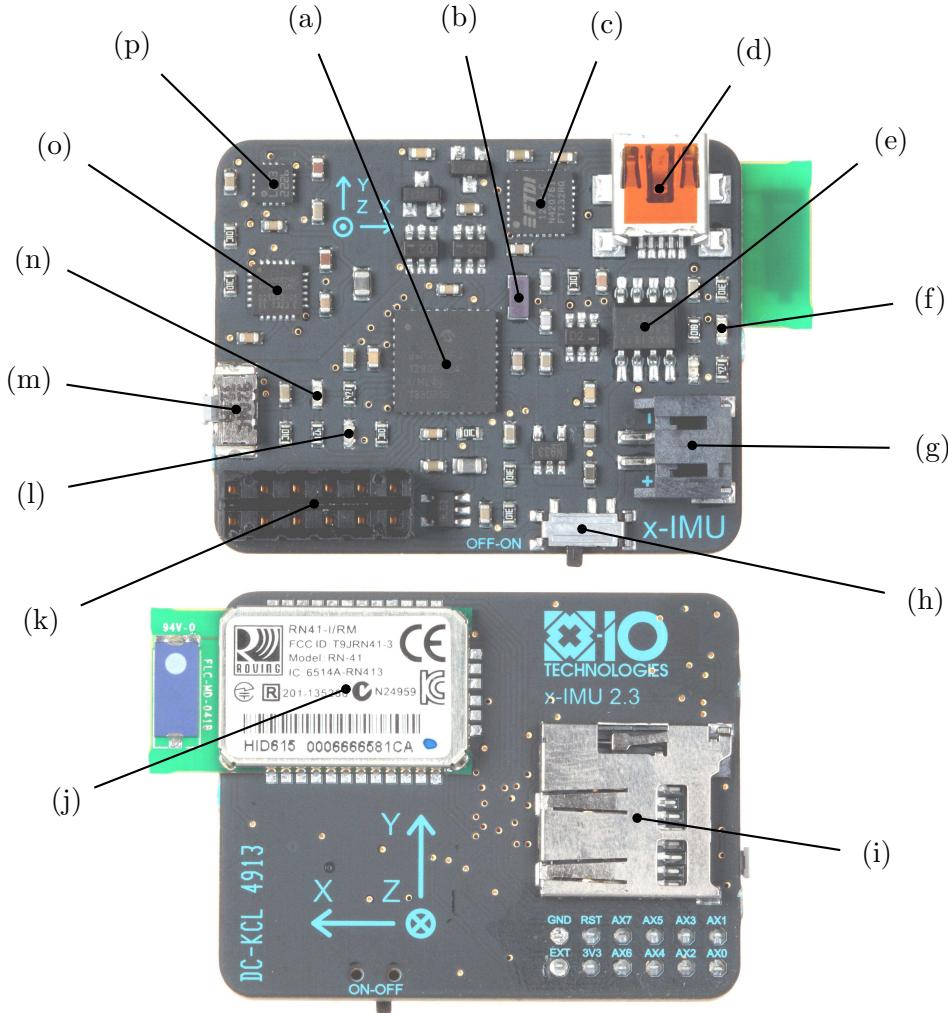


Figure 8.2: x-IMU top and bottom view with labelled key hardware components

- (d) **USB connector:** Provides interface for real-time wired communication as well as power for the device and battery charging.
- (e) **Maxim MAX1811:** Fully integrated Lithium-ion charger specifically designed for charging from a USB power supply [142].
- (f) **Bluetooth LED:** Blue Light-Emitting Diode (LED) that indicates the connection status of Bluetooth.
- (g) **Battery connector:** Socket for 3.7 V Lithium-ion cell. Battery voltage is measured on-board.
- (h) **Power switch:** Slide switch for switching the device on and off or selecting

the auxiliary port power input

- (i) **SD card socket:** Micro SD card socket supports FAT16 and FAT32 formats for capacities up to 32 GB.
- (j) **Bluetooth:** Microchip RN-41 Bluetooth module. See Section 8.2.2 for more information.
- (k) **Auxiliary port:** Eight channel configurable auxiliary port also includes power I/O pins. See Section 8.2.4 for more information.
- (l) **Status LED:** Green LED indicates the status of the x-IMU.
- (m) **Command button:** Configurable command button with a difficult mode to toggle sleep mode.
- (n) **SD card LED:** Amber LED that indicates the connection status of SD card.
- (o) **InvenSense MPU-6050:**  $\pm 2000^\circ/\text{s}$ , 16-bit resolution gyroscope;  $\pm 16 \text{ g}$ , 16-bit resolution accelerometer; and 16-bit thermometer [121].
- (p) **Honeywell HMC5883L:**  $\pm 8.1 \text{ Gauss}$ , 12-bit resolution magnetometer [122].

### 8.2.1.2 x-IMU software

The x-IMU Graphical User Interface (GUI) is a Windows application that provides an interface for configuring internal settings and displaying real-time data. Figure 8.3 shows a screen shot of the GUI displaying internal sensor measurements in 2D plots and the AHRS data as a 3D representation. The software also provides tools for synchronisation of real-time clock, firmware updates, inertial and magnetic calibration tools and data logging to Comma-Separated Values (CSV) files for exporting data to software such as MATLAB and Microsoft Excel. Figure 8.3 shows the x-IMU GUI with gyroscope and accelerometer measurements being displayed as 2D plots, and AHRS data being displayed as a 3D representation.

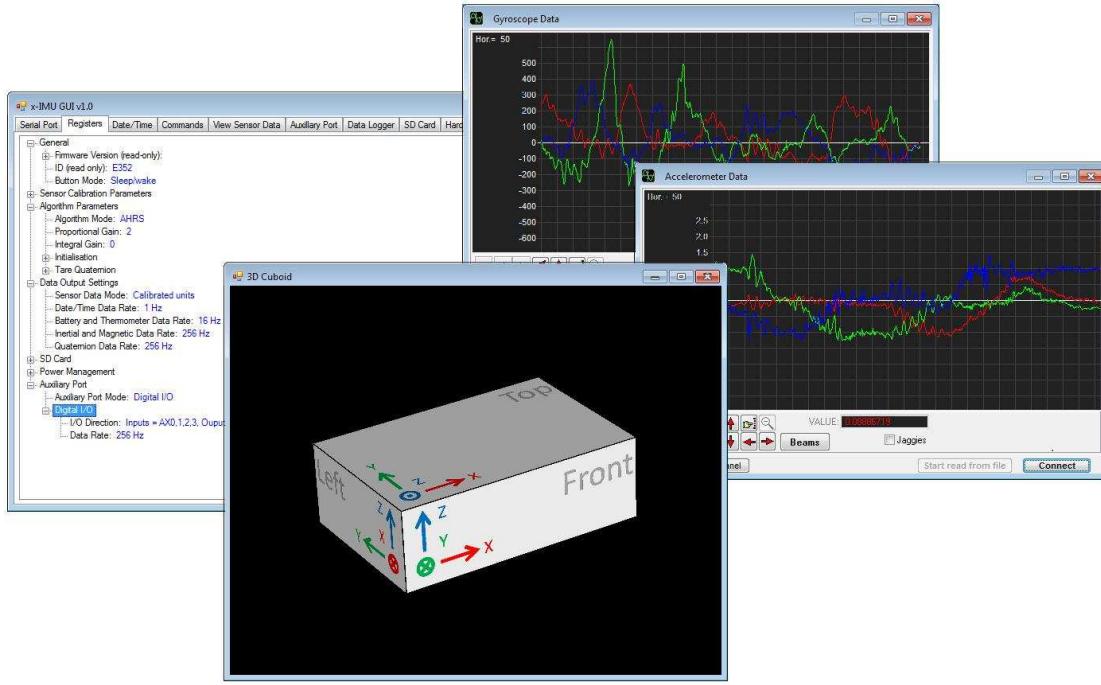


Figure 8.3: x-IMU GUI displaying real-time data with 2D and 3D graphics

The x-IMU GUI is open-source<sup>1</sup> and incorporates a generic C# Application Programming Interface (API) for the x-IMU that can be reused by other C# projects. There are also interface libraries available for C++<sup>2</sup> and Android<sup>3</sup>. A MATLAB library provides tools for importing and organising logged data acquired via the SD card.

### 8.2.2 Real-time wired and wireless communication

The x-IMU enables wireless communication via USB, Bluetooth or serial through the the axillary port. USB communication is enabled through a FTDI FT232R chip to maximise compatibility with FTDI drivers are available for Windows, OS X, Linux [143] and Android [144]. The RN-41 Bluetooth module is a class 1 device with a range of 100 m [145]. It uses the Serial Port Profile (SPP) Bluetooth profile eliminates the need for host drivers for compatibility with a wide range of platforms

<sup>1</sup><https://github.com/xioTechnologies/x-IMU-GUI>

<sup>2</sup><https://github.com/xioTechnologies/x-IMU-Arduino-Example>

<sup>3</sup><https://github.com/xioTechnologies/x-IMU-Android-Example>

such laptops and smart phones. The auxiliary port can be configured as a serial interface for communication with embedded systems. Settings support hardware flow control and standard baud rates.

### 8.2.3 Autonomous data logging

A feature of the x-IMU that has proved to be essential to many application is its ability to operate as an autonomous data logger. There are several features that facilitate this operation: SD card data logging, real-time clock and calendar, battery level monitoring and automated sleep/wake mechanisms. The x-IMU can be configured to log continuously to create a single file for the entire period, the x-IMU will automatically shut down when the battery reaches a specified minimum level. Alternately, the motion trigger wake-up and sleep timer used to automatically enter sleep mode during period of inactivity. Each time the device wake up, a new file is created on the SD card, time-stamped with the date and time. This may significantly extend the battery life for some applications. The x-IMU user manual [146] provides suggestions of how internal settings can be configured to minimise power consumption.

### 8.2.4 Auxiliary port

The auxiliary port is a 12 pin socket that provides access to power and eight general purpose analogue/digital I/O pins. The auxiliary port can be configured in several different modes including: analogue inputs, digital I/O, Pulse-Width Modulation (PWM), serial and external sleep/wake mode. Analogue and digital input modes allows data from external sensors to be sent in real-time or logged to the SD card. PWM and digital output modes allow real-time control of wide range of device, for example, LEDs or vibration motors for haptic feedback. The

external sleep/wake mode allows external controls to be used to enter/exit sleep mode. This has been of use to under water applications that require the x-IMU to be semi-permanently sealed within water proof housing.

## 8.3 x-BIMU: A low-power, wireless IMU for real-time applications

The x-BIMU was developed in 2012 to provide a more versatile wireless alternative to the x-IMU. The x-IMU uses Bluetooth which imposes several limitations. The Bluetooth standard limits the maximum number of active devices to seven [147] which may be insufficient for many applications. Throughput and latency become unreliable when using multiple devices with a single Bluetooth receiver which may comprise real-time applications. Bluetooth modules also consume more power than some other wireless technologies and so reduce the potential battery life.

The x-BIMU was designed to function as a low-power, wireless IMU to provide low-latency performance for real-time applications. Use of any specific wireless technology would represent a constraint to some applications; the x-BIMU instead employs a modular design for compatibility with a wide range of communication modules. The x-BIMU design was intended to meet the following specification.

### x-BIMU design specification

- Calibrated gyroscope, accelerometer and magnetometer with on-board AHRS
- Low-latency, real-time wireless communication
- Modular design to support a range communication interface
- Low-power operation

- Synchronisation of data between multiple devices

### 8.3.1 x-BIMU overview

The x-BIMU is designed to mate with XBee-style communication modules; hence the name “x-BIMU” which would be pronounced “XBee IMU”. XBee is the brand name for a product line of wireless communication modules produced by Digi. The XBee-style connector has been adopted by many manufacturers and so represents a standardised interface. Although the x-BIMU is designed to operate with any XBee-style module, it is optimised to provide its best low-power, low-latency and robust wireless performance with XBee 802.15.4 modules. Using these modules, up to 12 x-BIMUs can stream simultaneously with a single host to provide synchronised measurement data with sample rates >200 Hz. Figure 8.4 shows the x-BIMU with an unattached XBee 802.15.4 module.



Figure 8.4: x-BIMU with an unattached XBee 802.15.4 module

Size and weight are often important for wireless applications and so the x-BIMU was designed for a compact form factor. When enclosed within its plastic housing with a 320 mA battery and wireless module, the x-BIMU has a total size of 30 × 38 × 22 mm and weight of 22 g. Figure 8.5 shows the x-BIMU in plastic housing alone (left) and combined with a Velcro body strap (right) designed to secure the plastic housing for human motion applications.



Figure 8.5: x-BIMU in plastic housing with XBee 802.15.4 module and 320 mAh battery (left) and Velcro strap designed for human motion applications (right)

The principle operation of the x-BIMU is to process measurements from the on-board gyroscope, accelerometer and magnetometer to yield calibrated quantities alongside AHRS data and to stream this data via the connected communication module to the host platform; for example, to a computer or smart phone. Calibration parameters are evaluated for each x-BIMU during production and saved to internal memory. During normal operation, the on-board sensors are sampled and AHRS algorithm updated at 256 Hz. The rate at which individual sensor measurements or orientation data are then transmitted can be configured in internal settings.

The communication protocol was designed to maximise compatibility. Data can be transmitted as either American Standard Code for Information Interchange (ASCII) or binary packets. ASCII packets represent data as human-readable strings of CSVs terminated by a new-line character. This format may be convenient for interpretation by high-level programming languages such as MATLAB. Binary packets represent data in compact packets of fixed lengths to facilitate a deterministic latency and higher throughput. Binary packets are more convenient for interpretation by lower-level languages such as C++, C# or Java. An ASCII communication protocol is also used to configure all internal settings which allows the x-BIMU to be configured using any serial terminal.

### 8.3.1.1 x-BIMU hardware

The x-BIMU board dimensions are  $25 \times 28$  mm and it weights 5 g. Figure 8.6 shows an annotated top view with key hardware components are described below.

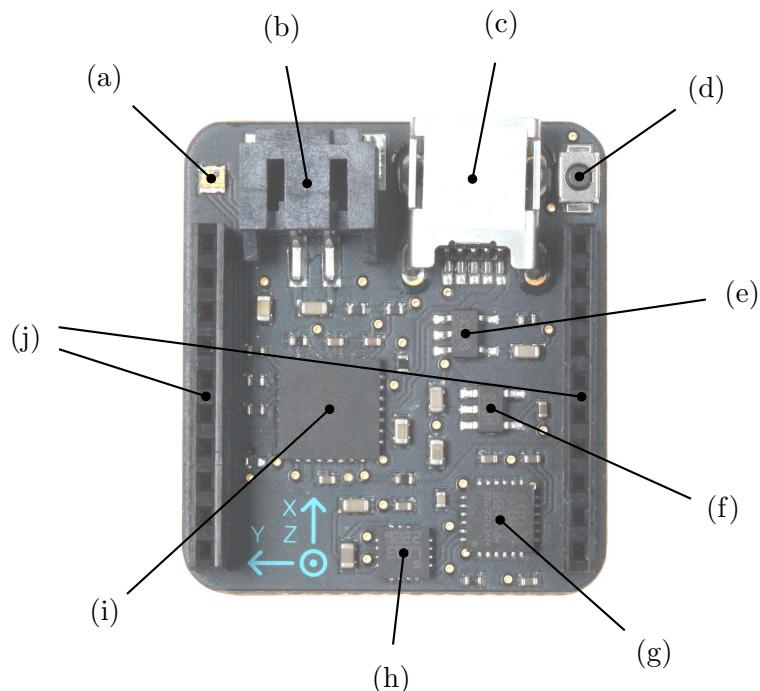


Figure 8.6: x-BIMU top with labelled key hardware components

- (a) **RGB status LED:** The colour indicates the selected wireless channel of the XBee 802.15.4 module or the battery charging status if the x-BIMU is connected to a charger.
- (b) **Battery connector:** Socket for 3.7 V Lithium-ion cell. Battery voltage is measured on-board and can be transmitted via the connected communication module to enable user to monitor the battery level.
- (c) **USB connector:** A socket for charging the connected battery, it does not provide USB communication.
- (d) **Power button:** Used to turn the device on and off. The button also provides a fail-safe mechanism of recovery if the device is configured with incorrect

settings.

- (e) **Microchip MCP73832T:** Fully integrated Lithium-ion charge management controller specifically designed for charging from a USB power supply [148].
- (f) **Micrel MIC5319:** Ultra-low dropout voltage regulator able to supply up to 500 mA [149] to accommodate high-power communication modules.
- (g) **InvenSense MPU-6050:**  $\pm 2000^\circ/\text{s}$ , 16-bit resolution gyroscope;  $\pm 16 \text{ g}$ , 16-bit resolution accelerometer; and 16-bit thermometer [121].
- (h) **Honeywell HMC5883L:**  $\pm 8.1 \text{ Gauss}$ , 12-bit resolution magnetometer [122].
- (i) **Microchip PIC24FJ64GA102:** 16-bit, 16 MIPS general purpose microcontroller with eXtreme Low Power (XLP) technology [150].
- (j) **XBee socket:** Connection for any XBee-style module. Provides interface to serial communication as well as control signals such as sleep and reset.

### 8.3.1.2 x-BIMU software

The x-BIMU Terminal is a windows application that functions as a basic serial terminal but also provides tools specific to the x-BIMU, including real-time 2D and 3D visualisations, calibration tools data logging to CSV files for exporting data to software such as MATLAB and Microsoft Excel. Figure 8.7 shows the x-BIMU Terminal with gyroscope, accelerometer, magnetometer and battery measurements each being displayed as 2D plots, and AHRS data being displayed as a 3D representation. The x-BIMU Terminal is open-source<sup>4</sup> and incorporates a generic C# interface library for the x-BIMU that can be reused by other C# projects. There are also interface libraries available for C++<sup>5</sup> and Android<sup>6</sup>.

---

<sup>4</sup><https://github.com/xioTechnologies/x-BIMU-Terminal>

<sup>5</sup><https://github.com/xioTechnologies/x-BIMU-Arduino-Example>

<sup>6</sup><https://github.com/xioTechnologies/x-BIMU-Android-Example>

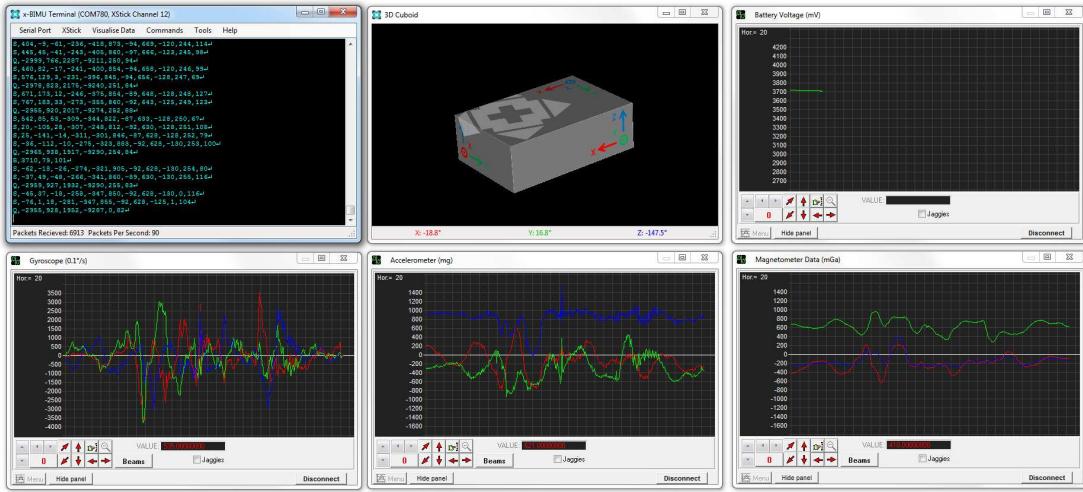


Figure 8.7: x-BIMU Terminal displaying real-time data with 2D and 3D graphics

### 8.3.2 Modular communication interface

The x-BIMU is optimised for XBee 802.15.4 modules but compatible with any XBee-style communication module. Upon start up, the x-BIMU will automatically detect if an XBee 802.15.4 module is present and configure it to operate in API mode. This gives the x-BIMU explicit control over packetisation at the 802.15.4 Medium Access Control (MAC) layer [151] to enable low-latency and low-power performance. If an XBee 802.15.4 module is not detected, communication reverts to a generic 115200 baud serial interface compatible with any XBee-style module with a matching configuration.

Digi support a number of different communication technologies within the XBee product line. Support for additional technologies is offered by other manufacturers who have adopted the XBee-style interface. Figure 8.8 shows a range of modules chosen to demonstrate the versatility of the x-BIMU when combined with different off-the-shelf modules. Each module is described below.

- (a) **Digi XBee 802.15.4 (1 mW):** Standard 802.15.4 module with a maximum transmit power for 1 mW for a range of 100 m [151]. The module shown incorporates a low-profile PCB antenna.

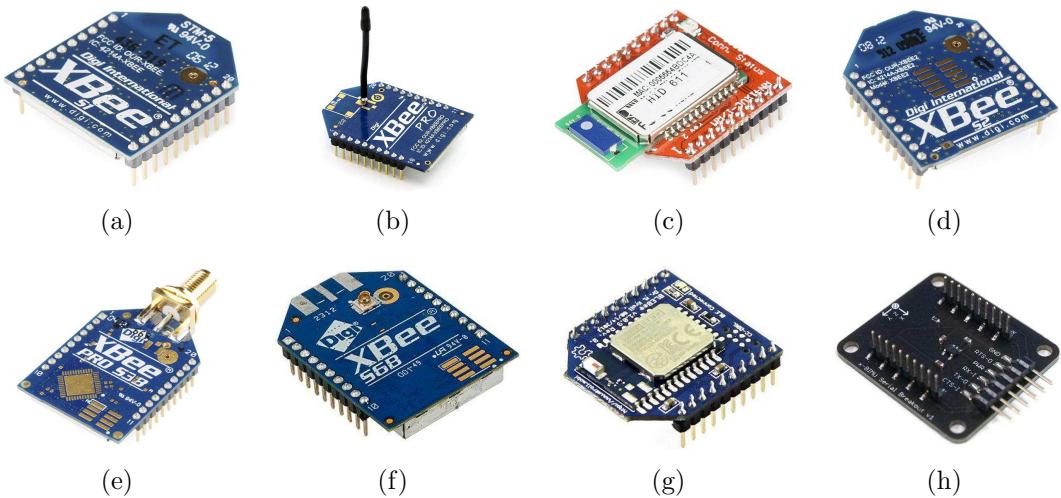


Figure 8.8: Example XBee-style communication modules compatible with the x-BIMU, for: 802.15.4 (a) (b), Bluetooth (c), ZigBee (d), Proprietary 900 MHz (e), Wi-Fi (f), Bluetooth 4.0 Low Energy (g) and wired serial (h).

- (b) **Digi XBee 802.15.4 (60 mW):** High-power variant of the 802.15.4 module; operates identically but has a maximum transmit power for 60 mW for a range of 1 km [151]. The module shown has a wire antennae to provide better omnidirectional performance.
- (c) **Microchip RN41XV Bluetooth:** Class 1 Bluetooth 2.1 module for 100 m range [152] Supports SPP to avoid the need for host drivers.
- (d) **Digi XBee ZB:** Supports licensed ZigBee for complex network configurations such as mesh networks, and compatibility with third-party ZigBee systems. The module shown is a 2 mW variant for 120 m range, 63 mW variants are also available for 3.2 km range [153].
- (e) **Digi XBee-PRO 900:** Uses proprietary 900 MHz communication protocol with range of 15.5 km [154]. The module requires an external antennae.
- (f) **Digi XBee Wi-Fi:** 802.11n for high-throughput User Datagram Protocol (UDP) or Transmission Control Protocol (TCP) transmission over ad hoc or existing infrastructure Wi-Fi networks [155].
- (g) **Seeed Studio BLEbee:** Bluetooth 4.0 Low Energy module for direct

connection to modern consumer platforms such as laptops and smart-phones.

BLEbee is an open-source<sup>7</sup> project developed by Dr Michael Kroll.

- (h) **x-BIMU Serial Breakout:** Open-source<sup>8</sup> serial adapter for direct wired interface to microelectronics or USB and RS-232 platforms when combined with a corresponding adapter.

### 8.3.3 Low-power optimisation and features

The x-BIMU incorporates several low-power optimisations and features to extend battery life and facilitate potential new applications. Central to the design is the Microchip PIC24FJ64GA102 XLP microcontroller. This part was chosen as its 16 MIPS would meet the demands of the calibration and AHRS processing requirements with a maximum current consumption of 11 mA during normal operation and 25 nA in its lowest power mode [150]. Standard hardware and firmware design techniques were employed to minimise power consumption [156]; including extensive use of interrupts allowing the processor maximise time spent in *idle* mode and sourcing peripheral component power from a microcontroller output to minimise current leakage when powered down.

#### 8.3.3.1 802.15.4 low-power optimisations

The 802.15.4 standard is specially designed for low-power devices to spend large amount of time asleep [157]. The XBee 802.15.4 idle current is 50 mA but also includes a <10 µA sleep mode with a typical wake up time of 10 ms [151]. The x-BIMU has explicit control of packetisation of the 802.15.4 MAC layer and so is able to cycle this sleep mode between packet transmission to achieve significant power savings for low data-rate applications.

---

<sup>7</sup><https://github.com/michaelkroll/BLEbee>

<sup>8</sup><https://github.com/xioTechnologies/x-BIMU-Serial-Breakout>

### 8.3.3.2 Sleep timer and motion trigger wake up

In some applications it may be desirable for the wireless IMU to power down during periods of inactivity to extend the battery life. The x-BIMU supports this through a sleep timer and motion trigger wake up. The sleep timer will trigger the x-BIMU to enter sleep mode if no motion is detected for a specified period of time. The motion trigger wake up will cause the x-IMU to immediately wake up again once motion is detected. An important feature of this mechanism is the low-latency wake up as any missed motion data could compromise measurements. To minimise power consumption, motion is detected using the MPU-6050 built-in Digital Motion Processor (DMP) [121]. This allows the all other electronics to completely power down until a hardware interrupt pulse is generated by the MPU-6050.

### 8.3.3.3 Measured current consumption

The current consumption of an x-BIMU with an XBee 802.15.4 module was measured for different usage scenarios. Table 8.1 summarises these measurements and indicates the corresponding battery life calculated for the 320 mAh battery shown previously in Figure 8.5. The nominal voltage of the battery is 3.7 V.

Test condition	Current	Battery life
Sleep mode	94 $\mu$ A	142 days
Sleep mode (motion trigger wake up enabled)	1 mA	13 days
Running (transmitting data at 16 Hz)	22 mA	15 hours
Running (transmitting data at 128 Hz)	64 mA	5 hours

Table 8.1: Measured current consumption of x-BIMU and XBee 802.15.4 module

### 8.3.4 Synchronisation of multiple devices

Many applications require measurements received from multiple IMUs to be synchronised in time. The x-BIMU is able to achieve this using the XBee 802.15.4 modules. An x-BIMU can be configured to transmit data on one of 12 channels. This allows up to 12 x-BIMUs to transmit simultaneously without interfacing with one another. The firmware was designed for a fixed and determined latency between the sampling of the sensors and transmission of the corresponding data packet. Synchronisation may therefore be achieved by time-stamping the time-of-arrival of each packet.

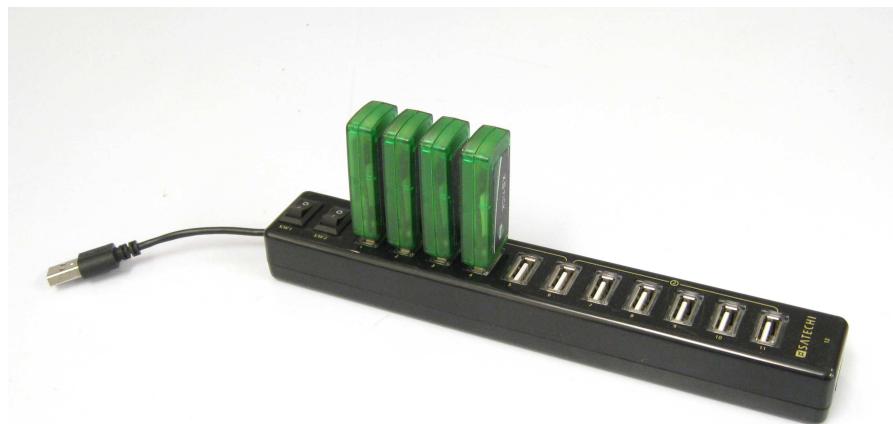


Figure 8.9: Four 802.15.4 receivers (Digi XSticks) in a USB hub provide synchronised measurements from x-BIMUs

This time-of-arrival synchronisation technique avoids the need to maintain independent synchronised clocks on each wireless node. However, it does require one receiving module per x-BIMU. This is currently achieved using a USB hub of multiple 802.15.4 receivers, as shown in Figure 8.9. An open-source<sup>9</sup> software application communicates with each receiver and time-stamps each received packet.

---

<sup>9</sup><https://github.com/xioTechnologies/x-BIMU-Logger>

## 8.4 x-OSC: High-performance wireless IMU and I/O interface

The x-OSC was developed in 2012 and intended to address the need for a combined high-performance wireless I/O interface and IMU. The x-OSC was introduced in a 2013 paper [9] which provides a survey of existing wireless I/O devices and discussion on their shortcomings. Recent years have seen the appearance of several commercial wireless IMU systems, for example the Xsens MTw [128] and APDM opal [158]. However, these systems utilise low-bandwidth wireless technologies and so are typically limited to an update rate of  $\sim$ 120 Hz for multiple IMUs. Furthermore, none of these systems offer an I/O interface for external sensors and output devices.

The x-OSC takes advantage of modern embedded Wi-Fi technology to achieve superior wireless performance and take advantage of the wide support and versatility of Wi-Fi and Ethernet networking. The x-OSC design was intended to meet the following specification.

### x-OSC design specification

- Calibrated gyroscope, accelerometer and magnetometer with on-board AHRS
- 32 configurable I/O channels
- Versatile Wi-Fi network configurations
- Maximise platform compatibility
- Optimise for throughput and latency

### 8.4.1 x-OSC overview

The x-OSC, shown in Figure 8.10, is a wireless I/O board that provides just about any software with access to 32 high-performance analogue/digital channels and on-board IMU measurements via Open Sound Control (OSC) messages over Wi-Fi. There is no user programmable firmware and no software or drivers to install making x-OSC immediately compatible with any Wi-Fi-enabled platform.



Figure 8.10: x-OSC

OSC is a lightweight network protocol designed specifically for communication between computers and multimedia devices [159]. It is an ideal solution to achieve a cross-platform and cross-language interface with over 80 languages/platform implementations [160]. Although OSC is widely supported, many platforms fail to incorporate the full specification [161]. To maximise compatibility, x-OSC messages are limited to the four fundamental data types: *int32*, *float32*, *OSC-string* and *OSC-blob*. A full description of x-OSC messages is available in the x-OSC user manual [162].

#### 8.4.1.1 x-OSC hardware

The x-OSC board dimensions are  $32 \times 45$  mm and it weights 11 g. Figure 8.11 shows an annotated top view with key hardware components are described below.

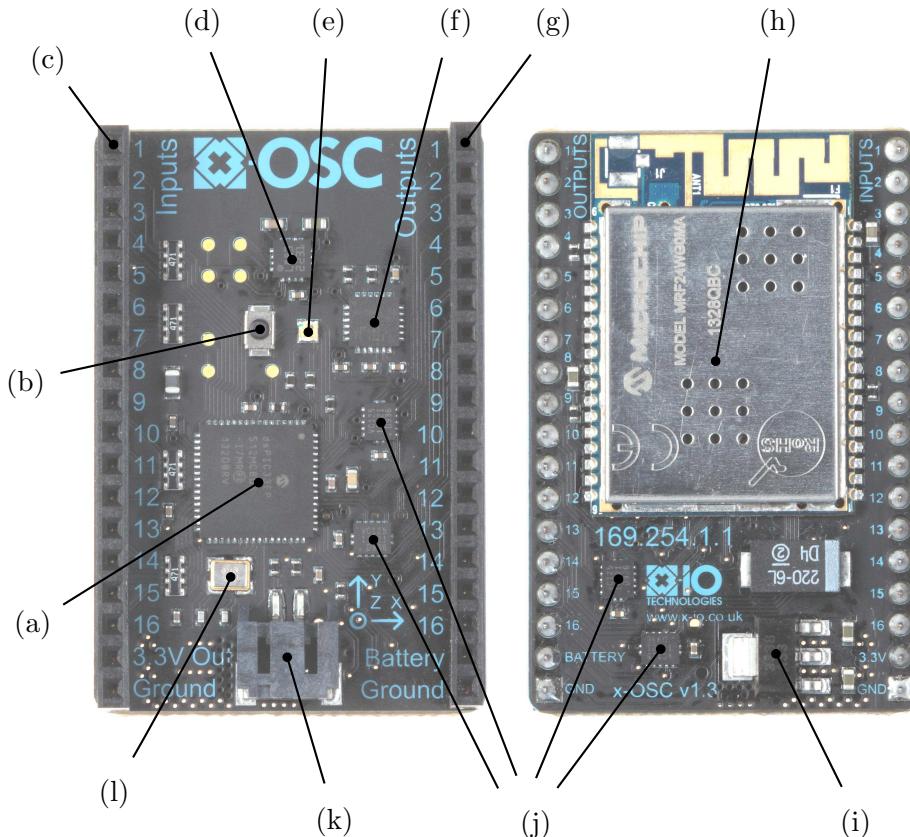


Figure 8.11: x-OSC top and bottom view with labelled key hardware components

- (a) **dsPIC33EP512MC806:** 16-bit, 70 MIPS DSC that represents one of Microchip's highest performing devices with extensive I/O and peripheral support [163].
- (b) **Ping button:** Used to trigger network 'pings' to facilitate device discover. Also provides a fail-safe mechanism of recovery if the device is configured with incorrect settings.
- (c) **16 Input channels:** See Section 8.4.2 for more information. Includes regulated 3.3 V output to power user electronics.

- (d) **Honeywell HMC5883L:**  $\pm 8.1$  Gauss, 12-bit resolution magnetometer [122].
- (e) **RGB LED:** Indicates the networking mode and connection status. Can also be controlled by user to provide visual feedback within an application.
- (f) **InvenSense MPU-6050:**  $\pm 2000^{\circ}/\text{s}$ , 16-bit resolution gyroscope;  $\pm 16$  g, 16-bit resolution accelerometer; and 16-bit thermometer [121].
- (g) **16 Input channels:** See Section 8.4.2 for more information. Includes unregulated battery voltage output to power user electronics.
- (h) **Microchip MRF24WG0MA:** Microchip's highest performing Wi-Fi module, capable of up to 5 Mbit sustained throughput and maximum transmit power of +18 dBm [164].
- (i) **Analog devices ADP3338:** 1A ultralow dropout regulator capable of meeting the power demands of both Wi-Fi and external user electronics [165]
- (j) **74LVC125A Quad buffers :** Output buffers enable each of the 16 outputs to sink/source up to 50 mA [166].
- (k) **Battery connector:** Socket for 3.7 V Lithium-ion cell. Battery voltage is measured on-board and available to the user as an OSC message.
- (l) **Crystal oscillator:** 16 MHz,  $\pm 20$  ppm crystal drives system clock [167].

#### 8.4.1.2 x-OSC browser interface

An embedded web server enables all internal settings to be configured using a web browser, see Figure 8.12. Settings may be viewed and modified during run-time without interrupting the acOSC messages. This eliminates the need for any device-specific software and allows the x-OSC to be configured using any laptop, smart phone or tablet.

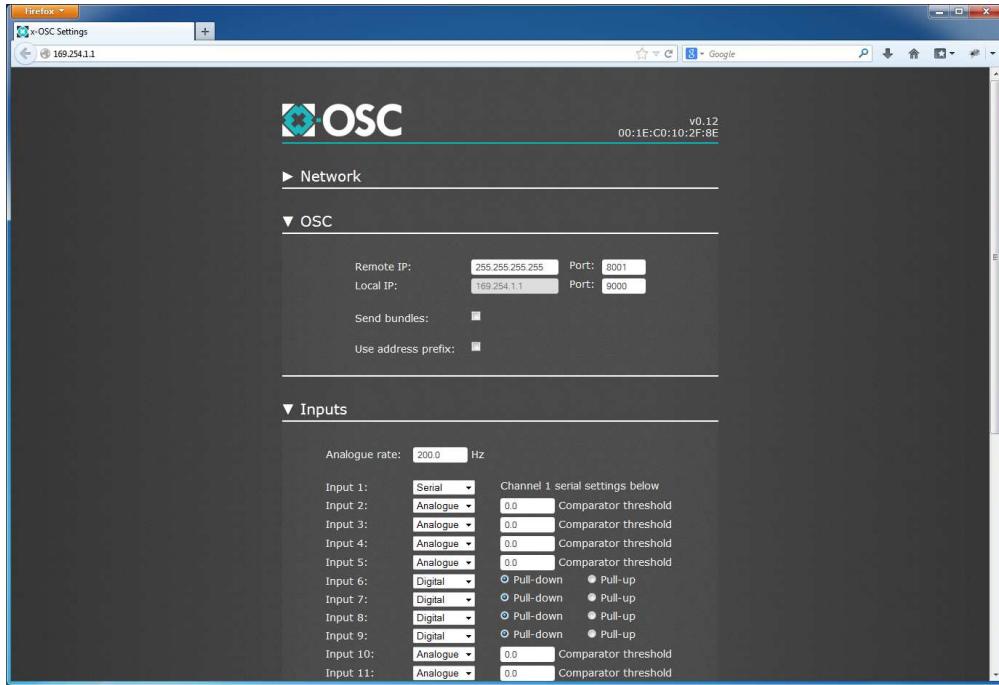


Figure 8.12: x-OSC settings viewed on web browser

### 8.4.2 I/O channels

16 dedicated inputs can be independently configured to be either analogue or digital. Digital inputs can be configured to use internal pull-up/down resistors and to minimise latency their state is only transmitted on change. All 16 analogue inputs are sampled with 13-bit resolution and sent simultaneously at a specified update rate up to 400 Hz. Analogue mode inputs also provide a comparator function to send a message each time a specified threshold is crossed. This enables low-latency threshold detection without the need for a high message rate.

16 dedicated outputs can be independently configured to digital, pulse or PWM modes. In digital mode, an output can be set to high or low enabling simple control of LEDs, relays, or generation of control logic signals. In pulse mode, an output can be triggered to generate a pulse with a period of 1 ms to 1 minute at a resolution of 1 ms. This may be useful for momentary actuators such a solenoid driving the strike mechanism of a percussive instrument. An output in PWM mode can generate a PWM waveform from 5 Hz to 250 kHz with a duty cycle

resolution up to 16-bit. PWM is commonly used as a DAC where fixed frequency and variable duty cycle approximate an analogue signal. For example, this may be used to control the brightness of a light or the speed of a motor.

In addition to modes described above, the first four inputs and outputs can be configured to serial mode with each transmit and receive pair utilising a dedicated hardware Universal Asynchronous Receiver/Transmitter (UART) module. Each serial channel supports baud rates in the range 9600 to 1 M baud and incorporates a 2 kB buffer to ensure high throughput without loss of data. Received serial data is framed before being sent as *OSC-blob* messages. Framing boundaries are determined by a user defined buffer size, timeout and optional framing character.

### 8.4.3 Network modes

x-OSC can be configured to operate in one of two network modes: ad hoc or infrastructure. In ad hoc mode, x-OSC creates a network for other devices to join. Multiple devices can connect to a single x-OSC with simultaneous access to its I/O. Infrastructure mode allows x-OSC to connect to an existing network. The device Internet Protocol (IP) address can be configured to be static or use DHCP to be assigned an appropriate IP address by the network server. The assigned IP address can be discovered by pressing the ping button, which causes x-OSC to broadcast a message indicating the IP address over the network. Alternatively, a ping message can be sent to x-OSC by another network device. Infrastructure mode enables multiple x-OSCs to operate on the same network and be addressed by multiple host devices also connected to the network. A connection to a router can also provide an inherent interface to x-OSC via Ethernet or from remote internet connections.

### 8.4.4 Practical throughput and latency

Investigations were conducted to demonstrate the practical latency and throughput that could be achieved for up to 15 x-OSCs. The results of the investigations were presented in a 2014 paper that also covered recommended networking optimisations for best performance [10]. Experiments were carried out in a university lab in the presence of other Wi-Fi networks, which were revealed using a Wi-Fi scanning application. A spectrum analyser was used to confirm that any significant use of the 2.4 GHz spectrum was limited to the visible Wi-Fi networks. When only one Wi-Fi channel was required, the scanning application was used to select an appropriate channel. Each investigation used a Late 2013 13" MacBook Pro running WireShark v1.10.5, fitted with a Broadcom BCM4360 transceiver and a Thunderbolt to Ethernet adaptor. All OSC messages were sent as unicast UDP packets. The infrastructure network was provided by a LevelOne WBR-6805 pocket router/AP configured as *open* (no encryption) in 802.11g mode (54 Mbps) with an Ethernet connection to the host computer.

#### 8.4.4.1 Round-trip Latency

The round-trip latency was evaluated by wiring an x-OSC digital input and output together and then measuring the time between an output toggle message being sent by the host computer and the input change message being received from x-OSC. The *digital input* and *digital output* OSC messages are 100 and 32 bytes long respectively, each incorporating an address pattern and either 16 or 1 *int32* argument/s. A software application was written to send a message to toggle the digital output every 50 ms and WireShark was used to log the time of each packet sent and received. This method of evaluating latency is different from that previously proposed [9] and has the advantage of eliminating the software application from the loop, which may impose an unknown latency specific to

the software, Operating System (OS) or processing load of the computer. The infrastructure network was created by connecting the computer and x-OSC to a single router/AP.

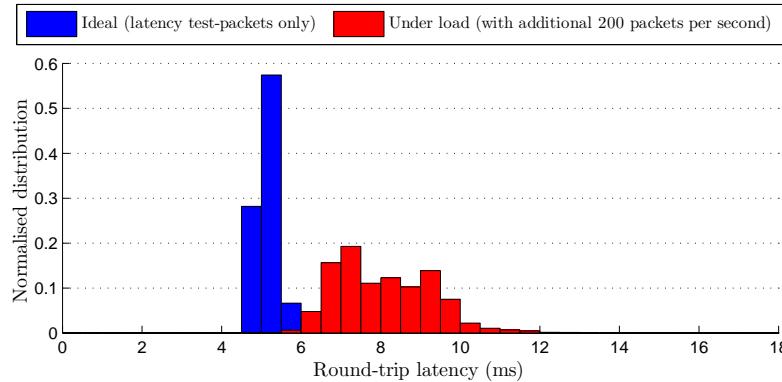


Figure 8.13: Normalised distribution of measured round-trip latency

Figure 8.13 shows the round-trip latency distribution of over 13000 samples achieved for either ‘ideal’ conditions, where communication is limited to only the *digital input/output* messages; and ‘under load’ conditions, where x-OSC was configured to simultaneously send *analogue input* data at 200 messages per second. Table 8.2 summarises the results. It may be assumed that the latency for communication in either direction is approximately half that of the observed round-trip latency, for example, <3 ms for ‘ideal’ conditions.

Test condition	Mean	95% less than
Ideal	5.30 ms	6.59 ms
Under load	8.09 ms	9.96 ms

Table 8.2: Summary of measured round-trip latency

#### 8.4.4.2 Throughput

Throughput was evaluated as the total packet rate achieved by up to 15 x-OSCs, each attempting to send 450 *analogue input* messages per second. This rate is intentionally greater than can be achieved by a single x-OSC to demonstrate saturated throughput. Each message is a UDP packet containing a 104 byte

OSC message including 16 *float32* arguments. Wireshark was used to log the time of arrival of each packet and packets per second was calculated as the number of packets arriving from each x-OSC within each one second window. Each experiment starts with a single running x-OSC. At one minute intervals, an additional x-OSC is activated for a period of 15 minutes, to yield a recording of throughput for 1 to 15 x-OSCs. Tests were conducted with the 15 x-OSCs sharing a single channel and evenly distributed between three non-overlapping channels to investigate the benefit.

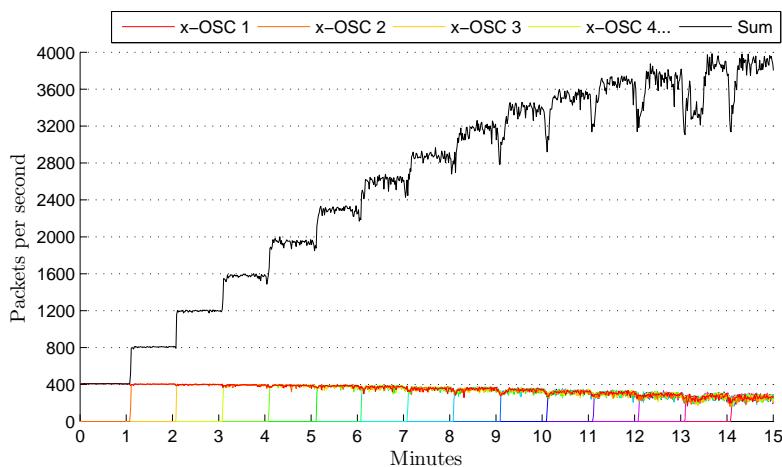


Figure 8.14: Throughput of 1 to 15 x-OSCs sending to a single AP on one channel

Figure 8.14 shows the throughput for 1 to 15 x-OSCs connected to a single AP and indicates that up to four x-OSCs can operate on a single channel without significantly impacting the 400 packets per second ceiling of a single x-OSC. Beyond this, additional x-OSCs reduce the throughput of each device so that when all 15 are active, the net throughput is ~3800 packets per second. An important observation is that this over-saturated network reduces the throughput of each device equally (from 400 to ~250 packets per second).

Figure 8.15 shows the throughput for 1 to 15 x-OSCs connected to three APs, each operating on a separate non-overlapping channel. Distribution between multiple channels can be seen to produce an increased net throughput of ~4800 packets per second. The first group of five x-OSCs were configured on channel 1, the next on

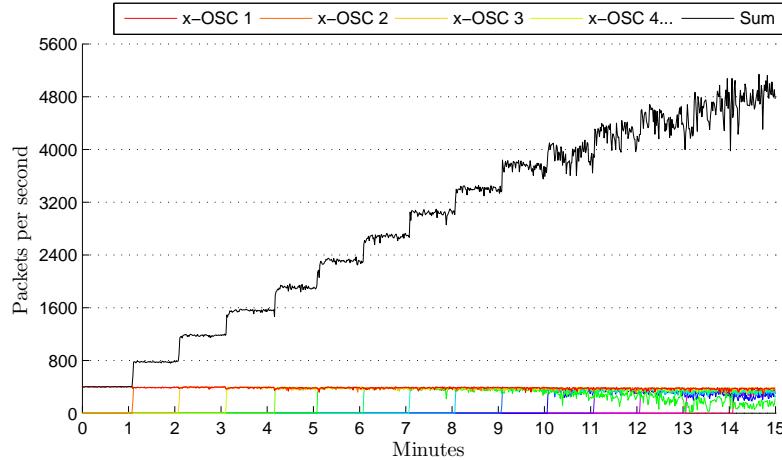


Figure 8.15: Throughput of 1 to 15 x-OSCs sending to three APs on three non-overlapping channels

channel 11 and the final five on channel 6. This specific order demonstrates that the channel 1 and 11 groups are able to operate simultaneously without interference. After 10 minutes, the inclusion of the the final group (channel 6) increases the net throughput proportionally but with significantly increased variance. Crucially, it can be seen that the the throughput of an x-OSC in the channel 11 group falls below that of the rest once the channel 6 group appears. This demonstrates the potential for channel 6 to interfere with channels 1 or 11 [168].

#### 8.4.4.3 Throughput-latency trade-off

Although the above results show that the x-OSC is able to achieve both high-throughput and low latency, the large channel access overhead involved in the transmission of 802.11 packets does produce a trade-off between throughput and latency [169]. For example, the x-OSC firmware was modified to demonstrate a maximum throughput of  $\sim 400$  packets per second with a 1 byte payload (0.4 kB/s), and  $\sim 295$  packets with a 1472 byte payload ( $\sim 434$  kB/s). This corresponds to a 1000 times increase in throughput at the cost of a  $\sim 25\%$  reduction in send rate. This indicates the potential for the x-OSC to achieve significantly higher sample rates by allowing the user to sacrifice latency in favour of bundling multiple OSC

messages within large UDP packets.

## 8.5 Collaborative applications

The development of the IMU platforms has coincided with exploration of numerous applications over the past four years. However, the largest projects were achieved through collaboration with other people. This section present two selected projects.

### 8.5.1 The gloves

“The gloves” are a pair of data gloves that allow the wearer interact with software though the natural motion and gesticulation of their hands. Although the gloves have potential for a wide variety of applications, development has focussed on live music performance where the gloves provide the wearer with an unlimited number of virtual musical instruments and access to all the tools of a professional production desk; all played and controlled live, through precise motion and gestures.

The project started in 2010 as a partnership between a musician Imogen Heap, and a computer scientist Dr Tom Mitchell of the University of the West of England. Together they created ‘SoundGrasp’ [170]; a single glove with a wrist-mounted mic that would allow the wearer to record and play back audio samples. The glove was a 5DT 14 Ultra glove which incorporated 14 fibre optic bend sensors to provide measurements finger flexion and abduction via USB. An ANN was used to identify predefined postures that would either trigger events or switch modes. The continuous motion of flexion also provided a means to control variable audio effects including reverb and filtering. Figure 8.16 shows the glove with the postures and their associated control audio function.



Figure 8.16: SoundGrasp gloves with wrist-mounted microphone and postures with associated control function. Image source: [170]

The fingertip-less gloves did not obstruct the wearer from playing most musical instruments and the wrist-mounted microphone could be used to sample both voice and acoustic instruments. SoundGrasp was an exciting proof of concept but it was clear that much more could be achieved with the adoption of IMU technology. The author of this dissertation joined the partnership to help achieve this.

### 8.5.1.1 Prototype gloves

The prototype gloves built upon the SoundGrasp system by using a pair of gloves and mounting an x-IMU to each wrist. Each x-IMU used a USB connection along side that of the 5DT gloves. The x-IMU auxiliary ports were configured in PWM mode to drive an Red, Green, Blue (RGB) LED on each hand. A dedicated headset microphone was also added. Figure 8.17 shows the prototype gloves system diagram (left) and being worn by musician (right).

The addition of the x-IMUs enabled numerous more dimensions of measurement on top of the gesture recognition and flexion offered by the 5DT glove alone; including: orientation, positional displacement and inertial peak detection. Furthermore, these separate sources control mechanisms could be combined to begin to resemble

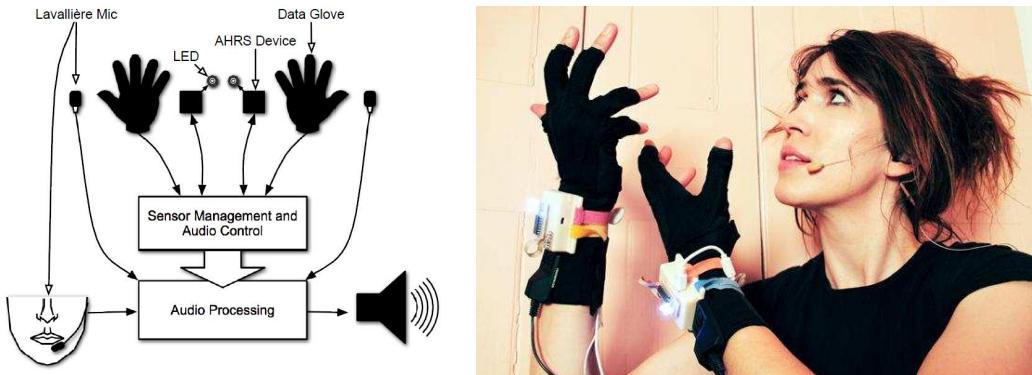


Figure 8.17: Prototype gloves system diagram (left) and being worn by musician (right)

‘real-world’ control interfaces. For example, by combining a segmented orientation measurement with velocity peak detection, the wearer could be surrounded by an invisible drum kit. A toolbox of gestural control mechanisms based on these prototype gloves was presented in a 2012 paper [11].

### 8.5.1.2 The performance gloves

The performance gloves represented a significant advance to the system. The 5DT gloves were abandoned in order to achieve a wireless system with custom designed components to meet technical requirements. The complete performance gloves system is shown in Figure 8.18 with labelled key components.

The system was comprised of the following components.

- **Gloves:** Custom made gloves incorporating an x-IMU, eight flex sensors, an RGB LED, vibration motor for haptic feedback and microphone. The x-IMU acquired the analogue flex sensor measurements via its auxiliary port. The RGB LED and vibration motor were driven by the auxiliary port of an x-IMU mounted within the arm bands.
- **Arm bands:** Contained an x-IMU with a wired connection to the RGB LED and vibration motor on the hand.

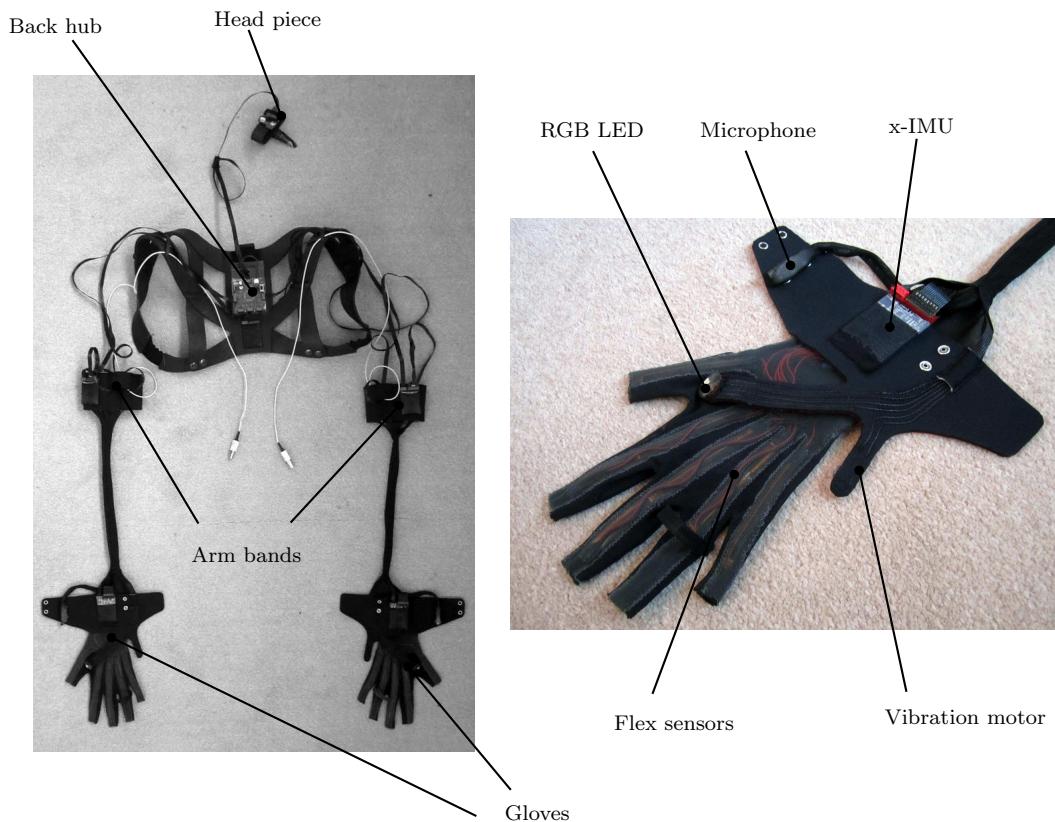


Figure 8.18: Performance glove parts

- **Harness and back hub:** Mounts wireless radio units for the two wrist-mounted microphones, headset microphone and audio earpiece. Also mounts a x-IMU and custom designed hub that routes wired serial communication between the five x-IMUs and the Bluetooth radio within the head piece.
- **Head piece:** Mounts a Bluetooth radio that provides a single wireless link between the five x-IMUs and the host software.

The performance gloves software utilised the same control mechanisms as outlined for the prototype gloves but was extended to provide a direct interface to existing professional music production software. The use of five x-IMUs distributed between the arms and torsos provided an opportunity to implement more detailed measurement of the body's motion. A key use was to use the torso orientation

measurement as a 'zero reference' for each glove so that orientation controls would be relative to the wearer and not the Earth.

The performance gloves have provided a unique tool for the composition and performance of music. They have also have potential for a use in a range of applications not yet fully explored. For example, Figure 8.19 shows the gloves being used to create and manipulate visuals projected on to a screen.

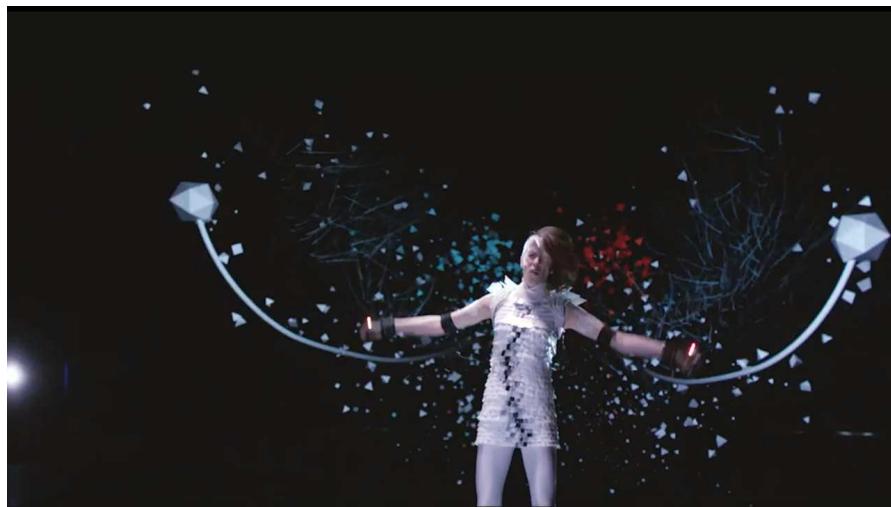


Figure 8.19: Gloves being used to create and manipulate visuals projected on to a screen

### 8.5.2 Gait analysis

Human motion tracking for biomedical monitoring applications has received significant attention in the engineering and medical research communities in recent years; see [41, 171, 172] for recent surveys. Zhoua [41], cited real time operation, wireless properties, correctness of data, and portability as major deficiencies that must be addressed to realize a clinically viable system. Body-worn sensors such as goniometers [173], pressure tubes [174], gyroscopes and accelerometers [175, 176, 177, 178, 179] have also been implemented in a range of activities and clinical conditions [179, 180, 181, 177]. There are strong reasons to consider inertial-based sensors in biomechanical analysis. They are small, light-weight,

low-powered, inexpensive, and unobtrusive and so can be arranged so that they do not interfere with the person's movements. They can be considered as viable for use outside a specialist laboratory and could be integrated into existing orthotics and clothing.

This project explored the potential application of the x-IMU to human gait studies. The autonomous data logging and motion triggered sleep/wake functionality make the x-IMU an idea tool ambulatory monitoring with the potential to address concerns outlined by Zhoua.

An algorithm was developed to obtain the translational position data from an x-IMU mounted on the foot using methods similar to [182, 183, 184, 185, 186]. Figure 8.20 shows data obtained using a prototype unit for tracking of the right foot of a test subject as they walked in a straight line.

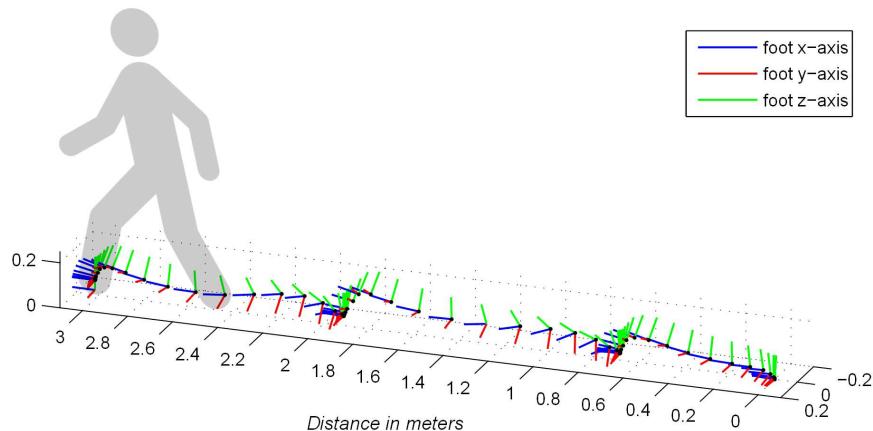


Figure 8.20: Measured foot position during walking (3 steps) obtained using an x-IMU

This provides numerous sources of measurement in addition to the IMU data, including: linear accelerations, velocities and position of the foot relative to the Earth, and a quantification of the stride characteristic such as cadence and stride length. Furthermore, these individual source of measurement can be combined to infer new information. For example, to correlate variations in patients gait when walking on different gradients or at different times of the days.

The system was used in collaboration with a team of students at Bristol University to evaluate the effectiveness of an active ankle foot orthosis with passive compliance. The x-IMU was used to measure the foot kinematics of a subject with and without the orthosis and so provided an objective qualification of how much the orthosis would impact natural gait. The project was presented in a 2011 paper [12]

## 8.6 User applications

The x-IMU has been on public sale since 2010 and the x-BIMU and x-OSC since 2013. More than 500 of these platforms have been sold to customers world-wide; including many commercial and academic research institutions. This section presents 15 selected example user applications that have utilised the platforms developed. The research projects chosen to appear in this section were selected to demonstrate the diversity of user applications and impact of the platforms in peer-reviewed publications.

### 8.6.1 Estimating upper limb kinematics

Multi IMUs mounted on an upper limb can be used to determine the limb joint kinematics and provide a valuable tool for rehabilitation of stroke patients. Daniel Galinski and Bruno Dehez used four x-IMUs to explore this in work conducted at Center for Research in Mechatronics, Université catholique de Louvain, Belgium. In a 2012 paper [187] they present an evaluation of initialisation procedures for this application of IMUs. The evaluation was achieved through a comparison of the measurement provided by four x-IMU fixed to an anthropomorphic upper limb and its built in absolute encoders. Figure 8.21 shows the corresponding mechanical model of upper limb, placement of x-IMUs and instrumented anthropomorphic

upper limb. Their investigation validated the tested initialisation procedures and suggested a potential advantage in one procedure for physically weak patients.

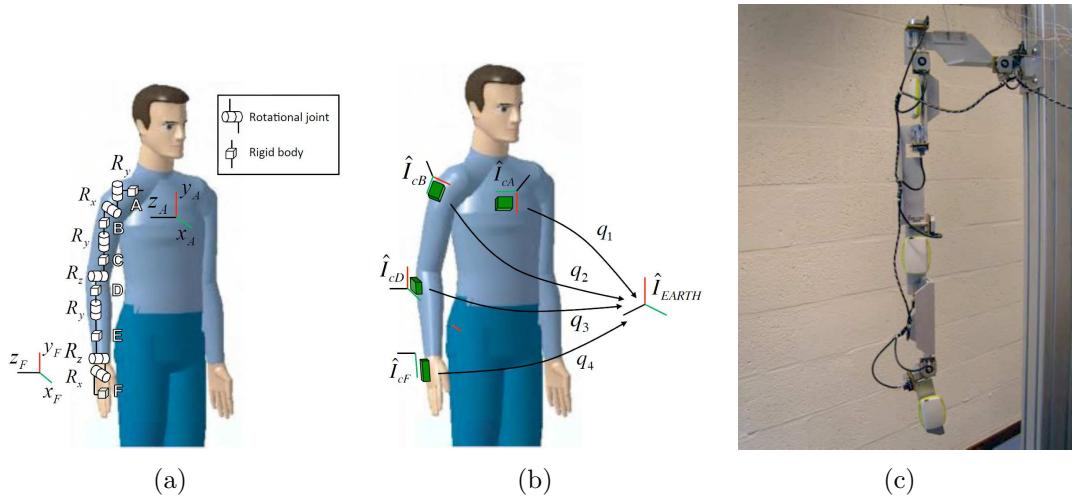


Figure 8.21: Mechanical model of upper limb (a), placement of x-IMUs (b) and instrumented anthropomorphic upper limb used for evaluation (c). Image source: [187]

### 8.6.2 Digits: Wrist-worn gloveless sensor

Digits is a wrist-worn sensor that recovers the full 3D pose of the user's hand without requiring any external sensing infrastructure or covering the hand itself. It is a Microsoft Research project in collaboration with the Culture Lab, Newcastle University; and FORTH, University of Crete. The hardware, shown in Figure is comprised on an x-IMU, an IR laser line projector, diffused IR LEDs, an and IR camera. The x-IMU was connected via USB and used to measure the wrist orientation while the IR LEDs line projector and LEDs project on to the hand to be seen by the camera. A wide range of signal processing techniques are used with a kinematic model of the hand to determine a full 3D posture from the data. The techniques and an evaluation of performance are discussed in detail in a 2012 paper [188] published by the team. A range of applications have been demonstrated on the Microsoft Research website<sup>10</sup> with digits providing an eyes-free interaction for

<sup>10</sup><http://research.microsoft.com/en-us/projects/digits/>

everyday devices such as a computer or phone.

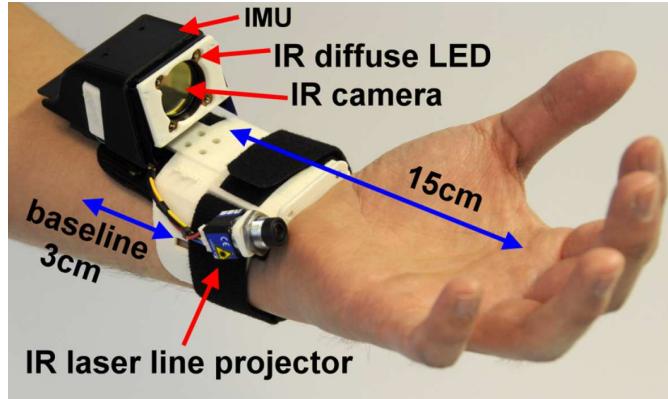


Figure 8.22: Digits wrist-worn hardware components. Image source: [188]

### 8.6.3 MONSUN II: A small and inexpensive AUV for underwater swarms

MONSUN II is a small Autonomous Underwater Vehicle (AUV) developed as a platform for research into under water swarm. It is an ongoing project at the Institute of Computer Engineering in the University of Lübeck, Germany. The robot, shown in Figure 8.23, is 60 cm long with a hull diameter of 10 cm and weighs <3 kg. Figure 8.24 summarises the key internal electronics of MONSUN II. The x-IMU is used to provide real-time orientation data to the microcontroller via a serial connection. The microcontroller uses this data within a closed-loop control algorithm to stabilise the craft [189].



Figure 8.23: MONSUN II AUV is 60 cm long, has a hull diameter of 10 cm and weighs <3kg. Image source: [190]

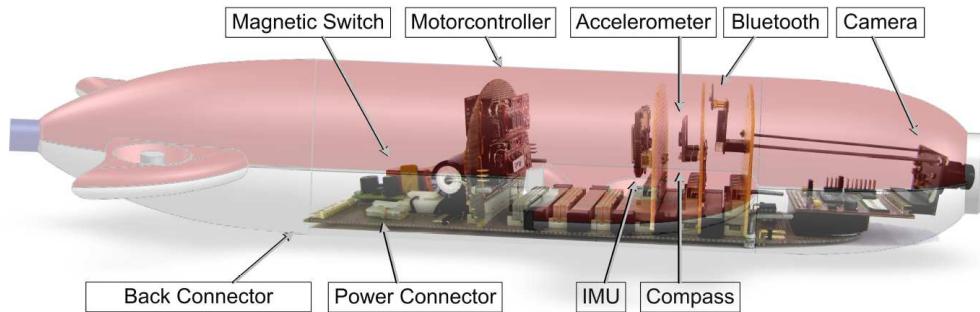


Figure 8.24: Key internal hardware complements including an x-IMU. Image source: [190]

The MONSUN project started in 2009 and has been a source of numerous publications covering the hardware and software design process [191] including novel components such as an acoustic modem for underwater communication [192]; simulation studies exploring practical applications such as environmental monitoring [190]; and significant towards fault-tolerant and energy-efficient swarms of underwater robots [193]. The most recent publication presented a preview of the forthcoming third generation MONSUN in 2013 [194] which may incorporate the x-BIMU.

#### 8.6.4 SMART-E: An omnidirectional AUV

SMART-E is an omnidirectional AUV at the Institute of Computer Engineering, University of Lübeck, Germany. The unique design of SMART-E enable high manoeuvrability and agility in 6 DOF to meet the challenges of surveying water bodies and underwater installations. The hardware design is detailed in a forthcoming paper [195]. The platform, shown in Figure 8.25, utilises a minimal configuration of three actuated thrusters arranged at 120° to each other, an actuated camera on the underside for use by a remote viewer, and a band of RGB LEDs to give clear visual feedback under water. Figure 8.26 describes the

on-board electronics. The x-IMU is used along side the Tritech Micron Scanning Sonar for localisation and navigation of the craft. SMART-E won the innovation award at the European SAUC-E competition 2012.



Figure 8.25: SMART-E AUV uses three actuated thrusters for omnidirectional movement and a band of RGB LEDs to provide visual feedback under water. Image source: [195]

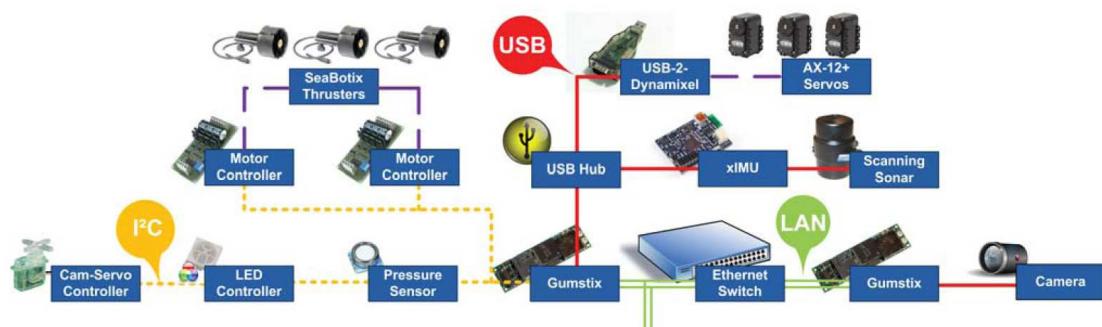


Figure 8.26: Internal electronics and communication networks between sensors, actuators and controllers. Image source: [195]

### 8.6.5 Tidal turbulence spectra from a compliant mooring

Detailed analysis of tidal turbulence is essential for the development of efficient tidal energy sites. Tidal Turbulence Moorings (TTM) incorporating Acoustic Doppler Velocimeters (ADVs) to provide an effective tool for acquiring tidal data but are subject to motion contamination in strong currents. In a recent paper [196], a team from the Northwest National Marine Renewable Energy Center at the University of Washington present a passively stabilised solution that utilises x-IMUs mounted on the ADVs to achieve motion correction in post-processing.

The team tested two versions of the TTM, each comprised of a steel float tethered to an anchor with several instruments mounted along the tether as shown in Figure 8.27. The mounted instruments include: ADVs, current meters, x-IMUs, and an Acoustic Wave And Current Profiler. The x-IMU logged data to its on-board SD card to provide orientation data necessary for motion correction. The TTMs were deployed at different locations through 2012 and 2013. Results showed although raw data saw sufficient to evaluate turbulence intensities and reasonable turbulence spectra, Motion correction would more detailed aspects of the turbulence.

### **8.6.6 Virtual reality based minimally invasive surgery training system**

Minimally Invasive Surgery (MIS) allow invasive surgery to be performed with only a small surgical wound. This presents many advantages to the patient including, less pain, reduced risk of haemorrhaging, and a faster recovery. However, manipulation of the surgical tools demands excellent hand-eye coordination of the surgeon which must be gained through training. A virtual reality based MIS training system was presented in a recent paper [197], developed through a collaboration between the Department of Electronic and Computer Engineering at the National Taiwan University of Science and Technology, as the Departments of Surgery at the Tri-Service General Hospital and National Defense Medical Center, Taiwan. The paper presents a low-cost MIS simulation training system composed of a surgical training box, surgical instrument with x-IMU, a stereo vision camera, a DE2-115 FPGA board and a Virtual Reality (VR) game. The x-IMU is mounted on the surgical instrument as shown in Figure 8.28 and provides orientation data along side location data provided by the camera.

A virtual reality 3D game developed in C# was used to demonstrate the system. In this game, the player must use the instrumented surgical tool to pick up balls

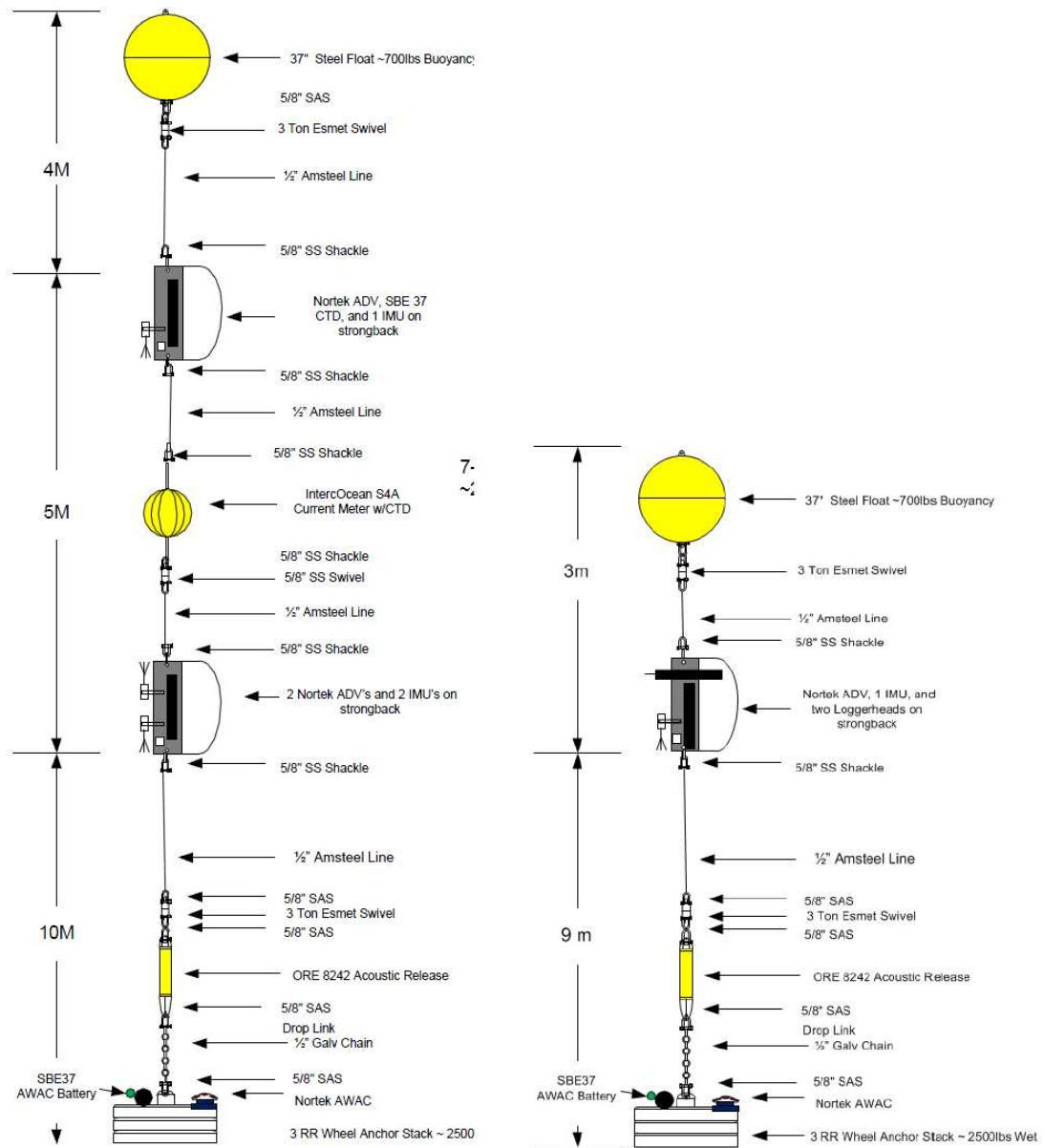


Figure 8.27: Dimensional drawings of Tidal Turbulence Moorings deployed in Admiralty Inlet (left) and Chacao Channel (right). Image source: [196]

and place them on small cylinders within a virtual environment. Figure 8.29 shows a screen shot of this software. The authors suggest that future development could improve the system through the creation of virtual human organs.

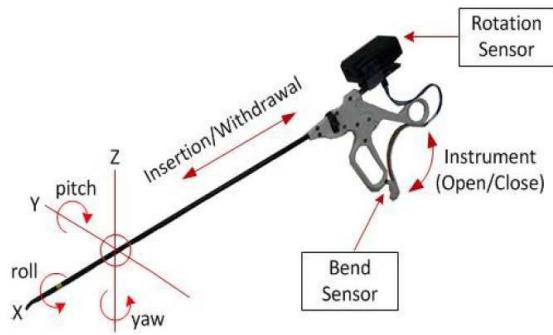


Figure 8.28: Instrumented surgical tool uses x-IMU to provide orientation data.  
Image source: [196]

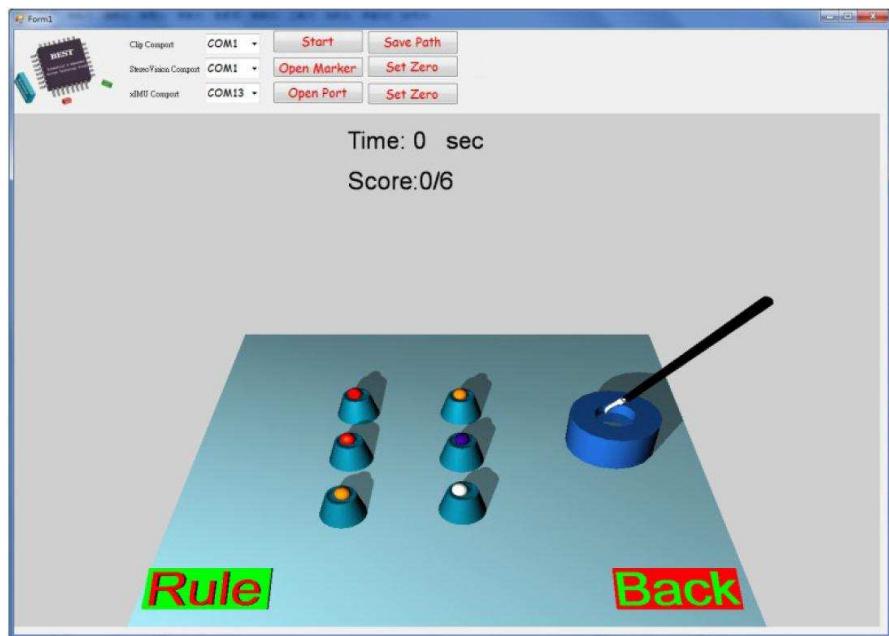


Figure 8.29: Virtual reality game controlled using the instrumented surgical tool.  
Image source: [196]

### 8.6.7 Instrumented soccer equipment to collected ankle joint kinematics in the field

In 2013, Jonathan Akins submitted a Ph.D Thesis [198] at the University of Pittsburgh documenting the development of instrumented soccer equipment to collect ankle joint kinematics in the field. The equipment was intended to establish the reliability and validity of a kinematic assessment during athletic manoeuvres and identify laboratory manoeuvres that elicited game-like demands from athletes.

The instrumented equipment included a soccer boot with an x-IMU embedded in the sole and a shin guard with the x-IMU mounted internally. Data from the two x-IMUs was synchronised by measuring a synchronisation pulse on the auxiliary ports of both x-IMUs. Figure 8.30 shows the instrumented soccer booth and shin guard connected to the synchronisation trigger. Akins concluded that instrumented equipment collected reliable and valid ankle joint kinematics and represents a viable technology for in-game data collection and injury prevention.



Figure 8.30: Instrumented equipment comprised an a soccer boot and shin guard; each with an x-IMU mounted internally. The equipment is shown here attached the the synchronisation trigger prior to a data acquisition session. Image source: [198]

### **8.6.8 Evaluating horse lameness using standalone ‘low-cost’ IMUs**

Standalone ‘low-cost’ IMUs, such as the x-IMU with its on-board data logging to an SD card, could facilitate large-scale studies into the orthopaedic deficits (lameness) of horses. In a 2013 paper [199] submitted to the journal of Computer Methods in Biomechanics and Biomedical Engineering , Brighton *et al.* prevent their investigation to the accuracy and limits of agreement of the x-IMU compared with an established IMU-based gait analysis system, the Xsens MTx [48]. The

study involved six horses, both IMUs were mounted on two anatomical landmarks, the sacrum and sternum. The vertical velocity and position of each location were estimated by combining the orientation and accelerometer measurements and the symmetry of the horse gait evaluated. The x-IMU was found to provide sufficient precision when assessing symmetry but was not able to provide reliable data for an alternative analytical method that as tested. The authors suggest that further investigations into specific calibration and processing algorithms could address these shortcomings.

### **8.6.9 Telepresences through “Flobi”, an anthropomorphic robot head**

“Flobi” is the anthropomorphic robot head developed at the Bielefeld University and serves as a long-term research platform to explore the emotional interaction of humans with robots. Flobi, shown in Figure 8.31, was designed to appear cartoon-like and babyfaced to avoid the ‘uncanny valley’ effect [200]. Emotions are expressed through 18 actuators controlling the neck, lips, eyes, eye lids and eye brows; and two LEDs for blushing cheeks. Studies including 259 participants demonstrated that Flobi is able to effectively communicate a wide range of emotions [201].

In a recent publication [202], the research team presented a system that allows a human operator to control Flobi by wearing a combined motion capture and display helmet. The system allows the direct and live transfer of human facial expressions, gaze and head movements while providing the operator with a real-time display of the scene as perceived by the robot’s vision sensors. The operators helmet, shown in Figure 8.31, incorporates a video projector, face camera, x-IMU and earphone. The x-IMU is used to provide real-time measurement of the head orientation to control Flobi’s neck actuators. The system

also records human motion datasets for analysis or playback. Other applications of Flobi have included its role as a fitness instructor [203] and an investigation into the effects of loneliness on psychological anthropomorphism [204].



Figure 8.31: Combined motion capture and display helmet (left) and robot head (right). Image source: [202]

### 8.6.10 Haptic communication between dance performers

Tychonas Michailidis, a researcher based at the Birmingham Conservatoire, has been using the x-IMU and x-BIMU in collaborations with musical and dance performers since 2011. In a recent paper [205], Michailidis et al demonstrate an attempt to create a “corporeal” link between physical performers and to give them the ability to become aware of each other’s actions in the space through haptic feedback. The project utilised x-IMUs as wearable devices to provide real-time measurements of the performers motion and gestures; as well as control a vibration motor via a PWM signal generated by the x-IMU’s auxiliary port. Each x-IMU communicated with a computer wirelessly using Bluetooth and interfaced to software via an open-source<sup>11</sup> x-IMU-OSC bridge, developed by one of the authors. The investigation explores three scenarios: ‘co-exist’, where performers would experience the haptic feedback of virtual objects as they moved around the space; ‘collaboration/play’, where the motion of one performer could result in feedback

---

<sup>11</sup><http://birminghamconservatoire.github.io/IMU2OSC/>

for another; and ‘intimacy’, where physical contact between performers is felt by others and also represented through visualisations to an audience. The authors conclude that the paper aims to invite further discussion on if such technology can enhance the experience for both performers and audience.

### 8.6.11 SLAM for an autonomous car

A team at the Department of Computer Science at the Hamburg University adopted the x-BIMU for use in their latest generation robot platform for entry in the annual Carolo-Cup competition. The competition tests the ability of a robot platform to navigate an urban road environment while identifying obstacles, crossings and missing road markings. The latest platform, shown in Figure 8.32, is build around a 45 cm electrics car with several on-board processors and sensors; including, IR proximity sensors, wheel encoders, an x-BIMU and a video camera.

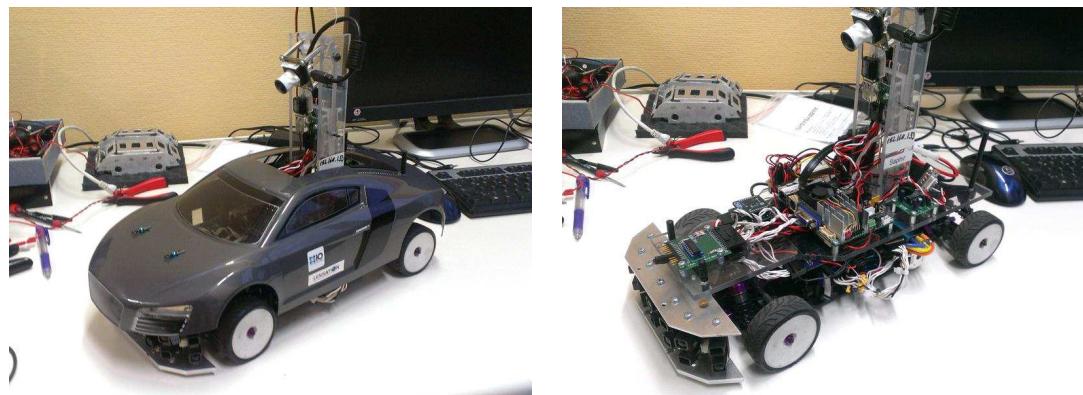


Figure 8.32: Latest generation autonomous car shown fully assembled (left) and with shell removed (right)

The x-BIMU is used with a serial breakout board to connect to the main processor via serial. The angular rate data provided by the x-BIMU is used along side images provided by the camera to estimate the location of the vehicle on a map using a RatSLAM algorithm. Simultaneous Localisation And Mapping (SLAM) encompasses all the processes required for autonomous navigation of a robot in a

new environment. RatSLAM is a SLAM algorithm that attempts to recreate the neurological mechanisms observed in rats navigating new environments. Figure 8.32 is a screen shot that shows the image provided by the platforms camera along side the map of the road being built up by the RatSLAM algorithm. The new platform competed at the 2012 and 2013 competition with limited success but the team are hopeful for the 2014 competition having now refined software components and increasing the car speed.



Figure 8.33: Screenshot showing view from camera mounted on car and the plotted location of the car as estimated by the RatSLAM algorithm

### 8.6.12 PERSIA: An autonomous humanoid robot to compete in RoboCup

RoboCup is an international humanoid robot football competition. The development of competing robots presents challenges in many disciplines including mechanical design, bipedal gait and balance, robot autonomous navigation, cooperative robotics, computer vision and artificial intelligence. The PERSIA team from the Islamic Azad University in Iran formed in 2007 and have ranked first place in national competitions on several occasions. In 2012 the team adopted the x-IMU for use on all their competing robots. The hardware and software design of

the PERSIA platform is described in a paper currently under review for the Elsevier journal of Mechatronics [206]. The robot, shown in Figure 8.34, is actuated by 20 servo motors and senses the environment through a camera and the x-IMU. The main processor implements an ANN to maintain balance and gait control. An x-IMU installed on the hip is used to provide feedback to facilitate the training of the ANN. Other software mechanisms for path planning and decision making are also described in the paper. In the recent RobotCup IranOpen 2013 competition, the team were awarded first place for the ‘adult size’ robot (150 cm) category.



Figure 8.34: ‘Adult size’ robot (150 cm) robot walking towards football (Photograph courtesy of Hamidreza Kasaei)

### 8.6.13 Low-cost structural usage monitoring of small-fleet military helicopters

Monitoring the in-service usage of military aircraft and comparing this usage with design assumptions is essential for assuring structural integrity throughout the life of a fleet. Traditional approach to these validation programmes can be expensive and so there is a demand for alternative low-cost solutions. A team at the Defence Science and Technology Laboratory, a part of the UK Ministry of Defence, has been evaluating the potential of the x-IMU to provide a cost-effective solution for monitoring the usage of historic and small-fleet rotary-wing aircraft. The x-IMU is used as a self-contained data logger to record gyroscope and accelerometer data to the on-aboard SD card. The sleep timer and motion trigger wake up allow the x-IMU to remain in situ for extended periods of time for minimal maintenance. The x-IMU has been used to capture flight data from the Army Historic Aircraft Flight Scout AH Mk 1 aircraft shown in Figure 8.35. Currently, over 20 flying hours of data have been captured and is being analysed and compared with the design usage assumptions for the aircraft.



Figure 8.35: The x-IMU is being used to collect data from the Army Historic Aircraft Flight Scout AH Mk 1 to monitor structural usage

### 8.6.14 Danceroom spectroscopy

Danceroom spectroscopy is an interactive installation and dance performance project that allows people to become part of real-time quantum partial simulations projected a large visuals and sonified through generative audio software. The project is collaboration between artists and scientists lead by theoretical chemist David Glowacki. The project is described as the chapter of the book *Molecular Aesthetics* [207]. The system uses multiple Microsoft Kinect to generate a real-time 3D model of people within a space and translate them to energy fields within a molecular simulation incorporating hundreds of particles. The simulation implements both quantum and classical mechanics but allows manipulation of the laws to design the aesthetics of the visualisations and audio generated by the particle interactions.

A recent development is the project is the use of an x-BIMU worn on a dancer wrist. The x-BIMU uses the 802.15.4 wireless modules to provide the dancer low-latency control of selected simulation parameters through their movements. Figure 8.36 shows a dancer performing with the x-BIMU mounted on her wrist as she manipulates the visualisations behind her. The project has won multiple awards and demonstrates the potential of novel approaches for accelerated particle simulations on GPU architectures [208].

### 8.6.15 Changibles: Shape changing constructive assembly

Changibles are interactive, shape-changing assembly blocks that connect together to create animated structures. The system was developed by a team at the University of Bristol and presented in a forthcoming paper [209]. A key competent of the project was the development of the algorithms that determine the construction of a shape. Figure 8.37 shows how the analysis software and

---

<sup>11</sup><http://paulblakemore.co.uk/>



Figure 8.36: An x-BIMU mounted on a dancer's wrist allows her to control the real-time visualisations of the molecular simulation. Photographer: Paul Blakemore

resultant physical construction.

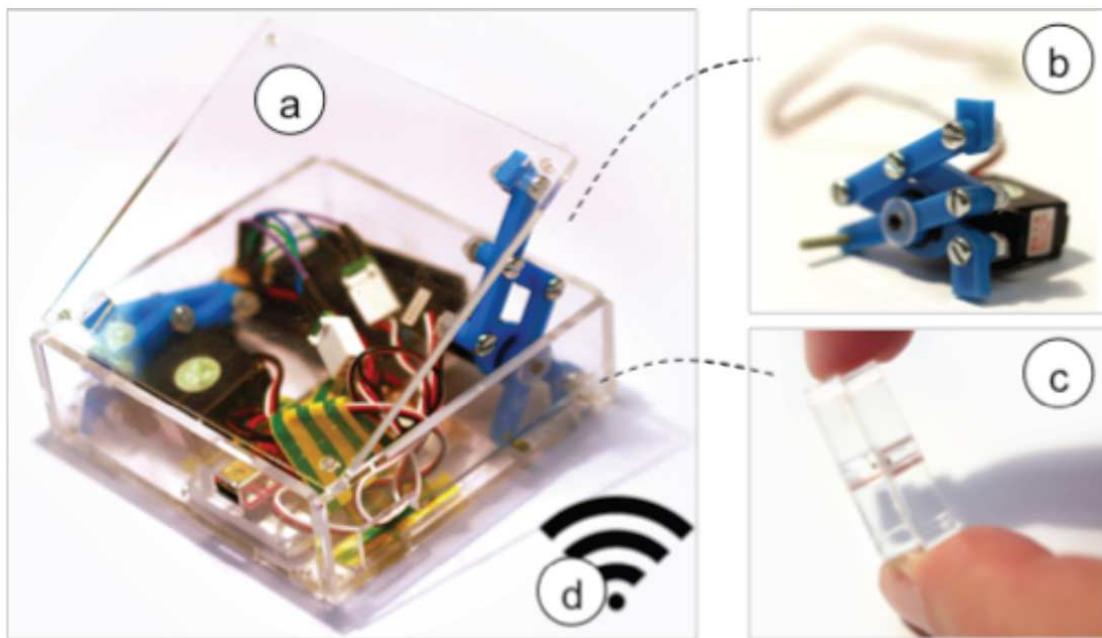


Figure 8.37: Changibles analysis software and resultant physical construction. Image source: [209]

Prototype hardware was developed using x-OSC, which provided the team with a versatile wireless interface board with the potential to explore a wide range of

sensor and actuator concepts. The current block incorporate two servo motors to actuate the one face of the block about two degrees of freedom. Each block connects to a router to achieve a single Wi-Fi connection to the computer running the software... x-OSC/Wi-Fi offers the potential for large number of nodes. The prototype block design, shown in Figure 8.38 incorporates two servo motors to acute the one face of the block about two degrees of freedom. The paper demonstrates these six blocks being assembled together to form pulsing heart.

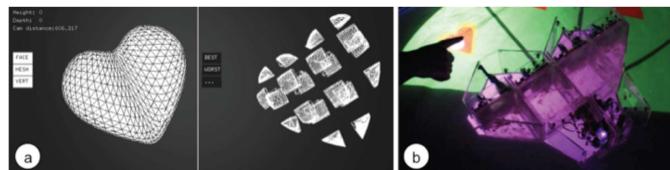


Figure 8.38: An individual block containing x-OSC for Wi-Fi control and 2 servos with scissor lift mechanism. Image source: [209]

## 8.7 Future work

Future work will see the IMU platforms continue to evolve based on user feedback and to take advantage of increasingly high-performance MEMS technology. Specific tasks have already be outlined for 2014. The modular communication of the x-BIMU as proved to be a versatile design paradigm but the device is limited by the processor; A new hardware revision will utilise the latest 32-bit processors to achieve equivalent low-power performance with superior same rates and signal processing capabilities. Planned x-OSC firmware developments will take full advantage of Wi-Fi bandwidth to enable sample rates in the order kHz with the potential to facilitate new applications.

The gloves project presented in Section 8.5.1 is an ongoing collaboration and will see the development of next generation hardware and software in 2014 using the x-OSC.



# Chapter 9

## Conclusions

This project set out to demonstrate how new applications can be realised using modern low-cost MEMS gyroscopes, accelerometers and magnetometers. This was achieved through the development of publicly available software and hardware platforms that have facilitated a wide range of commercial and academic research projects exploring a diverse range of applications. In achieving this, the project has met the following objectives:

- Development of a AHRS algorithm applicable to modern low-cost MEMS devices
- Calibration of gyroscopes, accelerometers and magnetometers for use in AHRS applications
- Design and manufacture of IMU platforms that combine the calibration and AHRS algorithm solutions
- Demonstration of a broad range of applications using the IMU platforms through collaborations

In achieving its objectives, this project has made several contributions. Chapters 3, 4 and 7 present sensor fusion algorithms applicable to conventional IMUs

and non-gyro IMUs. The conventional AHRS algorithms address the specific requirements of modern low-cost MEMS sensors and have been proven in the IMU platforms and applications presented in Chapter 8. The impact of these algorithms is also demonstrated by the numerous citations and code downloads. The non-gyro IMU was not demonstrated in application but represents an interesting alternative. The novel work presented in Chapter 4 demonstrated the potential practical benefits of such kinematically redundant sensor arrays.

Another contribution of this project is the development of a comprehensive calibration solution for gyroscopes, accelerometers and magnetometers. The calibration solutions build on the characterisation studies to address the specific needs of modern low-cost MEMS sensors. These characterisation studies revealed significant characteristics of the sensors that are not documented by the manufacturer.

The AHRS algorithm and sensor calibration works were brought together in the development of the Chapter IMU platforms presented in Chapter 8. Each of these platforms each addressed a specific design specification and together facilitated a wide range of new applications; demonstrated by the numerous scientific publications that resulted from collaborative projects and user projects.

This project achieved its original objectives but MEMS technology is continuing to advance and there are undoubtedly many more new and exciting applications that can be facilitated by the work conducted within this project. Future work will see the ongoing development of the IMU platforms to incorporate improved MEMS sensors and new features to meet the needs of an expanding user base. Work will also continue on the AHRS algorithm and sensor calibration solutions to meet the specific needs of improved MEMS sensors and new applications.

# References

- [1] N. Yazdi, F. Ayazi, and K. Najafi, “Micromachined inertial sensors,” *Proceedings of the IEEE*, vol. 86, pp. 1640–1659, Aug 1998.
- [2] M. Perlmutter and L. Robin, “High-performance, low cost inertial mems: A market in motion!,” in *Position Location and Navigation Symposium (PLANS), 2012 IEEE/ION*, pp. 225–229, April 2012.
- [3] A. Jimenez, F. Seco, C. Prieto, and J. Guevara, “A comparison of pedestrian dead-reckoning algorithms using a low-cost mems imu,” in *Intelligent Signal Processing, 2009. WISP 2009. IEEE International Symposium on*, pp. 37–42, Aug 2009.
- [4] J. L. Crassidis, F. L. Markley, and Y. Cheng, “Nonlinear attitude filtering methods,” *Journal of Guidance Control and Dynamics*, vol. 30, no. 1, pp. 12–28, 2007.
- [5] S. Madgwick, A. J. L. Harrison, and R. Vaidyanathan, “Estimation of imu and marg orientation using a gradient descent algorithm,” in *Rehabilitation Robotics (ICORR), 2011 IEEE International Conference on*, pp. 1–7, 2011.
- [6] “Seb Madgwick - Google Scholar Citations.” [http://scholar.google.co.uk/citations?user=pVoC0\\_cAAAAJ&hl=en](http://scholar.google.co.uk/citations?user=pVoC0_cAAAAJ&hl=en), 2014. [Accessed: February 26, 2014].
- [7] “Google Code - imumargalgorithm30042010sohm.” <http://code.google.com/p/imumargalgorithm30042010sohm/>, 2014. [Accessed: February 26, 2014].
- [8] S. O. Madgwick, A. J. Harrison, P. M. Sharkey, R. Vaidyanathan, and W. S. Harwin, “Measuring motion with kinematically redundant accelerometer arrays: Theory, simulation and implementation,” *Mechatronics*, vol. 23, no. 5, pp. 518 – 529, 2013.
- [9] S. Madgwick and T. Mitchell, “x-OSC: A versatile wireless I/O device for creative/music applications,” in *SMC Sound and Music Computing Conference 2013*, (KTH Royal Institute of Technology, Stockholm), August 2013.
- [10] T. Mitchell, S. Madgwick, S. Rankin, G. Hilton, and A. Nix, “Making the most of wi-fi: Optimisations for wireless robust live music performance,” in *Proceedings of the International Conference on New Interfaces for Musical*

- Expression (NIME2014)*, 2014. [under review].
- [11] T. Mitchell, S. Madgwick, and I. Heap, “Musical interaction with hand posture and orientation: A toolbox of gestural control mechanisms,” in *International Conference on New Interfaces for Musical Expression (NIME)*, (University of Michigan, Ann Arbor), May 2012.
  - [12] J. Carberry, G. Hinchly, J. Buckerfield, E. Tayler, T. Burton, S. Madgwick, and R. Vaidyanathan, “Parametric design of an active ankle foot orthosis with passive compliance,” in *Computer-Based Medical Systems (CBMS), 2011 24th International Symposium on*, pp. 1–6, 2011.
  - [13] R. Bogue, “Mems sensors: past, present and future,” *Sensor Review*, vol. 27, pp. 7–13, 2007.
  - [14] M. Douglass, “Lifetime estimates and unique failure mechanisms of the digital micromirror device (dmd),” in *Reliability Physics Symposium Proceedings, 1998. 36th Annual. 1998 IEEE International*, pp. 9–16, March 1998.
  - [15] V. Kempe, *Inertial MEMS - Principles and practise*. Cambridge University Press, 2011.
  - [16] C. Wingrave, B. Williamson, P. D. Varcholik, J. Rose, A. Miller, E. Charbonneau, J. Bott, and J. LaViola, “The wiimote and beyond: Spatially convenient devices for 3d user interfaces,” *Computer Graphics and Applications, IEEE*, vol. 30, pp. 71–85, March 2010.
  - [17] T. Schrömer, B. Poppinga, N. Henze, and S. Boll, “Gesture recognition with a wii controller,” in *Proceedings of the 2Nd International Conference on Tangible and Embedded Interaction, TEI ’08*, (New York, NY, USA), pp. 11–14, ACM, 2008.
  - [18] L. Gallo, G. De Pietro, and I. Marra, “3d interaction with volumetric medical data: Experiencing the wiimote,” in *Proceedings of the 1st International Conference on Ambient Media and Systems, Ambi-Sys ’08*, (ICST, Brussels, Belgium, Belgium), pp. 14:1–14:6, ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), 2008.
  - [19] E. L. Kramer, “Analog devices introduces industry’s lowest power mems accelerometer,” June 2012.
  - [20] M. Lemkin, B. Boser, D. Auslander, and J. Smith, “A 3-axis force balanced accelerometer using a single proof-mass,” in *Solid State Sensors and Actuators, 1997. TRANSDUCERS ’97 Chicago., 1997 International Conference on*, vol. 2, pp. 1185–1188 vol.2, Jun 1997.
  - [21] Y. Dong, P. Zwahlen, A. M. Nguyen, F. Rudolf, and J. M. Stauffer, “High performance inertial navigation grade sigma-delta mems accelerometer,” in *Position Location and Navigation Symposium (PLANS), 2010 IEEE/ION*, pp. 32–36, May 2010.

- [22] M. Pastre, M. Kayal, H. Schmid, P. Zwahlen, Y. Dong, and A. M. Nguyen, “A navigation-grade mems accelerometer based on a versatile front end,” in *IECON 2011 - 37th Annual Conference on IEEE Industrial Electronics Society*, pp. 4038–4043, Nov 2011.
- [23] P. Zwahlen, Y. Dong, A.-M. Nguyen, F. Rudolf, J.-M. Stauffer, P. Ullah, and V. Ragot, “Breakthrough in high performance inertial navigation grade sigma-delta mems accelerometer,” in *Position Location and Navigation Symposium (PLANS), 2012 IEEE/ION*, pp. 15–19, April 2012.
- [24] O. Aydin and T. Akin, “A bulk-micromachined fully differential mems accelerometer with split interdigitated fingers,” *Sensors Journal, IEEE*, vol. 13, pp. 2914–2921, Aug 2013.
- [25] M. Kline, Y. Yeh, B. Eminoglu, H. Najar, M. Daneman, D. Horsley, and B. Boser, “Quadrature fm gyroscope,” in *Micro Electro Mechanical Systems (MEMS), 2013 IEEE 26th International Conference on*, pp. 604–608, Jan 2013.
- [26] M. Weinberg and A. Kourepinis, “Error sources in in-plane silicon tuning-fork mems gyroscopes,” *Microelectromechanical Systems, Journal of*, vol. 15, pp. 479–491, June 2006.
- [27] F. Gulmammadov, “Analysis, modeling and compensation of bias drift in mems inertial sensors,” in *Recent Advances in Space Technologies, 2009. RAST '09. 4th International Conference on*, pp. 591–596, June 2009.
- [28] D. Keymeulen, C. Peay, K. Yee, and D. Li, “Effect of temperature on mems vibratory rate gyroscope,” in *Aerospace Conference, 2005 IEEE*, pp. 1–6, March 2005.
- [29] M. J. Caruso, C. H. Smith, T. Bratland, and R. Schneider, “A new perspective on magnetic field sensing.”
- [30] S. Maus, S. Macmillan, S. McLean, M. Nair, C. Rollins, S. Macmillan, B. Hamilton, and A. Thomson, “The US/UK world magnetic model for 2010-2015,” tech. rep., NOAA National Geophysical Data Center and British Geological Survey Geomagnetism Team, 2009.
- [31] A. Trusov, I. Prikhodko, S. Zotov, and A. Shkel, “High-q and wide dynamic range inertial mems for north-finding and tracking applications,” in *Position Location and Navigation Symposium (PLANS), 2012 IEEE/ION*, pp. 247–251, April 2012.
- [32] I. Prikhodko, A. Trusov, and A. Shkel, “North-finding with 0.004 radian precision using a silicon mems quadruple mass gyroscope with q-factor of 1 million,” in *Micro Electro Mechanical Systems (MEMS), 2012 IEEE 25th International Conference on*, pp. 164–167, Jan 2012.
- [33] I. Prikhodko, S. Zotov, A. Trusov, and A. Shkel, “What is mems gyrocompassing? comparative analysis of maytagging and carouseling,” *Microelectromechanical Systems, Journal of*, vol. 22, pp. 1257–1266, Dec

2013.

- [34] M. Tanenhaus, D. Carhoun, T. Geis, E. Wan, and A. Holland, "Miniature imu/ins with optimally fused low drift mems gyro and accelerometers for applications in gps-denied environments," in *Position Location and Navigation Symposium (PLANS), 2012 IEEE/ION*, pp. 259–264, April 2012.
- [35] S. K. Hong, "Fuzzy logic based closed-loop strapdown attitude system for unmanned aerial vehicle (uav)," *Sensors and Actuators A: Physical*, vol. 107, no. 2, pp. 109 – 118, 2003.
- [36] B. Barshan and H. Durrant-Whyte, "Inertial navigation systems for mobile robots," *Robotics and Automation, IEEE Transactions on*, vol. 11, pp. 328–342, Jun 1995.
- [37] L. Ojeda and J. Borenstein, "Flexnav: fuzzy logic expert rule-based position estimation for mobile robots on rugged terrain," in *Proc. IEEE International Conference on Robotics and Automation ICRA '02*, vol. 1, pp. 317–322, May 11–15, 2002.
- [38] D. H. Titterton and J. L. Weston, *Strapdown inertial navigation technology*. The Institution of Electrical Engineers, 2004.
- [39] S. Beauregard, "Omnidirectional pedestrian navigation for first responders," in *Proc. 4th Workshop on Positioning, Navigation and Communication WPNC '07*, pp. 33–36, Mar. 22–22, 2007.
- [40] H. J. Luinge and P. H. Veltink, "Inclination measurement of human movement using a 3-d accelerometer with autocalibration," vol. 12, pp. 112–121, Mar. 2004.
- [41] H. Zhou and H. Hu, "Human motion tracking for rehabilitation—a survey," *Biomedical Signal Processing and Control*, vol. 3, no. 1, pp. 1 – 18, 2008.
- [42] E. A. Heinz, K. S. Kunze, M. Gruber, D. Bannach, and P. Lukowicz, "Using wearable sensors for real-time recognition tasks in games of martial arts - an initial experiment," in *Proc. IEEE Symposium on Computational Intelligence and Games*, pp. 98–102, May 22–24, 2006.
- [43] S. Moafipoor, D. A. Grejner-Brzezinska, and C. Toth, "Adaptive calibration of a magnetometer compass for a personal navigation system," in *International Global Navigation Satellite Systems Society (IGNSS Symposium 2007)*, (The University of New South Wales, Sydney, Australia), December 2007.
- [44] M. Cordoba, "Attitude and heading refernce system i-ahrs for the efigenia autonomous unmanned aerial vehicles uav based on mems sensor and a neural network strategy for attitude estimation," in *Control Automation, 2007. MED '07. Mediterranean Conference on*, pp. 1–8, June 2007.
- [45] P. Martin and E. Salaun, "Invariant observers for attitude and heading estimation from low-cost inertial and magnetic sensors," in *Decision and*

- Control, 2007 46th IEEE Conference on*, pp. 1039–1045, Dec 2007.
- [46] R. E. Kalman, “A new approach to linear filtering and prediction problems,” *Journal of Basic Engineering*, vol. 82, pp. 35–45, 1960.
- [47] R. G. Brown and P. Y. C. Hwang, *Introduction to Random Signals and Applied Kalman Filtering*. John Wiley & Sons, Inc., third edition ed., 1997.
- [48] Xsens Technologies B.V., *MTi and MTx User Manual and Technical Documentation*, mt0100p, revision O ed., October 2010.
- [49] LORD Corporation, MicroStrain Sensing Systems, 459 Hurricane Lane, Suite 102, Williston, VT 05495 USA, *3DM-GX3-45 - Miniature GPS-Aided Inertial Navigation System (GPS/INS)*, version 8400-0035 revision 003 ed., 2014.
- [50] InterSense, LLC, 700 Technology Park Drive, Suite 102, Billerica, MA 01821, USA, *InertiaCube4 - Precision inertial orientation sensor*, 072-00136-0k12 revision 1.0 ed.
- [51] VectorNav Technologies, LLC, *VN-100 User Manual*, revision 1.2.10 ed., 2009.
- [52] CH Robotics LLC, *UM6 Ultra-Miniature Orientation Sensor Datasheet*, revision 2.4 ed., October 2013.
- [53] YEI Technology, 630 Second Street, Portsmouth, Ohio 45662, *3-Space Sensor Embedded Ultra-Miniature Attitude & Heading Reference System User's Manual*, August 2013.
- [54] R. B. McGhee, E. R. Bachmann, X. Yun, and M. J. Zyda, “Real-time tracking and display of human limb segment motions using sourceless sensors and a quaternion-based filtering algorithm - part i: Theory,” tech. rep., NAVAL POSTGRADUATE SCHOOL, Monterey, California, 2000.
- [55] R. Mahony, T. Hamel, and J.-M. Pflimlin, “Complementary filter design on the special orthogonal group  $so(3)$ ,” in *Decision and Control, 2005 and 2005 European Control Conference. CDC-ECC '05. 44th IEEE Conference on*, pp. 1477–1484, Dec 2005.
- [56] T. S. Yoo, S. K. Hong, H. M. Yoon, and S. Park, “Gain-scheduled complementary filter design for a mems based attitude and heading reference system,” *Sensors*, vol. 11, no. 4, pp. 3816–3830, 2011.
- [57] A.-J. Baerveldt and R. Klang, “A low-cost and low-weight attitude estimation system for an autonomous helicopter,” in *Intelligent Engineering Systems, 1997. INES '97. Proceedings., 1997 IEEE International Conference on*, pp. 391–395, Sep 1997.
- [58] J. Roberts, P. Corke, and G. Buskey, “Low-cost flight control system for a small autonomous helicopter,” in *Robotics and Automation, 2003. Proceedings. ICRA '03. IEEE International Conference on*, vol. 1, pp. 546–551 vol.1, Sept 2003.

- [59] R. Mahony, T. Hamel, and J.-M. Pflimlin, “Nonlinear complementary filters on the special orthogonal group,” *Automatic Control, IEEE Transactions on*, vol. 53, pp. 1203–1218, june 2008.
- [60] J. Marins, X. Yun, E. Bachmann, R. Mcghee, and M. Zyda, “An extended kalman filter for quaternion-based orientation estimation using marg sensors,” in *Intelligent Robots and Systems, 2001. Proceedings. 2001 IEEE/RSJ International Conference on*, vol. 4, pp. 2003–2011 vol.4, 2001.
- [61] R. A. Hyde, L. P. Ketteringham, S. A. Neild, and R. J. S. Jones, “Estimation of upper-limb orientation based on accelerometer and gyroscope measurements,” vol. 55, pp. 746–754, Feb. 2008.
- [62] M. Euston, P. Coote, R. Mahony, J. Kim, and T. Hamel, “A complementary filter for attitude estimation of a fixed-wing uav,” in *Intelligent Robots and Systems, 2008. IROS 2008. IEEE/RSJ International Conference on*, pp. 340–345, Sept 2008.
- [63] E. R. Bachmann, R. B. McGhee, X. Yun, and M. J. Zyda, “Inertial and magnetic posture tracking for inserting humans into networked virtual environments,” pp. 9–16, 2001.
- [64] J. M. Pflimlin, T. Hamel, and P. Souères, P.res, “Nonlinear attitude and gyroscope’s bias estimation for a VTOL UAV,” *International Journal of Systems Science*, vol. 38, no. 3, pp. 197–210, 2007.
- [65] S. Sukkarieh, P. Gibbens, B. Grocholsky, K. Willis, and H. F. Durrant-Whyte, “A low-cost, redundant inertial measurement unit for unmanned air vehicles,” *The International Journal of Robotics Research*, vol. 19, pp. 1089–1103, November 2000.
- [66] K. Mostov, A. Soloviev, and T.-K. Koo, “Accelerometer based gyro-free multi-sensor generic inertial device for automotive applications,” *Intelligent Transportation System, 1997. ITSC ’97., IEEE Conference on*, pp. 1047 –1052, nov. 1997.
- [67] Y. K. Peng and M. Golnaraghi, “A vector-based gyro-free inertial navigation system by integrating existing accelerometer network in a passenger vehicle,” in *Position Location and Navigation Symposium, 2004. PLANS 2004*, pp. 234 – 242, 26-29 2004.
- [68] H. Weinberg, “Mems sensors are driving the automotive industry,” *Sensors*, vol. 19, no. 2, pp. 36–41, 2002.
- [69] V. Krishnan, “Measurement of angular velocity and linear acceleration using linear accelerometers,” *Journal of the Franklin Institute*, vol. 280, no. 4, pp. 307 – 315, 1965.
- [70] A. R. Schuler, A. Grammatikos, and K. A. Fegley, “Measuring rotational motion with linear accelerometers,” in *Aerospace and Electronic Systems, IEEE Transactions on*, vol. AES-3, pp. 465 – 472, May 1967.

- [71] Kionix, *Using Two Tri-Axis Accelerometers for Rotational Measurements (AN 019)*, Jan 2008. Application Note.
- [72] W. Dong, I.-M. Chen, K. Y. Lim, and Y. K. Goh, "Measuring uniaxial joint angles with a minimal accelerometer configuration," in *i-CREAtE '07: Proceedings of the 1st international convention on Rehabilitation engineering &#38; assistive technology*, (New York, NY, USA), pp. 88–91, ACM, 2007.
- [73] J.-H. Chen, S.-C. Lee, and D. B. DeBra, "Gyroscope free strapdown inertial measurement unit by six linear accelerometers," *JOURNAL OF GUIDANCE, CONTROL, AND DYNAMICS*, vol. 17, pp. 286–290, March-April 1994.
- [74] S.-C. Lee and C.-Y. Liu, "An innovative estimation method with own-ship estimator for an allaccelerometer-type inertial navigation system," *International Journal of Systems Science*, vol. 30, no. 12, pp. 1259–1266, 1999.
- [75] C.-W. Tan and S. Park, "Design of accelerometer-based inertial navigation systems," *Instrumentation and Measurement, IEEE Transactions on*, vol. 54, no. 6, pp. 2520–2530, 2005.
- [76] C.-W. Tan, K. Mostov, and P. Varaiya, "Feasibility of a gyroscope-free inertial navigation system for tracking rigid body motion," research report, California Partners for Advanced Transit and Highways (PATH), Berkeley, CA, May 2000.
- [77] K. Liu, "Discussion on measurement of angular acceleration of a rigid body using linear accelerometers," *J. Appl. Mech*, vol. 43, pp. 977–978, 1976.
- [78] A. J. Padgaonkar, K. Krieger, and A. I. King, "Measurement of angular acceleration of a rigid body using linear accelerometers," *Journal of Applied Mechanics*, vol. 42(3), pp. 552–556, September 1976.
- [79] D. Giansanti, V. Macellari, G. Maccioni, and A. Cappozzo, "Is it feasible to reconstruct body segment 3-d position and orientation using accelerometric data?," *Biomedical Engineering, IEEE Transactions on*, vol. 50, pp. 476 –483, apr. 2003.
- [80] Q. Wang, M. Ding, and P. Zhao, "A new scheme of non-gyro inertial measurement unit for estimating angular velocity," vol. 2, pp. 1564 – 1567 Vol.2, 2-6 2003.
- [81] W. Xiao-xu, X. Hong-xiang, X. Quan-xi, and S. Ming, "Design and simulation analysis of gyroscope-free inertial measurement unit," *Journal of Chinese Inertial Technology*, vol. 16, pp. 154–158, April 2008.
- [82] J. Xudan, M. Zheng, W. Fuling, and W. Yali, "Research on gyroscope free strapdown inertial navigation system based on 3-axis accelerometer," in *Electronic Measurement and Instruments, 2007. ICEMI '07. 8th International Conference on*, pp. 1–967–1–970, Aug 2007.

- [83] C.-W. Ho and P. chun Lin, "Design and implementation of a 12-axis accelerometer suite," *Intelligent Robots and Systems, 2009. IROS 2009. IEEE/RSJ International Conference on*, pp. 2197 –2202, oct. 2009.
- [84] P. Schopp, L. Klingbeil, C. Peters, A. Buhmann, and Y. Manoli, "Sensor fusion algorithm and calibration for a gyroscope-free imu," *Procedia Chemistry*, vol. 1, no. 1, pp. 1323 – 1326, 2009. *[ce:title; Proceedings of the Eurosensors XXIII conference]*
- [85] P. Schopp, L. Klingbeil, C. Peters, and Y. Manoli, "Design, geometry evaluation, and calibration of a gyroscope-free inertial measurement unit," *Sensors and Actuators A: Physical*, vol. 162, no. 2, pp. 379 – 387, 2010. *[ce:title; Eurosensors XXIII, 2009]*
- [86] K. Parsa, J. Angeles, and A. K. Misra, "Rigid-body pose and twist estimation using an accelerometer array," *Archive of Applied Mechanics*, vol. 74, pp. 223–236, Dec 2004.
- [87] Z. Qin, L. Baron, and L. Birglen, "Robust design of inertial measurement units based on accelerometers," *Journal of Dynamic Systems, Measurement, and Control*, vol. 131, no. 3, p. 031010, 2009.
- [88] P. Cappa, L. Masia, and F. Patanè, "Numerical validation of linear accelerometer systems for the measurement of head kinematics," *Journal of Biomechanical Engineering*, vol. 127, no. 6, pp. 919–928, 2005.
- [89] J. Vreeburg, "Analysis of the data from a distributed set of accelerometers, for reconstruction of set geometry and its rigid body motion," Tech. Rep. NLR-TP-98343, National Aerospace Laboratory NLR, 1999.
- [90] K. Parsa, T. Lasky, and B. Ravani, "Design and implementation of a mechatronic, all-accelerometer inertial measurement unit," *Mechatronics, IEEE/ASME Transactions on*, vol. 12, pp. 640 –650, dec. 2007.
- [91] P. Cappa, F. Patanè, and S. Rossi, "A redundant accelerometric cluster for the measurement of translational and angular acceleration and angular velocity of the head," *Journal of Medical Devices*, vol. 1, no. 1, pp. 14–22, 2007.
- [92] J. J. Craig, *Introduction to Robotics Mechanics and Control*. Pearson Education International, 2005.
- [93] J. B. Kuipers, *Quaternions and Rotation Sequences: A Primer with Applications to Orbits, Aerospace and Virtual Reality*. Princeton University Press, 1999.
- [94] "MathsWorks Documentation Center - dcm2quat." <http://www.mathworks.co.uk/help/aerotbx/ug/dcm2quat.html>, 2014. [Accessed: February 20, 2014].
- [95] "The Open Toolkit library 1.1 - OpenTK.Quaternion Struct Reference." [http://www.openTK.com/files/doc/struct\\_open\\_t\\_k\\_](http://www.openTK.com/files/doc/struct_open_t_k_)

- 1\_1\_quaternion.html, 2014. [Accessed: February 23, 2014].
- [96] “Microsoft Developer Network - Quaternion Constructor (Single, Single, Single, Single).” <http://msdn.microsoft.com/en-us/library/bb195806.aspx>, 2014. [Accessed: February 19, 2014].
- [97] I. Y. Bar-Itzhack, “New method for extracting the quaternion from a rotation matrix,” *AIAA Journal of Guidance, Control and Dynamics*, vol. 23, pp. 1085–1087, November-December 2000. (Engineering Note).
- [98] J. M. Cooke, M. J. Zyda, D. R. Pratt, and R. B. Mcghee, “NPSNET: Flight simulation dynamic modeling using quaternions,” *Presence*, vol. 1, pp. 404–420, 1992.
- [99] J. A. Jacobs, *The earth's core*, vol. 37 of International geophysics series. Academic Press, 2 ed., 1987.
- [100] E. R. Bachmann, X. Yun, and C. W. Peterson, “An investigation of the effects of magnetic variations on inertial/magnetic orientation sensors,” in *Proc. IEEE International Conference on Robotics and Automation ICRA '04*, vol. 2, pp. 1115–1122, Apr. 2004.
- [101] W. de Vries, H. Veeger, C. Baten, and F. van der Helm, “Magnetic distortion in motion labs, implications for validating inertial magnetic sensors,” *Gait & Posture*, vol. 29, no. 4, pp. 535–541, 2009.
- [102] Speake & Co Limited, “Autocalibration algorithms for FGM type sensors.” Application note.
- [103] D. Gebre-Egziabher, G. H. Elkaim, J. D. Powell, and B. W. Parkinson, “Calibration of strapdown magnetometers in magnetic field domain,” *Journal of Aerospace Engineering*, vol. 19, no. 2, pp. 87–102, 2006.
- [104] A. Sabatini, “Quaternion-based extended kalman filter for determining orientation by inertial and magnetic sensing,” *Biomedical Engineering, IEEE Transactions on*, vol. 53, pp. 1346–1356, July 2006.
- [105] Xsens Technologies B.V., *MTi and MTx User Manual and Technical Documentation*. Pantheon 6a, 7521 PR Enschede, The Netherlands, May 2009.
- [106] Vicon Motion Systems Limited., *Vicon MX Hardware*. 5419 McConnell Avenue, Los Angeles, CA 90066, USA, 1.6 ed., 2004.
- [107] Vicon Motion Systems Limited., *Vicon Nexus Product Guide - Foundation Notes*. 5419 McConnell Avenue, Los Angeles, CA 90066, USA, 1.2 ed., November 2007.
- [108] LORD Corporation, MicroStrain Sensing Systems, 459 Hurricane Lane, Suite 102, Williston, VT 05495, *3DM-GX3-25 Attitude Heading Reference System Data Sheet*, version 8400-0033 revision 002 ed., 2013.
- [109] VectorNav Technologies, LLC, *VN-100 Product Brief*, 12-0002-R1 ed.

- [110] InterSense, Inc., *InertiaCube2+ Manual*. 36 Crosby Drive, Suite 150, Bedford, MA 01730, USA, 1.0 ed., 2008.
- [111] P. Martin and E. Salaün, “Design and implementation of a low-cost observer-based attitude and heading reference system,” *Control Engineering Practice*, vol. 18, no. 7, pp. 712 – 722, 2010. Special Issue on Aerial Robotics.
- [112] P. Cardou, G. Fournier, and P. Gagnon, “A nonlinear program for angular-velocity estimation from centripetal-acceleration measurements,” *Mechatronics, IEEE/ASME Transactions on*, vol. PP, no. 99, pp. 1 –13, 2010.
- [113] National Imagery and Mapping Agency, “World geodetic system 1984: Its definition and relationships with local geodetic systems,” Tech. Rep. NIMA TR8350.2, U.S. Department of National Defense, January 2000.
- [114] K. Parsa, J. Angeles, and A. Misra, “Attitude calibration of an accelerometer array,” *Robotics and Automation, 2002. Proceedings. ICRA '02. IEEE International Conference on*, vol. 1, pp. 129 – 134 vol.1, 2002.
- [115] Ideal Aerosmith, Inc., East Grand Forks, MN USA, Grand Forks, ND USA, *2003HP Series Three-Axis Precision Motion Simulation Table Systems*.
- [116] A. Rosy Amlani, “Quotation: IMU testing.” Private communication, May 2012.
- [117] K. Shcheglov, C. Evans, R. Gutierrez, and T. Tang, “Temperature dependent characteristics of the jpl silicon mems gyroscope,” in *Aerospace Conference Proceedings, 2000 IEEE*, vol. 1, pp. 403–411 vol.1, 2000.
- [118] P. Aggarwal, Z. Syed, X. Niu, and N. El-Sheimy, “A standard testing and calibration procedure for low cost mems inertial sensors and units,” *THE JOURNAL OF NAVIGATION*, vol. 61, pp. 323–336, 2008.
- [119] “Ieee standard specification format guide and test procedure for coriolis vibratory gyros,” *IEEE Std 1431-2004*, pp. 1–78, Dec 2004.
- [120] “IEEE standard specification format guide and test procedure for linear, single-axis, non-gyroscopic accelerometers,” *IEEE Std 1293-1998*, pp. i–, 1999.
- [121] InvenSense, Inc., *MPU-6000 and MPU-6050 Product Specification Revision 3.4*, revision 3.4 ed., 2013.
- [122] Honeywell International Inc., *3-Axis Digital Compass IC HMC5883L*, revision E ed., February 2013.
- [123] by Harvey Weinberg, “Gyro mechanical performance: The most important parameter,” Tech. Rep. MS-2158, Analog Devices, Inc., Semptember 2011.
- [124] Invensense Developers Support, “Re: Development of an accurate MPU-6050 error model.” Private communication, March 2012.
- [125] “IEEE standard specification format guide and test procedure for single-axis

- interferometric fiber optic gyros," *IEEE Std 952-1997*, pp. i–, 1998.
- [126] D. W. Stockwell, "Bias stability measurement: Allan variance," tech. rep., Crossbow Technology, Inc., 2003.
- [127] I. Skog and P. Händel, "Calibration of a mems inertial measurement unit," in *in Proc. XVII IMEKO WORLD CONGRESS, (Rio de Janeiro)*, 2006.
- [128] Xsens Technologies B.V., *MTw User Manual*, revision F ed., January 2013.
- [129] "UM6 orientation sensor product webpage." <http://www.chrobotics.com/shop/orientation-sensor-um6>, 2014. [Accessed: February 25, 2014].
- [130] Ideal Aerosmith, Inc., East Grand Forks, MN USA, Grand Forks, ND USA, *1270VS Series Single-Axis Rate Table Datasheet*, rev: q ed.
- [131] Panasonic Company Division of Matsushita Electric Corporation of America, One Panasonic Way, Secaucus, New Jersey 07094, *SL-1200MK2 Turntable System Service Manual (M), (MC)*.
- [132] Thurlby Thandar Instruments Ltd., Glebe Road, Huntingdon, Cambridgeshire, PE29 7DR, England (United Kingdom), *TF960 6GHz Universal Counter Instruction Manual*, 48581-1440 issue 2 ed.
- [133] "Application note AN3192: Using LSM303DLH for a tilt compensated electronic compass," Tech. Rep. Doc ID 17353 Rev 1, STMicroelectronics, August 2010.
- [134] J. Vasconcelos, G. Elkaim, C. Silvestre, P. Oliveira, and B. Cardeira, "A geometric approach to strapdown magnetometer calibration in sensor frame," *Aerospace and Electronic Systems, IEEE Transactions on*, vol. 47, pp. 1293–1306, April 2011.
- [135] Y. Petrov, "Matlab central - ellipsoid fit." <http://www.mathworks.co.uk/matlabcentral/fileexchange/24693-ellipsoid-fit>, July 2009. [Accessed: March 26, 2014].
- [136] InvenSense, Inc., *MPU-6000 and MPU-6050 Register Map and Descriptions Revision 4.2*, revision: 4.2 ed., 2013.
- [137] Y. Xiaoping, E. Bachmann, and R. McGhee, "A simplified quaternion-based algorithm for orientation estimation from earth gravity and magnetic field measurements," *Instrumentation and Measurement, IEEE Transactions on*, vol. 57, pp. 638 –650, mar. 2008.
- [138] N. S. Nise, *Control Systems Engineering*. John Wiley & Sons, Inc., fourth edition ed., 2004.
- [139] STMicroelectronics, *TA0343 Technical article - Everything about STMicroelectronics' 3-axis digital MEMS gyroscopes*, doc id 022032 rev 1 ed., July 2011.
- [140] D. Vargha and M. Maia, "An overview of motion processing solutions for consumer products," tech. rep., InvenSense, Inc., 1197 Borregas Avenue,

## References

---

- Sunnyvale, CA 94089.
- [141] Microchip Technology Inc., *dsPIC33FJ32GP302/304, dsPIC33FJ64GPX02/X04, and dsPIC33FJ128GPX02/X04 Data Sheet*, DS70292G ed., 2012.
  - [142] Maxim Integrated Products, *MAX1811 - USB-Powered Li+ Charger*, rev 2 ed., 2003.
  - [143] Future Technology Devices International Limited, *FT232R USB UART IC Datasheet*, version 2.10 ed., 2012.
  - [144] Future Technology Devices International Ltd., *TN\_132 Adding FTDI Devices VCP Driver Support to Android*, version 1.1 ed., 2011.
  - [145] Roving Networks, Inc., *RN41/RN41N Class 1 Bluetooth Module*, version 3.42r ed., November 2013.
  - [146] x-io Technologies Limited, *x-IMU User Manual*, version 5.2 ed., November 2013.
  - [147] bluetooth.org, *BLUETOOTH SPECIFICATION Version 2.1 + EDR*, July 2007.
  - [148] Microchip Technology Inc., *MCP73831/2 Datasheet*, DS20001984F ed., 2013.
  - [149] Micrel Inc., *MIC5319 Datasheet*, M9999-071812 ed., July 2012.
  - [150] Microchip Technology Inc., *PIC24FJ64GA104 Family Data Sheet*, DS39951C ed., 2010.
  - [151] Digi International Inc., *XBee XBee-PRO RF Modules: Product Manual v1.xEx - 802.15.4 Protocol*, 90000982\_M ed., July 2013.
  - [152] Roving Networks, Inc., *RN41XV & RN42XV Bluetooth Module*, version 1.0 ed., December 2013.
  - [153] Digi International Inc., *XBee XBee-PRO ZB RF Modules*, 90000976\_P ed., April 2013.
  - [154] Digi International Inc., *XBee-PRO 900HPXBee-PRO XSC RF Modules*, 90002173\_H ed., September 2013.
  - [155] Digi International Inc., *XBee Wi-Fi RF Module*, 90002180\_G ed., September 2013.
  - [156] B. Ivey, *AN1416 - Low-Power Design Guide*. Microchip Technology Inc., ds01416a ed., 2011.
  - [157] “IEEE standard for local and metropolitan area networks—part 15.4: Low-rate wireless personal area networks (lr-wpans),” *IEEE Std 802.15.4-2011 (Revision of IEEE Std 802.15.4-2006)*, pp. 1–314, 2011.
  - [158] APDM Inc., 2828 Southwest Corbett Avenue, Portland, OR 97201, *opal - Technical Specifications*, 2012.

- [159] M. Wright, “Open sound control: An enabling technology for musical networking,” *Org. Sound*, vol. 10, pp. 193–200, December 2005.
- [160] “opensoundcontrol.org - implementations.” <http://opensoundcontrol.org/implementations>, 2014. [Accessed March 14, 2013].
- [161] A. Schmeder and A. Freed, “uosc: The open sound control reference platform for embedded devices,” in *NIME*, (Genova, Italy), 2008.
- [162] x-io Technologies Limited, *x-OSC User Manual*, version 0.3 ed., November 2013.
- [163] Microchip Technology Inc., *dsPIC33EPXXX(GP,MC,MU)806,810,814 and PIC24EPXXX(GP,GU)810,814 Data Sheet*, DS70616F ed., 2012.
- [164] Microchip Technology Inc., *MRF24WG0MA/MB Data Sheet*, DS70686B ed., 2012.
- [165] Analog Devices, Inc., One Technology Way, P.O. Box 9106, Norwood, MA 02062-9106, U.S.A., *ADP3338 - High Accuracy, Ultralow IQ, 1 A, anyCAP Low Dropout Regulator*, rev. b ed., 2005.
- [166] Philips Electronics N.V., *74LVC125A - Quad buffer/line driver with 5 V tolerant input/outputs; 3-state*, 9397 750 10532 ed., May 2003.
- [167] Abracon Corporation, 30332 Esperanza, Rancho Santa Margarita, California 92688, *ABM8 - ULTRA MINIATURE CERAMIC SMD CRYSTAL*, August 2007.
- [168] P. Fuxjager, D. Valerio, and F. Ricciato, “The myth of non-overlapping channels: interference measurements in ieee 802.11,” in *Wireless on Demand Network Systems and Services, 2007. WONS '07. Fourth Annual Conference on*, pp. 1–8, Jan 2007.
- [169] A. Willig, K. Matheus, and A. Wolisz, “Wireless technology in industrial networks,” *Proceedings of the IEEE*, vol. 93, pp. 1130–1151, June 2005.
- [170] T. Mitchell and I. Heap, “Soundgrasp: A gestural interface for the performance of live music,” in *Proceedings of the International Conference on New Interfaces for Musical Expression (NIME)*, (Oslo, Norway), pp. 465 – 468, May 2011.
- [171] D. Rodriguez-Silva, F. Gil-Castineira, F. Gonzalez-Castano, R. Duro, F. Lopez-Pena, and J. Vales-Alonso, “Human motion tracking and gait analysis: a brief review of current sensing systems and integration with intelligent environments,” pp. 166 –171, 2008.
- [172] A. Godfrey, R. Conway, D. Meagher, and G. ÓLaighin, “Direct measurement of human movement by accelerometry,” *Medical Engineering & Physics*, vol. 30, pp. 1364–1386, Dec 2008.
- [173] V. Pomeroy, E. Evans, and J. Richards, “Agreement between an electrogoniometer and motion analysis system measuring angular velocity of

- the knee during walking after stroke," *Physiotherapy*, vol. 92, no. 3, pp. 159 – 165, 2006.
- [174] A. Vega-Gonzàlez and M. H. Granat, "Continuous monitoring of upper-limb activity in a free-living environment," *Archives of Physical Medicine and Rehabilitation*, vol. 86, no. 3, pp. 541 – 548, 2005.
- [175] K. Aminian, B. Najafi, C. Büla, P. F. Leyvraz, and P. Robert, "Spatio-temporal parameters of gait measured by an ambulatory system using miniature gyroscopes," *Journal of Biomechanics*, vol. 35, no. 5, pp. 689 – 699, 2002.
- [176] F. B. V. M. M. S. T. E. Uswatte G, Miltner WH, "Objective measurement of functional upper-extremity movement using accelerometer recordings transformed with a threshold filter," *Stroke*, pp. 662–667, 2000.
- [177] G. Uswatte, W. L. Foo, H. Olmstead, K. Lopez, A. Holand, and L. B. Simms, "Ambulatory monitoring of arm movement using accelerometry: An objective measure of upper-extremity rehabilitation in persons with chronic stroke," *Archives of Physical Medicine and Rehabilitation*, vol. 86, no. 7, pp. 1498 – 1501, 2005.
- [178] E. Bernmark and C. Wiktorin, "A triaxial accelerometer for measuring arm movements," *Applied Ergonomics*, vol. 33, no. 6, pp. 541 – 547, 2002.
- [179] A. V. Rowlands, P. W. M. Thomas, R. G. Eston, and R. Topping, "Validation of the RT3 triaxial accelerometer for the assessment of physical activity," *Medicine and Science in Sports and Exercise*, vol. 36, pp. 518–524, March 2004.
- [180] S. Y. Kumahara H, Tanaka H, "Daily physical activity assessment: What is the importance of upper limb movements vs whole body movements," *Int J Obes*, pp. 1105–1110, 2004.
- [181] P. Veltink and H. Franken, "Detection of knee unlock during stance by accelerometry," *Rehabilitation Engineering, IEEE Transactions on*, vol. 4, pp. 395 –402, Dec. 1996.
- [182] X. Yun, E. R. Bachmann, H. Moore, and J. Calusdian, "Self-contained position tracking of human movement using small inertial/magnetic sensor modules," in *ICRA*, pp. 2526–2533, 2007.
- [183] E. Foxlin, "Pedestrian tracking with shoe-mounted inertial sensors," *IEEE Comput. Graph. Appl.*, vol. 25, no. 6, pp. 38–46, 2005.
- [184] H. M. Schepers, H. Koopman, and P. H. Veltink, "Ambulatory assessment of ankle and foot dynamics," *IEEE Transactions on Biomedical Engineering*, vol. 54, no. 5, pp. 895–902, 2007.
- [185] R. Stirling, K. Fyfe, and G. Lachapelle, "Evaluation of a new method of heading estimation for pedestrian dead reckoning using shoe mounted sensors," *The Journal of Navigation*, vol. 58, no. 01, pp. 31–45, 2005.

- [186] F. Cavallo, A. Sabatini, and V. Genovese, “A step toward gps/ins personal navigation systems: real-time assessment of gait by foot inertial sensing,” pp. 1187 – 1191, aug. 2005.
- [187] D. Galinski and B. Dehez, “Evaluation of initialization procedures for estimating upper limb kinematics with marg sensors,” in *Biomedical Robotics and Biomechatronics (BioRob), 2012 4th IEEE RAS EMBS International Conference on*, pp. 1801–1806, 2012.
- [188] D. Kim, O. Hilliges, S. Izadi, A. D. Butler, J. Chen, I. Oikonomidis, and P. Olivier, “Digits: freehand 3d interactions anywhere using a wrist-worn gloveless sensor,” in *Proceedings of the 25th annual ACM symposium on User interface software and technology*, UIST ’12, (New York, NY, USA), pp. 167–176, ACM, 2012.
- [189] C. Osterloh, T. Pionteck, and E. Maehle, “Monsun II: A small and inexpensive auv for underwater swarms,” in *Robotics; Proceedings of ROBOTIK 2012; 7th German Conference on*, pp. 1–6, 2012.
- [190] C. Osterloh, B. Meyer, A. Amory, T. Pionteck, and E. Maehle, “Monsun II - towards autonomous underwater swarms for environmental monitoring,” in *IROS2012 - Workshop on Robotics for Environmental Monitoring*, (Vilamoura, Algarve, Portugal), pp. 1–6, 2012.
- [191] C. Osterloh, M. Litza, and E. Maehle, “Hard- and software architecture of a small autonomous underwater vehicle for environmental monitoring tasks,” in *Advances in Robotics Research* (T. Kröger and F. Wahl, eds.), pp. 347–356, Springer Berlin Heidelberg, 2009.
- [192] C. Osterloh and E. Maehle, “Low-power microcontroller-based acoustic modem for underwater robot communication,” in *Robotics (ISR), 2010 41st International Symposium on and 2010 6th German Conference on Robotics (ROBOTIK)*, pp. 1–6, 2010.
- [193] A. Amory, B. Meyer, C. Osterloh, T. Tosik, and E. Maehle, “Towards fault-tolerant and energy-efficient swarms of underwater robots,” in *Parallel and Distributed Processing Symposium Workshops PhD Forum (IPDPSW), 2013 IEEE 27th International*, pp. 1550–1553, 2013.
- [194] B. Meyer, C. Osterloh, and E. Maehle, “Monsun - a modular testbed for swarms of autonomous underwater vehicles,” in *ICRA2013 - Workshop on Many Robot Systems: Crossing the Reality Gap - From Single to Multi- to Many Robot Systems*, (Karlsruhe), 2013.
- [195] B. Meyer, K. Ehlers, C. Osterloh, and E. Maehle, “Smart-E: An autonomous omnidirectional underwater robot,” *Paladyn, Journal of Behavioral Robotics*, vol. 4, pp. 202–208, 2013.
- [196] J. Thomson, L. Kilcher, M. Richmond, J. Talbert, A. deKlerk, B. Polagye, M. Guerra, and R. Cienfuegos, “Tidal turbulence spectra from a compliant mooring,” in *Proceedings of the 1st Marine Energy Technology Symposium*,

2013.

- [197] W.-H. Chen, B.-Y. Huang, S.-R. Chen, S.-F. Yang, C.-C. Lee, D.-S. Yu, , and Y.-H. Lin, “FPGA and virtual reality based minimally invasive surgery training system,” in *Proceedings of 2013 FPGA Workshop an Design Contest*, (Southeast UNiversity, Nanjing, China), November 2013.
- [198] J. S. Akins, *DEVELOPMENT AND EVALUATION OF INSTRUMENTED SOCCER EQUIPMENT TO COLLECT ANKLE JOINT KINEMATICS IN THE FIELD*. PhD thesis, Swanson School of Engineering, University of Pittsburgh, 2013.
- [199] C. Brighton, E. Olsen, and T. Pfau, “Is a standalone inertial measurement unit accurate and precise enough for quantification of movement symmetry in the horse?,” *Computer Methods in Biomechanics and Biomedical Engineering*, vol. 0, pp. 1–6, 2013.
- [200] F. Hegel, F. Eyssel, and B. Wrede, “The social robot ‘flobi’: Key concepts of industrial design,” in *RO-MAN, 2010 IEEE*, pp. 107–112, 2010.
- [201] I. Lutkebohle, F. Hegel, S. Schulz, M. Hackel, B. Wrede, S. Wachsmuth, and G. Sagerer, “The bielefeld anthropomorphic robot head “Flobi”,” in *Robotics and Automation (ICRA), 2010 IEEE International Conference on*, pp. 3384–3391, 2010.
- [202] S. Schulz, F. Lier, I. Lutkebohle, and S. Wachsmuth, “Robot reality - a motion capture system that makes robots become human and vice versa,” in *Robotics and Automation (ICRA), 2013 IEEE International Conference on*, pp. 2126–2133, 2013.
- [203] L. Sussenbach, K. Pitsch, I. Berger, N. Riether, and F. Kummert, ““Can you answer questions, Flobi?”: Interactionally defining a robot’s competence as a fitness instructor,” in *RO-MAN, 2012 IEEE*, pp. 1121–1128, 2012.
- [204] F. Eyssel and N. Reich, “Loneliness makes the heart grow fonder (of robots) - on the effects of loneliness on psychological anthropomorphism,” in *Human-Robot Interaction (HRI), 2013 8th ACM/IEEE International Conference on*, pp. 121–122, 2013.
- [205] T. Michailidis, D. Polydorou, and J. Bullock, “A multimodal integration of sensory feedback modalities for dance performers,” in *Proceedings of the 1st Fascinate Conference*, (Falmouth University, Cornwall, UK), 28-30 August 2013.
- [206] S. Kasaei, S. Kasaei, and S. Kasaei, “Persia: A fully autonomous humanoid robot based on hybrid ann controller and vision system,” *Mechatronics*, 2013. [under review].
- [207] Glowacki, D. R., Tew, P., Mitchell, T., Kriefman, L, Hyde, J., Malcolm, L. J., Price, J., McIntosh-Smith, and S., “Sculpting molecular dynamics in real-time using human energy fields,” *Molecular Aesthetics*, 2013.

- 
- [208] Glowacki, D. R., Tew, P., Mitchell, T., Price, J., McIntosh-Smith, and S., “danceroom spectroscopy: Interactive quantum molecular dynamics accelerated on gpu architectures using opencl,” in *UK Many Core Development Conference 2012 (UKMAC ’12)*, (Bristol), 2012.
  - [209] A. Roudaut, R. Reed, T. Hao, and S. Subramanian, “Changibles: Analyzing and designing shape changing constructive assembly,” in *SIGCHI Conference on Human Factors in Computing Systems (CHI ’14)*, (ACM, New York, NY, USA), 2014. under review.
  - [210] P. Cardou and J. Angeles, “Singularity analysis of accelerometer strapdowns for the estimation of the acceleration field of a planar rigid-body motion,” in *12th IFToMM World Congress*, 2007.
  - [211] T. R. Williams and K. R. Fyfe, “Planar accelerometer configurations,” *Journal of Applied Mechanics*, vol. 71, no. 1, pp. 10–14, 2004.
  - [212] B. Zappa, G. Legnani, A. J. van den Bogert, and R. Adamini, “On the number and placement of accelerometers for angular velocity and acceleration determination,” *Journal of Dynamic Systems, Measurement, and Control*, vol. 123, no. 3, pp. 552–554, 2001.
  - [213] G. Baselli, G. Legnani, P. Franco, F. Brognoli, A. Marras, F. Quaranta, and B. Zappa, “Assessment of inertial and gravitational inputs to the vestibular system,” *Journal of Biomechanics*, vol. 34, no. 6, pp. 821 – 826, 2001.
  - [214] P. Spevak and P. Forstner, “Msp430 32-khz crystal oscillators,” Tech. Rep. SLAA322B, Texas Instruments Incorporated, August 2006.
  - [215] Micro Crystal Switzerland, Micro Crystal AG, Mühlestrasse 14, CH-2540 Grenchen, Switzerland, *Micro Crystal Switzerland - CC7V-T1A Tuning Fork Crystal 32.768 kHz*, version 5 ed., 2009.
  - [216] K. Kudapali, *AN1155 - Run-Time Calibration of Watch Crystals*, 2008.
  - [217] S. Bible, *AN826 - Crystal Oscillator Basics and Crystal Selection for rfPICTM and PICmicro Devices*. Microchip Technology Inc., ds00826a ed., 2002.

## References

---

# Appendix A

## Non-gyro IMU planar simplification

### A.1 Planar simplification

If motion is constrained to a planar coordinate system so that translational motion occurs only in the  $x$  and  $y$  dimensions and rotational motion only around the  $z$  axis, each sensor is reduced to a dual-axis arrangement and Equations 4.3, 4.8 and 4.9 simplify to Equations A.1, A.2 and A.3 respectively. For planar systems the rotational velocity is denoted by  $\omega$ . The planar simplification gives a useful insight into the arrangement of redundant sensor constellations (see A.2).

$${}^i\boldsymbol{\alpha} = \begin{bmatrix} \alpha_x & \alpha_y \end{bmatrix}^T \quad (\text{A.1})$$

$$\mathbf{G}_i = \begin{bmatrix} 1 & 0 & -d_x & -d_y \\ 0 & 1 & -d_y & d_x \end{bmatrix} \quad (\text{A.2})$$

$$\mathbf{s} = \begin{bmatrix} \ddot{x} & \ddot{y} & \omega^2 & \dot{\omega} \end{bmatrix}^T \quad (\text{A.3})$$

### A.1.1 Angular velocity estimation

For the simplified planar system, the objective function is simply the difference between the squared estimated angular velocity term and the squared angular velocity term yielded by the state vector  $\mathbf{s}$ . The error originally defined in Equation 4.24 in the estimated angular velocity may be redefined as the scalar quantity of Equation A.4.

$$\tilde{e} = \text{sign}(\tilde{\omega}(\tilde{\omega}^2 - \omega^2)) \left| |\tilde{\omega}| - \sqrt{\omega^2} \right| \quad (\text{A.4})$$

## A.2 Optimal array configurations

For ideal accelerometers, all valid array configurations perform equally. In practice, errors due to noise and signal saturation in accelerometer measurements will mean that some array configurations perform better than others. We wish to evaluate the effect of individual array characteristics to understand what determines an optimal array. An accelerometer array may be characterised by the following:

1. Position of the kinematic origin relative to a given sensor constellation;
2. Orientation of the kinematic origin relative to a given sensor constellation;
3. Sensor constellation, i.e. a set of specific sensor locations; and
4. Volume occupied by a given the sensor constellation.

Qin *et al.* [87] use inspection of matrix singular values to determine the suitability of a sensor configuration and determined that an optimal design is represented by an isotropic matrix; that is, a matrix with a condition number of one where all singular values are identical and nonzero. This is achieved by a sensor constellation represented by the vertices of a Platonic solid. The analysis of other authors exist for planar [210, 211] and 3D [212] [213] accelerometer arrays.

### A.2.1 Array volume and kinematic origin

To analyse the effect of the kinematic origin location and array volume we will consider a simplified planar array of 2 dual-axis accelerometers represented by the measurements  $\alpha'_1$  to  $\alpha'_4$ , each incorporating an error ( $\delta_1$  to  $\delta_4$ ). The kinematic origin of the array is equidistant from each sensor and may be displaced by a distance  $p$  along the origin  $y$  axis (remaining equidistant from each sensor). The volume of the array is defined by the distance of separation,  $l$ . The system is described by the schematic shown in Figure A.1. Equation A.5 describes the forward kinematic solution.

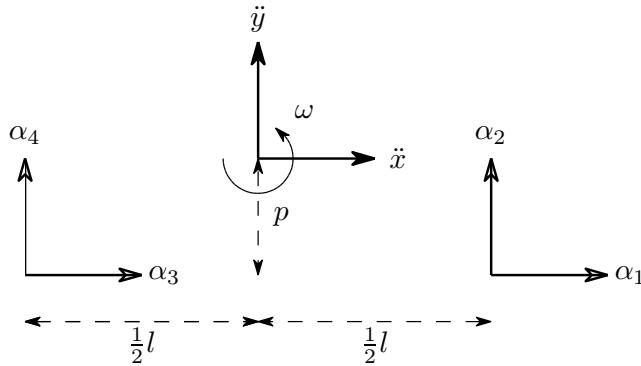


Figure A.1: Schematic describing the planar sensor array characterised by parameters  $p$  (origin displacement) and  $l$  (sensor separation)

$$\boldsymbol{\alpha}' = \mathbf{H}\mathbf{s} + \boldsymbol{\delta} \quad (\text{A.5})$$

$$\begin{bmatrix} \alpha'_1 \\ \alpha'_2 \\ \alpha'_3 \\ \alpha'_4 \end{bmatrix} = \begin{bmatrix} 1 & 0 & -\frac{l}{2} & p \\ 0 & 1 & p & \frac{l}{2} \\ 1 & 0 & \frac{l}{2} & p \\ 0 & 1 & p & -\frac{l}{2} \end{bmatrix} \begin{bmatrix} \ddot{x} \\ \ddot{y} \\ \omega^2 \\ \dot{\omega} \end{bmatrix} + \begin{bmatrix} \delta_1 \\ \delta_2 \\ \delta_3 \\ \delta_4 \end{bmatrix} \quad (\text{A.6})$$

The inverse kinematic solution may be arranged to provide the measured kinematic states of the body origin in terms of the true kinematic states and measurement errors as shown by Equations A.7 and A.8. In these equations,  $\ddot{x}'$ ,  $\ddot{y}'$ ,  $\omega'^2$  and  $\dot{\omega}'$

represent the measured kinematic states of the body origin.

$$\begin{aligned}
 \mathbf{s}' &= \mathbf{H}^{-1} \boldsymbol{\alpha}' \\
 &= \mathbf{H}^{-1}(\mathbf{H}\mathbf{s} + \boldsymbol{\delta}) \\
 &= \mathbf{s} + \mathbf{H}^{-1}\boldsymbol{\delta}
 \end{aligned} \tag{A.7}$$

$$\begin{bmatrix} \ddot{x}' \\ \ddot{y}' \\ \omega^{2'} \\ \dot{\omega}' \end{bmatrix} = \begin{bmatrix} \ddot{x} \\ \ddot{y} \\ \omega^2 \\ \dot{\omega} \end{bmatrix} + \begin{bmatrix} \frac{1}{2} & -\frac{p}{l} & \frac{1}{2} & \frac{p}{l} \\ \frac{p}{l} & \frac{1}{2} & -\frac{p}{l} & \frac{1}{2} \\ -\frac{1}{l} & 0 & \frac{1}{l} & 0 \\ 0 & +\frac{1}{l} & 0 & -\frac{1}{l} \end{bmatrix} \begin{bmatrix} \delta_1 \\ \delta_2 \\ \delta_3 \\ \delta_4 \end{bmatrix} \tag{A.8}$$

For this simplified equidistant configuration it can be seen from Equation A.8 how the distance from the origin ( $p$ ) and sensor separation ( $l$ ) may be manipulated to minimise the error in the measured kinematics states. The error in the measured angular states ( $\omega^{2'}$  and  $\dot{\omega}'$ ) is independent of the origin displacement and inversely proportional to the sensor separation. Error components in the measured linear states ( $\ddot{x}', \ddot{y}'$ ) are proportional to the origin displacement and inversely proportional to the sensor separation. This method extends to assessing redundant planar sensor arrays, and leads to a weighted average (linear accelerations) or a weighted sum of difference (angular acceleration). Thus it can be demonstrated that an optimal array would maximise the sensor separation. The kinematics origin should be located at the point where the expected mean acceleration is minimised. Where more specific information is not available this could be considered as the centre of mass of the limb. Other authors have arbitrarily chosen the centre of volume as the kinematic origin of a sensor array [86, 87], so assuming a homogeneous body the methodology we describe gives a justification for their decision.

Analysis of the equidistant planar structure supports Qin's assertion [87] that a reasonable sensor array would be to distribute the sensors on the vertices of a Platonic solid. This only applies if the body is assumed to be rotating around the

centre of the sensor constellation. Where more sensors than vertices are used in the IMU, a consideration of the singular values of the reconstruction matrix is an appropriate method to optimise sensor placement.



# Appendix B

## Crystal oscillator thermal response

### B.1 Introduction

An AHRS determines orientation through numerical integration of gyroscope measurements. The accuracy of the orientation is therefore dependent on both the calibrated performance of the gyroscope and the accuracy of the system clock. The system clock accuracy is also relevant to systems intended to achieve synchronisation of multiple devices. This appendix describes a characterisation of the crystal oscillator to determine its accuracy over an operating temperature range and considers how this error may be compensated for using existing hardware.

These investigations focusses specially on the 32.678 kHz crystal used by the x-IMU. This is the clock source for both the RTC and sample clock. Crystal oscillators typically have an error of around  $\pm 20$  ppm though this will increase with changes in temperature. Temperature Compensated Crystal Oscillator (TCXO) and Oven-Controlled Crystal Oscillator (OCXO) provide far greater accuracy over a temperature range but are large, expensive and require a lot of power. IMU

platforms typically already incorporate a thermometer to measure the temperature of the on-board electronics and so have the potential to achieve thermal calibration using only the existing hardware.

## B.2 Crystal error model

Equation B.1 [214] describes the crystal error model where  $e$  is the period error in ppm,  $e_0$  is the error at the turnover frequency,  $c$  is the temperature coefficient,  $T$  is the temperature of the crystal and  $T_0$  is the turnover temperature. Table B.1 summarises the values of each of these parameters are stated in the datasheet of the crystal used by the x-IMU [215].

$$e = e_0 + c(T - T_0)^2 \quad (\text{B.1})$$

Parameter	Typical value	Tolerance
$e_0$	0 ppm	$\pm 20$ ppm
$c$	0.035 ppm/ $^{\circ}\text{C}$ <sup>2</sup>	$\pm 10\%$
$T_0$	25 $^{\circ}\text{C}$	$\pm 5^{\circ}\text{C}$

Table B.1: Crystal error model parameters as specified in datasheet [215]

A typical error of  $\pm 20$  ppm at 25 $^{\circ}\text{C}$  can only be achieved if the design of the crystal oscillator circuit accounts for electrical characteristics of the PCB layout. For example, a poorly matched load capacitance can result in an error of almost 400 ppm [216]. Component values were initially selected according to manufacturer application notes [217] but were found to provide errors of the order of 150 ppm at 25 $^{\circ}\text{C}$ . Instead, component values were selected as those found to provide a error centred around 0 ppm 25 $^{\circ}\text{C}$  for the eight tested devices.

### B.3 Crystal characterisation

The thermal response of eight crystals were characterised using the thermal chapter described in Chapter 5. Each device was exposed to a monotonic ascent from -20°C to 70°C over 3 hours. A TTi TF930 frequency counter was used to measure the crystal frequency with an accuracy of  $\pm 0.2$  ppm [132]. Software was written to log the device temperature along side the crystal error. Figure B.1 shows measured period error for each device over the temperature range.

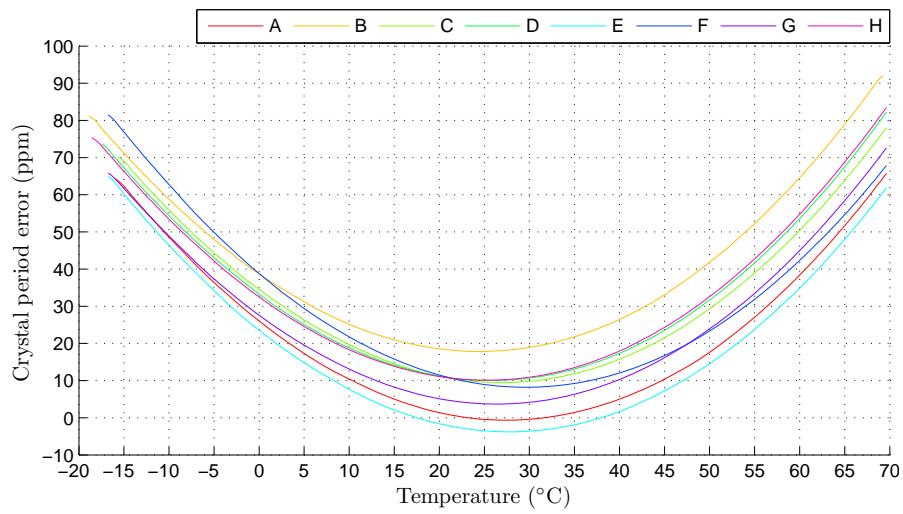


Figure B.1: Thermal response of eight crystal oscillators

A second-order polynomial was fit was performed for each device to determine the model parameters. The results are summarised in Table B.2.

Device	$e_0$ (ppm)	$c$ (ppm/°C <sup>2</sup> )	$T_0$ (°C)
A	-0.6	0.0360	27.07
B	17.8	0.0358	23.87
C	9.6	0.0357	26.20
D	10.0	0.0360	25.18
E	-3.7	0.0361	27.31
F	8.3	0.0357	29.11
G	3.7	0.0353	25.82
H	10.1	0.0359	24.75

Table B.2: Crystal error model parameters for eight devices over temperature range

## B.4 Crystal calibration

The thermal response of the crystal may be compensated for in calibration by determining the error for a given temperature and adjusting time calculations accordantly. The following sections present three different calibration solutions; each solution is able to offer a different accuracy over the operating temperature with different practical implications.

### B.4.1 Crystal calibration assuming typical parameters

The simplest calibration solution is to use the generic error model defined by Equation B.1 and the typical parameters specified in Table B.1. The resultant error of the eight devices is shown as Figure B.2.

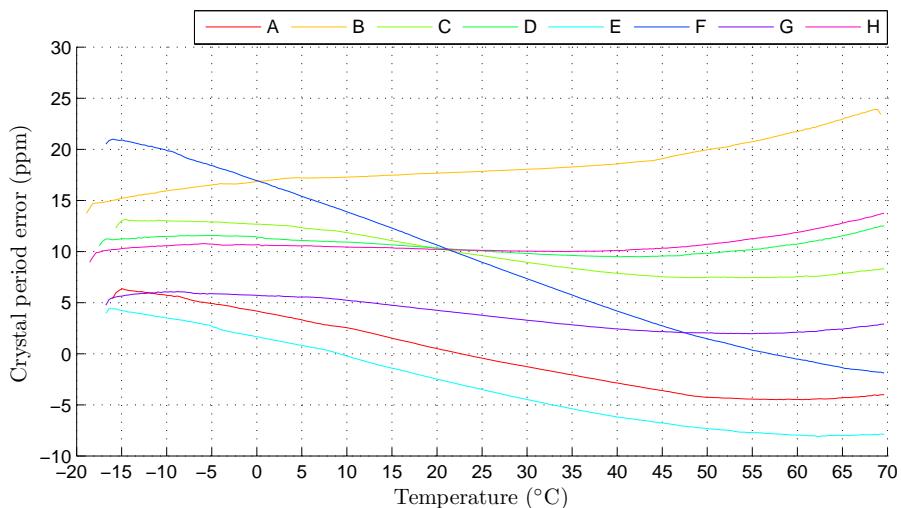


Figure B.2: Crystal oscillator error using the generic calibration model and typical parameters specified in the datasheet

This simple calibration solution is able to reduce the crystal error from 90 ppm to 25 ppm. This solution can be implemented in software and does not require any specific values to be evaluated through testing.

### B.4.2 Crystal calibration based on error model

A significant reduction in error can be achieved if the generic error model is used in conjunction with the specific specific parameters evaluated for each device and summarised in Table B.2. The resultant error of the eight devices is shown as Figure B.3. This calibration solution reduces the errors to within ppm.

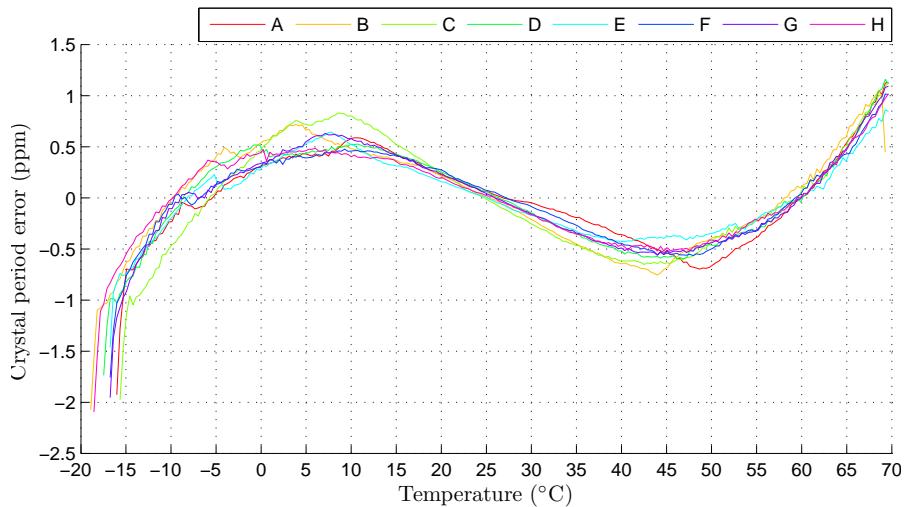


Figure B.3: Crystal oscillator error using the generic calibration model and specific parameters evaluated empirically

### B.4.3 Crystal calibration through third order model

The curve of the error shown in Figure B.3 indicates that the second-order generic error model is too low an order. This may due to an inaccuracy in the model form or may be due to a non-linearity in the thermometer. The source of the error is irreverent provide that a calibration solution accounts for the characteristics of both the thermometer and crystal. Figure B.4 shows the crystal oscillator error using a third-order calibration model.

This third-order calibration model is able to achieve an error of 0.2 ppm. Initial large error is thought to be because of an initial temperature difference between the thermometer device and the crystal oscillator.

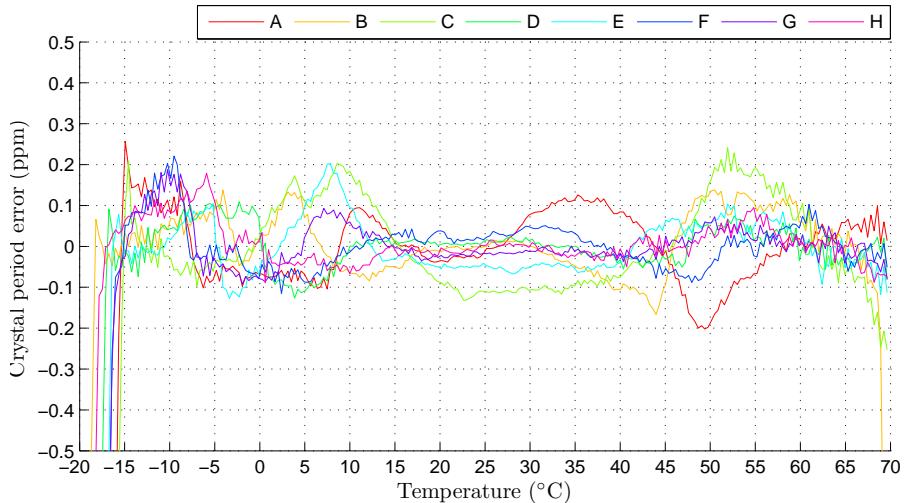


Figure B.4: Crystal oscillator error using a third-order calibration model

## B.5 Conclusion

The crystal characterisation indicates that without calibration crystal variation over temperature is equivalent to a  $\pm 0.009\%$  gyroscope sensitivity error over  $-20^{\circ}\text{C}$  to  $70^{\circ}\text{C}$ . This error would not be significant relative to other gyroscope error sources though it can be reduced then this will be of benefit. Reducing the crystal error may be of greater interest to achieving synchronisation between multiple devices. Three crystal calibration solutions were proposed; the first reduced the error to 25 ppm and can be implemented without any imposing any additional tasks on device production. Both the second and third calibration solution require a precise dataset to evaluate calibration parameters but yield an accuracy of up to 0.2 ppm.

# Appendix C

## Effect of geographic location on AHRS heading accuracy

The Earth's magnetic field is not constant around the world and it is of interest to consider how this will affect the accuracy of a heading measurement obtained from a magnetometer-based AHRS. The field varies in declination (deviation from geodetic north), inclination (deviation from the horizontal plane) and in intensity. Given that a magnetometer-based AHRS is intended to measure heading relative to magnetic north and not geodetic north, declination variations are irreverent. Maus *et al.* provide a comprehensive explanation and quantification of the Earth's magnetic field in *The US/UK World Magnetic Model for 2010-2015* [30]. It is reported that the intensity can be expected to between 0.22 Gauss and 0.67 Gauss around the world. The varying inclination is presented as the Mercator projection overlay shown as Figure C.1. The contours indicate that non-polar regions can expect a maximum inclination of  $\sim 80^\circ$ , with the  $0^\circ$  inclination contour remaining with  $\pm 15^\circ$  latitude of the equator. The inclination in the UK is indicated as being between  $66^\circ$  and  $70^\circ$ .

A mathematical model can be derived to investigate the effect of varying inclination

## Appendix C. Effect of geographic location on AHRS heading accuracy

---

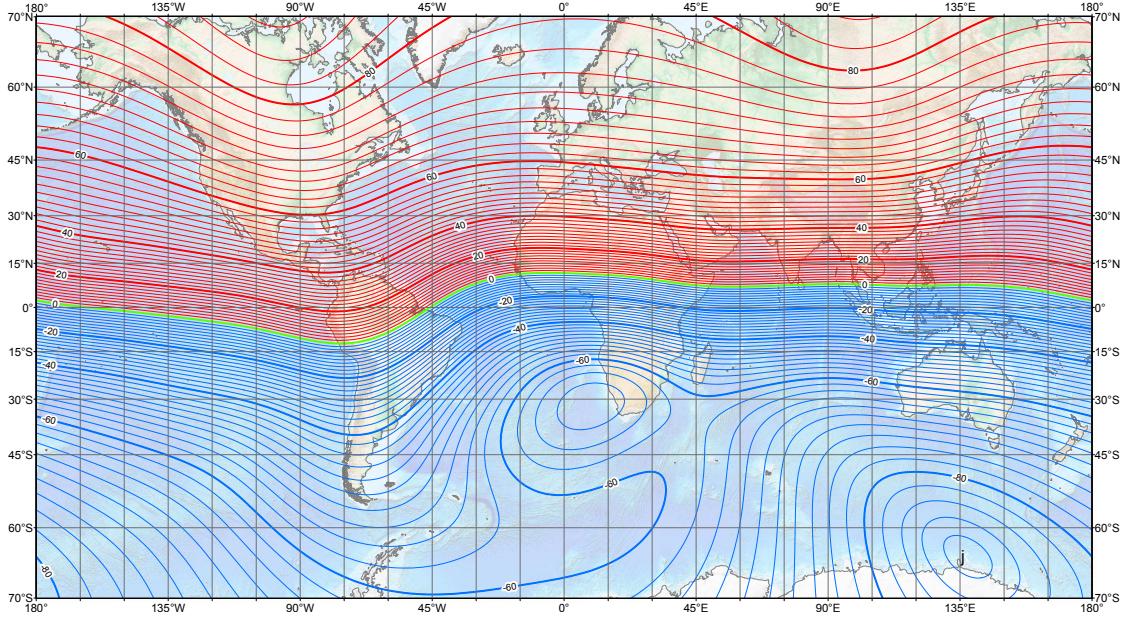


Figure C.1: Contour interval: 2 degrees, red contours positive (down); blue negative (up); green zero line. Image source: [30]

and field intensity on the measurement of heading obtained from a magnetometer. For this we will assume a two-axis magnetometer aligned with the Earth's surface so that the axis measurements,  $x$  and  $y$ , are equal to the horizontal components of the Earth's magnetic field. The magnetometer measurement,  $\mathbf{m}$ , can be defined as Equation C.1 where  $b$  is the magnetic field intensity,  $\gamma$  is the inclination and  $\theta$  is the heading of the device relative to magnetic north.

$$\mathbf{m} = \begin{bmatrix} x \\ y \end{bmatrix} = b \cos \gamma \begin{bmatrix} \sin \theta \\ \cos \theta \end{bmatrix} \quad (\text{C.1})$$

The heading as determined by the magnetometer measurements would be calculated as Equation C.2.

$$\theta = \text{atan2} \left( \frac{x}{\|\mathbf{m}\|}, \frac{y}{\|\mathbf{m}\|} \right) \quad (\text{C.2})$$

For an ideal sensor, the measurement of heading would be insensitive to field intensity and inclination provided  $b > 0$  and  $\gamma < 90$ . In practice, the magnetometer

will be subject to measurement errors from a variety of sources. For the purposes of this investigation we will consider an error of  $e$  in the magnetometer magnetometer  $x$  axis to yield the measurement  $\tilde{\mathbf{m}}$  as described by Equation C.3.

$$\tilde{\mathbf{m}} = \begin{bmatrix} \tilde{x} \\ \tilde{y} \end{bmatrix} = \mathbf{m} + \begin{bmatrix} e \\ 0 \end{bmatrix} \quad (\text{C.3})$$

The error in the corresponding measurement of heading may be calculated as Equation C.4.

$$\theta_{error} = \text{atan2}\left(\frac{\tilde{x}}{\|\tilde{\mathbf{m}}\|}, \frac{\tilde{y}}{\|\tilde{\mathbf{m}}\|}\right) - \theta \quad (\text{C.4})$$

Using Equations C.1, C.3 and C.4, it is possible to determine  $\theta_{error}$  for selected values of  $b$  and  $\gamma$ . The relationship of Equation C.1 suggests that  $\theta_{error}$  will have a greater sensitivity to variations in  $\gamma$  and so we will first consider a numerical example for only variations in  $\gamma$ . Fixed values of  $b = 0.5$  Gauss and  $e = 0.0044$  Gauss were chosen as this ratio was found to provided a maximum heading error  $\pm 0.5^\circ$  for an inclination angle of  $0^\circ$ ; this is heading accuracy specified by high-end commercial IMUs such as the MicroStrain 3DM-GX3-25 [108]. Figure C.2 shows the corresponding heading error achieved for an inclination of  $0^\circ$  (on the equator),  $30^\circ$ ,  $70^\circ$  (in the UK) and  $80^\circ$ .

The above results suggests a significant disparity in the accuracy of heading measurements dependent on the geographic location. However, this numerical example assumes a fixed magnetic field intensity. The actual intensity varies around the world and is typically greater away from the equator which can be expected to result in lower heading errors. To verify the accuracy of this simplified model,  $\theta_{error}$  was calculated for specific locations corresponding to the above values of inclination. The locations were selected to demonstrate the least disparity in the values of  $\theta_{error}$ . The results are summarised in Table C.1.

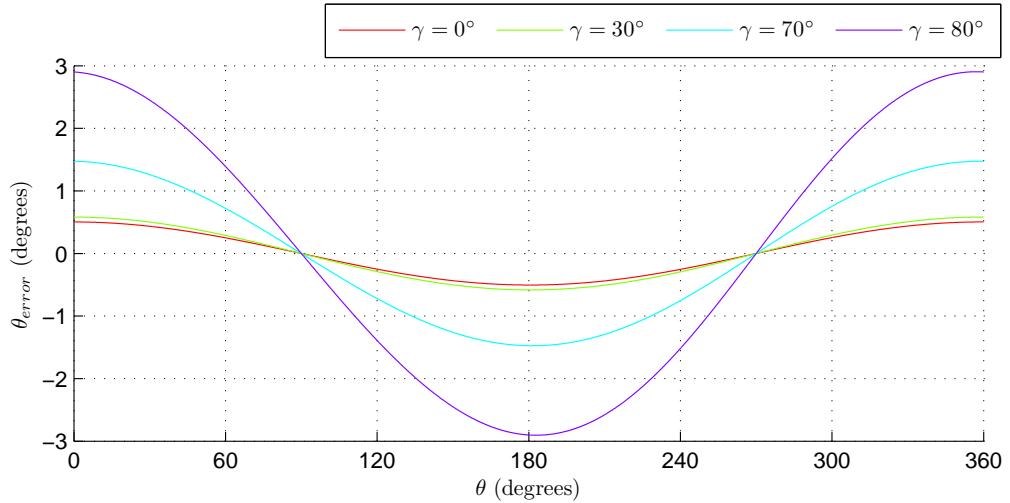


Figure C.2: Heading error achieved for an inclination of  $0^\circ$  (on the equator),  $30^\circ$ ,  $70^\circ$  (in the UK) and  $80^\circ$  and fixed values of  $b = 0.5$  Gauss and  $e = 0.0044$  Gauss

Location	$b$	$\gamma$	$\theta_{error}$
Phuket, Thailand	0.41 Gauss	$0^\circ$	$\pm 0.6^\circ$
Mina, Peru	0.26 Gauss	$0^\circ$	$\pm 1.0^\circ$
Edinburgh, UK	0.5 Gauss	$70^\circ$	$\pm 1.5^\circ$
Trua, Russia	0.6 Gauss	$80^\circ$	$\pm 2.5^\circ$

Table C.1: Extreme values of  $b$  and  $\gamma$  selected from [30] to demonstrate the least disparity in  $\theta_{error}$  for differing geographic locations

The results in Table C.1 confirm that the variation in heading performance for differing geographic locations may be less than that assumed by an inclination-only model but that the specific geographic location will still have a significant effect on the heading performance. A heading error performance observed at the equator may correspond to an error almost times greater in the UK and equally a heading accuracy observed in the UK may be expected to improve by a factor of three when operating on the equator.