



Departamentos
de Física e Matemática e de Engenharia Eletrotécnica

Ferramenta de Teste e Validação para Algoritmos de Fusão Sensorial

Dissertação apresentada para a obtenção do grau de Mestre em
Instrumentação Biomédica

Autor

Sérgio Miguel Cravo Patrão

Orientador

Professor Doutor José Pedro M. N. Amaro

Instituto Superior de Engenharia de Coimbra

Coimbra, Dezembro, 2015

AGRADECIMENTOS

Começo por endereçar os meus primeiros agradecimentos ao Professor Doutor José Pedro M. N. Amaro, orientador desta dissertação, pela amizade, partilha de conhecimento e confiança demonstrada nas minhas capacidades. Aqui fica um muito obrigado por todo o tempo dedicado a acompanhar a elaboração deste trabalho.

Aos meus pais, irmã e restantes familiares, por sempre terem acreditado em mim e por todas as palavras de conforto e incentivo que me ajudaram a chegar ao fim desta viagem. Por todas as ausências forçadas e telefonemas esquecidos, por todos os fins-de-semana escondidos atrás do computador, aqui fica um muito obrigado pela vossa compreensão.

A todos os meus amigos e colegas, que mesmo inconscientemente contribuíram para o sucesso de mais uma etapa da minha vida. Obrigado por todos os momentos de descontração e boa disposição, tão importantes para o meu equilíbrio emocional.

À minha namorada, Maria José, por tudo. Não há estrelas no céu que quantifiquem o quanto és importante para mim. Obrigado pelas infinitas correções, pela paciência e por seres uma fonte inesgotável de força de vontade que me contagia todos os dias. Por me motivares, por me fazeres sorrir e por me compreenderes. Muito obrigado.

Aos eternamente ausentes. Sei que estão orgulhosos.

RESUMO

O crescimento recente da utilização de sensores inerciais e da sua aplicabilidade em diversas áreas como saúde, desporto, sistemas de navegação e engenharias aeronáutica/aeroespacial é um fenómeno que poderá produzir impactos significativos na utilização desta tecnologia. Para além do baixo custo, do tamanho reduzido e baixo consumo de energia, os sensores inerciais possuem a capacidade de medir a intensidade de grandezas físicas de um objeto em movimento sem a necessidade de referências externas. Na área da saúde, nomeadamente em monitorização, este tipo de sensores tem sido alvo de investigação em situações de apoio à reabilitação física e monitorização da locomoção de doentes de Parkinson e Alzheimer. A eficácia destes sensores depende da capacidade dos algoritmos em reunir a informação de cada um deles e estimar informação mais precisa, mediante algoritmos de fusão sensorial. Este trabalho propõe a avaliação de vários algoritmos de fusão sensorial em termos de fiabilidade de resultados, utilizando um braço robótico para realizar movimentos pré-definidos e fornecer um valor de referência para a orientação dos sensores. Para o efeito foi realizado um estudo teórico sobre os métodos de representação da orientação de um objeto no espaço tridimensional, nomeadamente ângulos de Euler e quaterniões, e também um estudo sobre as características dos sensores acelerómetro, giroscópio e magnetómetro. É proposto o desenvolvimento de um dispositivo sem-fios para obtenção de resultados, que permite a comunicação com uma aplicação para Windows através do protocolo Bluetooth Low Energy.

Palavras-chave – fusão sensorial, acelerómetro, giroscópio, magnetómetro, ângulos de Euler, quaterniões.

ABSTRACT

Inertial sensors present a wide range of applications such as health, sports, navigation systems and aeronautical/aerospace engineering. Apart from the low cost, small size and low power consumption, the inertial sensors have the ability to measure the intensity of physical quantities of a moving object without the need for external references. In health monitoring context, they have been targets of research to support physical rehabilitation and monitoring gait patterns of Parkinson's and Alzheimer patients. The effectiveness of these sensors depends on the ability of a number of algorithms that gather data from each sensor and estimate movement parameters with high accuracy and reliability, a technique called sensor fusion. This work proposes the evaluation of several sensor fusion algorithms in terms of reliability, using a robotic arm to perform predefined movements and to provide a reference value for the orientation of the sensors. To this end, it's conducted a theoretical study on the methods of representation of the orientation of an object in three-dimensional space, namely Euler angles and quaternions, and also a study on the characteristics of the sensors (accelerometer, gyroscope and magnetometer). In order to gather results, hardware was developed to communicate via Bluetooth Low Energy Protocol with an application for Windows.

Keywords – sensor fusion, accelerometer, gyroscope, magnetometer, Euler angles, quaternions.

ÍNDICE

RESUMO	i
ABSTRACT	iii
ÍNDICE.....	v
ÍNDICE DE FIGURAS	vii
ÍNDICE DE QUADROS	ix
SIMBOLOGIA	xi
ABREVIATURAS	xvii
1 INTRODUÇÃO.....	1
1.1 OBJETIVOS E METODOLOGIA ADOTADA.....	2
1.2 ESTRUTURA DO DOCUMENTO	2
1.3 CONTRIBUIÇÕES	3
2 FUNDAMENTAÇÃO TEÓRICA	4
2.1 ÂNGULOS DE EULER.....	4
2.2 QUATERNIÕES	7
2.2.1 IGUALDADE E SOMA.....	8
2.2.2 MULTIPLICAÇÃO	8
2.2.3 NORMA	9
2.2.4 CONJUGADO.....	9
2.2.5 ROTAÇÕES COM QUATERNIÕES	9
2.2.6 CONVERSÃO DE ÂNGULOS DE EULER PARA QUATERNIÕES	10
2.2.7 CONVERSÃO DE QUATERNIÕES PARA MATRIZES DE ROTAÇÃO	11
2.2.8 CONVERSÃO DE QUATERNIÕES PARA ÂNGULOS DE EULER	11
2.3 ACELERÓMETRO.....	12
2.3.1 ESTIMAR ÂNGULOS DE ROTAÇÃO COM O ACELERÓMETRO	13
2.3.2 REMOÇÃO DA ACELERAÇÃO GRAVÍTICA	13
2.3.3 ESTIMAR VELOCIDADE E POSIÇÃO	14
2.4 GIROSCÓPIO	15
2.4.1 ESTIMAR ÂNGULOS DE ROTAÇÃO COM O GIROSCÓPIO.....	15
2.5 MAGNETÓMETRO	16
2.5.1 COMPENSAÇÃO DA INCLINAÇÃO	16
2.5.2 COMPENSAÇÃO DE INTERFERÊNCIAS MAGNÉTICAS.....	17

2.6	SINGULARIDADES NA UTILIZAÇÃO DOS ÂNGULOS DE EULER	19
3	FUSÃO SENSORIAL.....	22
3.1	FILTRO DE KALMAN ESTENDIDO.....	22
3.2	GRADIENTE DESCENDENTE.....	24
3.3	FILTRO DE KALMAN ESTENDIDO COM DUPLA FASE DE CORREÇÃO	27
3.4	FILTRO COMPLEMENTAR EXPLÍCITO	29
3.4.1	IMU	29
3.4.2	AHRS	30
3.5	GAUSS-NEWTON.....	31
3.5.1	FILTRO COMPLEMENTAR.....	33
3.5.2	FILTRO DE KALMAN.....	33
4	INFRAESTRUTURA PARA O TESTE E DESENVOLVIMENTO DE ALGORITMOS DE FUSÃO SENSORIAL	35
4.1	BLUETOOTH LOW ENERGY	35
4.2	SENSORTAG	36
4.3	MPU-9150.....	40
4.4	PROTÓTIPO DESENVOLVIDO	45
4.5	MÉTODO DE OBTENÇÃO DE RESULTADOS	49
5	ANÁLISE COMPARATIVA DO COMPORTAMENTO DOS ALGORITMOS DE FUSÃO SENSORIAL.....	51
5.1	AVALIAÇÃO RELATIVA À PRESTAÇÃO DOS ALGORITMOS NA DETERMINAÇÃO DA ORIENTAÇÃO	51
5.2	AVALIAÇÃO DA PRESTAÇÃO DO ALGORITMO DE REMOÇÃO DA GRAVIDADE.....	57
6	CONCLUSÃO	61
6.1	PRINCIPAIS CONCLUSÕES	61
6.2	DESENVOLVIMENTOS FUTUROS.....	65
	REFERÊNCIAS BIBLIOGRÁFICAS.....	66

ÍNDICE DE FIGURAS

Figura 2-1 – Referenciais inercial e do objeto alinhados na posição inicial. Imagem obtida em [14].4	4
Figura 2-2 – Referencial <i>Vehicle-1</i> . Os subscritos I e v1 denotam referencial inercial e <i>Vehicle-1</i> , respetivamente. Imagem obtida em [14].5	5
Figura 2-3 - Referencial <i>Vehicle-2</i> . Os subscritos v1 e v2 denotam referencial <i>Vehicle-1</i> e <i>Vehicle-2</i> , respetivamente. Imagem obtida em [14]6	6
Figura 2-4 - Referencial do objeto obtido após a última rotação - <i>roll</i> . Os subscritos B indicam referencial do objeto. Imagem obtida em [14].6	6
Figura 2-5 - Rotação com quatérniões.8	8
Figura 2-6 - Esquema simplificado de um acelerómetro e o seu circuito equivalente. Imagem obtida em [17]. 12	12
Figura 2-7 - Efeitos <i>Hard-Iron</i> e <i>Soft-Iron</i> nas medições de um magnetómetro. Imagem obtida em [25]. 18	18
Figura 2-8 – Representação em anéis de três rotações sucessivas. Imagem obtida em [27]. 19	19
Figura 2-9 – Alinhamento de dois eixos de rotação. Imagem obtida em [27]. 19	19
Figura 3-1 - Diagrama de implementação do Filtro de Kalman Estendido num IMU. 22	22
Figura 3-2 - Algoritmo do Filtro de Kalman Estendido. 24	24
Figura 3-3 – Diagrama de implementação do algoritmo Gradiente Descendente num sistema AHRS. 24	24
Figura 3-4 - Algoritmo do Gradiente Descendente. 27	27
Figura 3-5 - Algoritmo do Filtro de Kalman Estendido com dupla fase corretiva. 29	29
Figura 3-6 - Algoritmo do Filtro Complementar Explícito implementado num IMU. 30	30
Figura 3-7 - Algoritmo do Filtro Complementar Explícito implementado num sistema AHRS. 31	31
Figura 3-8 - Algoritmo Gauss-Newton..... 32	32
Figura 3-9 - Esquema de funcionamento do método de Gauss-Newton..... 33	33
Figura 3-10 – Algoritmo do filtro complementar para fusão de dados do método de Gauss-Newton. 33	33
Figura 3-11 – Filtro de Kalman para fusão de dados do método de Gauss-Newton. 34	34
Figura 4-1 – Camadas do protocolo de comunicação BLE. Imagem obtida em [33]. 35	35
Figura 4-2 – Características de um dispositivo BLE num modo de operação cliente/servidor. Imagem obtida em [33]. 36	36
Figura 4-3 - SensorTag da Texas Instruments. Imagem obtida em [34]. 36	36
Figura 4-4 - CC2540 USB Evaluation Module Kit. 37	37
Figura 4-5 - Organização da aplicação desenvolvida em Matlab. 38	38
Figura 4-6 - Sistema de eixos definido para o SensorTag. Imagem do SensorTag obtida em [38]. 39	39
Figura 4-7 - Ligações entre Arduino Leonardo e a placa de teste MPU-9150..... 41	41
Figura 4-8 - Descrição da orientação dos sensores na placa de teste e referencial escolhido para o MPU-9150. Imagem retirada de [42] 42	42
Figura 4-9 - Aplicação desenvolvida para receber dados da interface MPU-9150/Arduino. 44	44
Figura 4-10 - Protótipo desenvolvido..... 45	45
Figura 4-11 - Inclusão do serviço IMU no protocolo BLE. 46	46
Figura 4-12 - Inclusão do serviço DMP no protocolo BLE. 47	47
Figura 4-13 - Aplicação para receber dados do protótipo. 48	48
Figura 4-14 - Braço robótico da Motoman..... 50	50
Figura 5-1 - Variação do ângulo <i>roll</i> em comparação com a variação da posição do robot. 52	52
Figura 5-2 - Variação do ângulo <i>pitch</i> em comparação com a variação da posição do robot. 53	53
Figura 5-3 - Variação do ângulo <i>yaw</i> em comparação com a variação da posição do robot..... 54	54

Figura 5-4 - Convergência das estimativas do ângulo <i>roll</i> partindo de uma posição inicial diferente..	55
Figura 5-5 - Convergência das estimativas do ângulo <i>pitch</i> partindo de uma posição inicial diferente.	56
Figura 5-6 - Convergência das estimativas do ângulo <i>yaw</i> partindo de uma posição inicial diferente.	56
Figura 5-7 - Resposta dos algoritmos quando sujeitos a elevadas acelerações.....	57
Figura 5-8 - Na esquerda é representada a variação da aceleração no eixo Y. Na direita, variação da aceleração no eixo Z.	58
Figura 5-9 – Variação da posição angular dos sensores durante rotação em torno do eixo X.	58
Figura 5-10 - Na esquerda é representada a variação da aceleração no eixo X. Na direita, variação da aceleração no eixo Z.	58
Figura 5-11 - Variação da posição angular dos sensores durante rotação em torno do eixo Y.	59
Figura 5-12 – Na esquerda é representada a variação da posição angular dos sensores (roll a azul, pitch a verde e yaw a vermelho). Na direita é exibida a aceleração do eixo Y com gravidade (azul) e sem gravidade (verde).....	60

ÍNDICE DE QUADROS

Tabela 4-1 - Transformações necessárias aos dados do SensorTag.	40
Tabela 4-2 - Correspondências entre os pinos do MPU-9150 e entradas do Arduino Leonardo.	41
Tabela 4-3 - Transformações necessárias aos dados do MPU-9150.	42
Tabela 4-4 - Transformações de alinhamento dos dados do DMP com o referencial definido.	43
Tabela 5-1 - Desvio médio ao longo do movimento para o ângulo <i>roll</i>	52
Tabela 5-2 - Desvio médio ao longo do movimento para o ângulo <i>pitch</i>	53
Tabela 5-3 - Desvio médio ao longo do movimento para o ângulo <i>yaw</i>	54
Tabela 5-4 - Desvio na remoção as componentes de aceleração gravítica.	59
Tabela 6-1 - Síntese dos resultados para os testes de orientação.	62
Tabela 6-2 - Resultados para os testes de resposta dos algoritmos.	63
Tabela 6-3 - Levantamento do número de operações matemáticas necessárias à implementação dos algoritmos de fusão sensorial.	64

SIMBOLOGIA

θ – Rotação *pitch* em torno do eixo Y.

ψ – Rotação *yaw* em torno do eixo Z.

ϕ – Rotação *roll* em torno do eixo X.

$R(\psi)$ – Matriz de rotação em torno do eixo Z.

$R(\theta)$ – Matriz de rotação em torno do eixo Y.

$R(\phi)$ – Matriz de rotação em torno do eixo X.

R_I^B – Matriz de rotação do referencial inercial para o do objeto.

R_B^I – Matriz de rotação do referencial do objeto para o inercial.

Sen – Função trigonométrica Seno.

Cos – Função trigonométrica Cosseno.

Tan – Função trigonométrica Tangente.

i, j, k – Números imaginários.

q, p, r – Quaterniões.

q_0, p_0, r_0 – Parte escalar do quaternião q, p e r , respetivamente.

q_1, p_1, r_1 – Componente em X do quaternião q, p e r , respetivamente.

q_2, p_2, r_2 – Componente em Y do quaternião q, p e r , respetivamente.

q_3, p_3, r_3 – Componente em Z do quaternião q, p e r , respetivamente.

\mathbf{q} – Parte vetorial do quaternião q .

c – Escalar.

\otimes - Símbolo que denota a multiplicação entre quaterniões.

$\|q\|$ – Norma do quaternião q .

q^* - Conjugado do quaternião q .

w, v – Quaterniões puros.

\mathbf{w}, \mathbf{v} – Parte vetorial dos quaterniões puros w e v , respetivamente.

v_1, v_2, v_3 – Elementos de um vetor ou quaternião puro v .

q_z, q_y, q_x – Quaterniões que fazem a rotação em torno do eixo Z, Y e X, respetivamente.

a_m – Aceleração total.

a_x – Aceleração em X.

a_y – Aceleração em Y.

a_z – Aceleração em Z.

g – Aceleração da gravidade ($1g = 9,8 \text{ ms}^{-2}$) ou unidade de aceleração.

$a_{linear_inercial}$ – Vetor de aceleração linear no referencial inercial.

a_{linear_obj} – Vetor de aceleração linear no referencial do objeto.

$a_{gravítica}$ – Vetor de aceleração gravítica.

\vec{a} – Vetor de aceleração.

\vec{v} – Vetor de velocidade.

\vec{p} – Vetor de posição.

$\Delta t, dt$ – Intervalo de tempo.

\int – Integral.

$n_{ruído}$ – Ruído presente numa amostra.

$\dot{\theta}, \dot{\theta}_t$ – Valor de velocidade angular e valor de velocidade angular no instante t , respetivamente.

$\dot{\theta}_{total}$ – Valor de velocidade angular total.

θ_t, θ_{t-1} – Ângulo de rotação no instante t e $t-1$, respetivamente.

ω_{obj} – Vetor de velocidade angular no referencial do objeto.

ω_i – Vetor de velocidade angular no referencial inercial.

$\omega_{ix}, \omega_{iy}, \omega_{iz}$ – Valor de velocidade angular no referencial inercial no eixo X, Y e Z, respetivamente.

B – Magnitude do campo magnético terrestre.

α – Inclinação do campo magnético terrestre.

B_i – Vetor de campo magnético no referencial inercial.

B_{obj} – Vetor de campo magnético no referencial do objeto.

$B_{xobj}, B_{yobj}, B_{zobj}$ – Valor de campo magnético no referencial do objeto nos eixos X, Y e Z, respetivamente.

B_{tilt} – Vetor de campo magnético após a compensação da inclinação.

$B_{xtilt}, B_{ytilt}, B_{ztilt}$ – Valor de campo magnético após a compensação da inclinação nos eixos X, Y e Z, respetivamente.

$B_{xobj}, B_{yobj}, B_{zobj}$ – Valor de campo magnético no referencial do objeto nos eixos X, Y e Z, respetivamente.

B_{cal} – Vetor de campo magnético após a compensação de *Soft-Iron* e *Hard-Iron*.

$\bar{y}, \bar{z}, \bar{x}$ – Valor médio do respetivo eixo do magnetómetro.

$y_{max}, z_{max}, x_{max}$ – Valor máximo do respetivo eixo do magnetómetro.

$y_{min}, z_{min}, x_{min}$ – Valor mínimo do respetivo eixo do magnetómetro.

O – Vetor de offsets em cada eixo do magnetómetro.

S – Matriz de calibração para efeitos de *Soft-Iron*.

\bar{q}_t – Quaternião de estimativa *a priori* – Filtro de Kalman Estendido.

$\bar{q}_{t0}, \bar{q}_{t1}, \bar{q}_{t2}, \bar{q}_{t3}$ – Elementos do quaternião da estimativa *a priori*. Elemento escalar, X, Y e Z, respetivamente – Filtro de Kalman Estendido.

\bar{p}_t – Matriz de covariância do erro da estimativa *a priori* – Filtro de Kalman Estendido.

K – Ganho do filtro de Kalman – Filtro de Kalman Estendido.

q_t – Quaternião da estimativa *a posteriori* – Filtro de Kalman Estendido.

q_{t-1} – Quaternião da estimativa *a posteriori* na iteração anterior – Filtro de Kalman Estendido.

p_t – Matriz de covariância do erro da estimativa *a posteriori* – Filtro de Kalman Estendido.

p_{t-1} – Matriz de covariância do erro da estimativa *a posteriori* na iteração anterior – Filtro de Kalman Estendido.

Q – Matriz de covariância do ruído de transição de estado – Filtro de Kalman Estendido.

R – Matriz de covariância do ruído da medida de referência – Filtro de Kalman Estendido.

A – Matriz de transição de estado – Filtro de Kalman Estendido.

$h(\bar{q}_t)$ – Vetor expectável da medida de referência calculada com a estimativa *a priori* – Filtro de Kalman Estendido.

H – Matriz Jacobiano de derivadas parciais de $h(\bar{q}_t)$ – Filtro de Kalman Estendido.

V – Matriz Jacobiano de derivadas parciais do ruído da equação não linear $h(\bar{q}_t)$ – Filtro de Kalman Estendido.

Ω – Matriz que relaciona dois quaterniões em referenciais distintos – Filtro de Kalman Estendido.

I – Matriz identidade 4x4 – Filtro de Kalman Estendido.

z – Vetor das leituras do acelerómetro – Filtro de Kalman Estendido.

$\omega_x, \omega_y, \omega_z$ – Valor reportado pelo giroscópio no eixo X, Y e Z, respetivamente – Filtro de Kalman Estendido.

ω – Vetor de dados do giroscópio – Filtro de Kalman Estendido.

$\dot{q}_{\omega t}$ – Taxa de variação de um quaternião – Gradiente Descendente.

-
- \hat{q}_{t-1} – Quaterniño estimado na iteraço anterior – Gradiente Descendente.
- $q_{\omega t}$ – Quaterniño estimado com dados do giroscpio – Gradiente Descendente.
- \hat{q} – Quaterniño estimado na iteraço anterior – Gradiente Descendente.
- $\hat{q}_0, \hat{q}_1, \hat{q}_2, \hat{q}_3$ – Elementos do quaterniño estimado na iteraço anterior – Gradiente Descendente.
- J – Jacobiano.
- f – Funço objetivo
- \hat{d} – Campo de referncia – Gradiente Descendente.
- \hat{s} – Campo definido pelas leituras dos sensores – Gradiente Descendente.
- \hat{a} – Vetor de dados do acelermetro – Gradiente Descendente.
- \hat{m} – Vetor de dados do magnetmetro – Gradiente Descendente – ou previso do campo magntico – Filtro Complementar Explcito.
- m_t – Vetor de dados do magnetmetro no instante t – Gradiente Descendente.
- m_x, m_y, m_z – Elementos de m_t nos eixos X, Y e Z, respetivamente – Gradiente Descendente.
- \hat{h}_t – Vetor de dados do magnetmetro no instante t aps a compensaço da inclinaço – Gradiente Descendente.
- h_x, h_y, h_z – Elementos de \hat{h}_t em X, Y e Z, respetivamente – Gradiente Descendente.
- \hat{b}_t – Vetor de dados do magnetmetro no instante t , normalizado para ter componentes apenas nos eixos X e Z – Gradiente Descendente.
- b_x, b_z – Elementos de \hat{b}_t em X e Z, respetivamente – Gradiente Descendente.
- \hat{b} – Vetor de dados do magnetmetro aps compensaço – Gradiente Descendente.
- ∇ – Smbolo que indica mtodo do Gradiente Descendente.
- $q_{\nabla t}$ – Quaterniño estimado com mtodo do Gradiente Descendente.
- β – Parmetro de afinaço do mtodo do Gradiente Descendente.
- ω_{max} – Erro mximo do giroscpio em cada eixo – Gradiente Descendente.
- ∇f – Vetor que indica o erro da funço objetivo – Gradiente Descendente.
- $\|\nabla f\|$ – Norma de ∇f – Gradiente Descendente.
- \hat{q}_t – Quaterniño final do algoritmo Gradiente Descendente.
- \dot{q}_t – Taxa de variaço do quaterniño estimado com o giroscpio – Gradiente Descendente.
- q_{t1} – Estimativa *a posteriori* da primeira fase de correço – Filtro de Kalman Estendido com dupla fase de correço.
-

$q_{t1,0}, q_{t1,1}, q_{t1,2}, q_{t1,3}$ – Elementos de q_{t1} – Filtro de Kalman Estendido com dupla fase de correção.

q_1 – Estimativa final da primeira fase do Filtro de Kalman Estendido com dupla fase de correção.

q_{t2} – Estimativa *a posteriori* da segunda fase de correção – Filtro de Kalman Estendido com dupla fase de correção.

$q_{t2,0}, q_{t2,1}, q_{t2,2}, q_{t2,3}$ – Elementos de q_{t2} – Filtro de Kalman Estendido com dupla fase de correção.

$h_2(\bar{q}_t)$ – Vetor expectável da medida de referência calculada com a estimativa *a priori* (segunda fase de correção).

H_2 – Matriz Jacobiano de derivadas parciais de $h_2(\bar{q}_t)$ – Filtro de Kalman Estendido com dupla fase de correção.

δ – Fator de correção do Filtro Complementar Explícito.

Kp – Ganho proporcional.

Ki – Ganho integral.

e – Rotação relativa entre dois vetores.

\hat{v} – Previsão do campo gravítico.

\bar{v} – Vetor de dados de aceleração.

\bar{m} – Vetor de dados do campo magnético.

x_k – Estado num método de otimização de Gauss-Newton. O subscrito k indica o número de passos de otimização.

z_0 – Vetor de referência do campo magnético e gravitacional.

z_t – Vetor de dados de campo magnético e aceleração.

M – Matriz que faz rotação de dois vetores.

M_t – Matriz DCM equivalente a R_B^I .

\hat{q}_{obs} – Quaterniões de observação calculado com o método de Gauss-Newton.

q_k – Subscrito k no quaterniões indica os k passos de otimização do método Gauss-Newton.

\hat{q}_{gt} – Quaterniões estimado com dados do giroscópio – Gauss-Newton.

k – Parâmetro de afinação do Filtro Complementar.

Range – Corresponde ao tamanho do intervalo de valores que os sensores podem reportar.

x_i, y_i, z_i – Dados que chegam inicialmente de cada sensor.

x_f, y_f, z_f – Dados depois da transformação.

$q_{wi}, q_{xi}, q_{yi}, q_{zi}$ – Quaternião inicial.

$q_{wf}, q_{xf}, q_{yf}, q_{zf}$ – Quaternião final.

°/s – Unidade de medida de velocidade angular (graus por segundo).

μT – Unidade de medida de campo magnético (micro Tesla)

ABREVIATURAS

AHRS – *Attitude and Heading Reference System*

API – *Application Programming Interface*

BLE – *Bluetooth Low Energy*

DCM – *Direction Cosine Matrices*

DMP – *Digital Motion Processor*

GAP – *Generic Application Profile*

GATT – *Generic Attribute Profile*

GUI – *Graphical User Interface*

ICEUBI 2015 - *International Conference on Engineering – University of Beira Interior 2015*

IMU – *Inertial Measurement Unit*

I2C – *Inter-Integrated Circuit*

LL – *Link Layer*

MEMS – *Micro-Electro-Mechanical-System*

NED – *North-East-Down*

OSI – *Open Systems Interconnection*

PHY – *Physical Layer*

SIG – *Special Interest Group*

TI – *Texas Instruments*

1 INTRODUÇÃO

A Engenharia Biomédica é associada ao desenvolvimento tecnológico nas áreas da saúde, aplicando conceitos de engenharia às práticas médicas com o objetivo de estudar e desenvolver novos métodos de prestação de cuidados de saúde, para fins de diagnóstico, terapêutica ou acompanhamento de doentes. A monitorização de movimentos com recurso a dispositivos eletrónicos enquadra-se de forma particularmente relevante no desenvolvimento de trabalhos na área da Engenharia com aplicações médicas. A relevância destes estudos caracteriza-se pela aplicabilidade a outras áreas como o desporto ou os sistemas de navegação, e é atualmente uma área de investigação onde é possível identificar contribuições relevantes [1], [2]. O método mais utilizado para o cálculo da orientação de um objeto faz uso de tecnologias de reconhecimento de imagem recorrendo a câmaras de vídeo. Embora este método alcance bons resultados tem a desvantagem de se restringir a um ambiente controlado. Como método alternativo destaca-se o uso de sensores inerciais que têm a capacidade de detetar quantidades físicas de um objeto em movimento, independentemente de referências externas [1], [3] e [4]. Os mais recentes avanços neste tipo de sensores, com base em Microssistemas Eletromecânicos (MEMS – *Micro-Electro-Mechanical-System*), permitiram o desenvolvimento de sistemas de pequenas dimensões, leves, baratos e de baixo consumo, conduzindo a um aumento rápido no uso deste tipo de tecnologia [1], [5] e [6]. Estas tecnologias consubstanciam características indispensáveis nos dispositivos móveis como *smartphones* ou *tablets*. Os dados destes sensores são utilizados por exemplo para detetar a orientação do ecrã, melhorar a jogabilidade, efetuar comandos gestuais, capturar fotografias panorâmicas, fazer navegação pedestre ou em aplicações de realidade aumentada [7].

Na área da saúde, foram propostos nos últimos anos vários estudos sobre a utilização de sensores inerciais para avaliar doentes em reabilitação física [8], assim como doentes de Alzheimer [9] e de Parkinson [10], [11]. Neste âmbito os sensores são integrados em dispositivos portáteis que permitem a sua colocação em qualquer parte do corpo humano dependendo da informação que se pretende estudar. Estes dispositivos têm a capacidade de comunicação sem-fios com uma aplicação remota para conforto do sujeito de estudo. Destes dispositivos, destacam-se as Unidades de Medição Inercial (IMU - *Inertial Measurement Unit*) e os sistemas AHRS (*Attitude and Heading Reference System*). Na bibliografia existem duas definições distintas para caracterizar um IMU: uma que considera que este é composto por um acelerómetro, um giroscópio e um magnetómetro [1], [6] e outra que exclui a presença do magnetómetro por este não se tratar de um sensor inercial [2], [5]. Apesar de neste trabalho se incluir o magnetómetro na lista de sensores inerciais – apenas por uma questão de facilidade de referência aos três sensores estudados – é adotada a segunda definição por [2] ser um dos trabalhos mais citados na literatura. Por essa razão, define-se que os IMU são compostos por dois tipos de sensores triaxiais - acelerómetros e giroscópios - produzindo resultados com 6 graus de liberdade. Estes permitem descobrir a orientação de um objeto relativamente ao vetor da gravidade [2]. Os sistemas AHRS contribuem não só com a orientação do objeto em relação

ao vetor da gravidade, como também com respeito ao campo magnético terrestre [2]. Para o efeito, contam com um magnetómetro triaxial, produzindo resultados com 9 graus de liberdade [2]. A técnica de combinar os dados individuais de cada sensor para produzir nova e mais fiável informação é designada na literatura por fusão sensorial. A eficácia destes sistemas depende da capacidade dos algoritmos de fusão sensorial em processar e minorar os efeitos dos erros inerentes à utilização dos sensores inerciais [12], [13]. No âmbito do cálculo da orientação de um objeto no espaço tridimensional, o objetivo da fusão sensorial é a determinação dos ângulos de Euler ou quaterniões que representam a sua posição angular.

1.1 OBJETIVOS E METODOLOGIA ADOTADA

Esta dissertação tem como principal objetivo estudar algoritmos de fusão sensorial para aferir sobre a sua fiabilidade e capacidade de apresentar resultados sobre a orientação de um objeto no espaço tridimensional. Foi também feita uma comparação sobre a carga computacional que cada algoritmo impõe, fazendo um levantamento do número de operações matemáticas necessárias para a sua aplicação. Foram identificados na literatura trabalhos que utilizam dados de aceleração para estudar a locomoção de indivíduos em reabilitação física [8] ou com doença de Alzheimer [9] ou Parkinson [11]. Por essa razão foi também estudado um algoritmo de remoção das componentes de aceleração gravítica nas leituras de um acelerómetro. Com esses objetivos em mente, foram investigadas as vantagens e desvantagens dos sensores inerciais, os mecanismos de representação da orientação de um objeto no espaço tridimensional (ângulos de Euler e quaterniões) e diversos algoritmos de fusão sensorial: Filtro de Kalman Estendido, Filtro Complementar Explícito e os métodos do Gradiente Descendente e de Gauss-Newton. Esses conhecimentos foram aplicados a dispositivos disponíveis no mercado, nomeadamente o SensorTag da Texas Instruments e o MPU-9150 da Invensense, ambos dispositivos que permitem o acesso aos sensores inerciais que incorporam. O SensorTag é um dispositivo Bluetooth Low Energy (BLE) para o qual foi desenvolvida uma aplicação em Matlab, bem como uma ferramenta em C# compatível com o protocolo de comunicação BLE. O MPU-9150 faz parte de uma placa de teste fornecida pela Sparkfun Electronics que comunica com uma aplicação para o sistema operativo Windows por intermédio de uma interface com uma ferramenta de desenvolvimento Arduino. Foi ainda desenvolvida uma solução que permite ao MPU-9150 comunicar por BLE com uma aplicação em ambiente Windows. Foi com este *hardware* que foram realizados os testes de fiabilidade dos algoritmos de fusão sensorial, utilizando um braço robótico da família MOTOMAN NX100 da Yaskawa que permite posicionar o dispositivo proposto em diversas orientações pré-definidas, que servem de referência para os resultados obtidos.

1.2 ESTRUTURA DO DOCUMENTO

Esta dissertação é estruturada da seguinte forma: no Capítulo 2 são introduzidos os ângulos de Euler e quaterniões, e descritos os princípios de funcionamento dos sensores inerciais. É identificada a informação que pode ser extraída destes sensores, bem como identificadas as suas

vantagens e desvantagens. No Capítulo 3 são descritos os algoritmos de fusão sensorial mais relevantes identificados na literatura no decorrer deste trabalho. No Capítulo 4 são apresentados os dispositivos estudados, o método para efetuar a aquisição de dados bem como as aplicações desenvolvidas. No Capítulo 5 são apresentados os resultados obtidos. As conclusões e o trabalho futuro são descritos no Capítulo 6.

1.3 CONTRIBUIÇÕES

Na sequência do trabalho desenvolvido no âmbito desta dissertação foi submetido um artigo à *International Conference on Engineering – University of Beira Interior 2015 (ICEUBI 2015)* com o título “*A Framework for test and verification of inertial sensors and positioning algorithms*”. O artigo proposto apresenta uma ferramenta de teste e verificação para algoritmos de cálculo de orientação espacial com recurso a sensores inerciais, com o objetivo de desenvolver um método de avaliação de um algoritmo de remoção das componentes de aceleração gravítica nas leituras de um acelerómetro. O artigo foi aceite e apresentado no dia 3 de Dezembro de 2015.

2 FUNDAMENTAÇÃO TEÓRICA

Neste capítulo são definidas um conjunto de convenções adotadas ao longo deste documento. São apresentados os ângulos de Euler [14], quatérniões [15] e descrito o mecanismo utilizado para definir a orientação de um corpo rígido no espaço tridimensional. É igualmente descrito o funcionamento dos sensores utilizados – acelerómetro, giroscópio e magnetómetro – bem como o processamento efetuado sobre os dados recolhidos. É ainda justificada a utilização de quatérniões nos algoritmos de fusão sensorial que são estudados no Capítulo 3.

2.1 ÂNGULOS DE EULER

Para ser estimada a orientação de um corpo rígido no espaço tridimensional são definidos dois sistemas de coordenadas distintos: um centrado no objeto e outro na referência. O primeiro refere-se ao sistema de eixos que é fixo ao corpo do objeto e que acompanha todos os seus movimentos. O último é um sistema de coordenadas de referência, fixo, denominado referencial inercial. O referencial inercial é constituído por dois eixos ortogonais num plano tangente à superfície da Terra, e um eixo com direção geocêntrica. O referencial adotado neste trabalho segue a convenção NED – *North-East-Down* – comumente utilizada em aplicações aeroespaciais, em que o eixo X aponta para Norte, o eixo Y aponta para Este e o Z aponta para baixo [14], [15]. Numa posição inicial, é considerado que o referencial do objeto e o inercial estão alinhados, como é ilustrado na Figura 2-1.

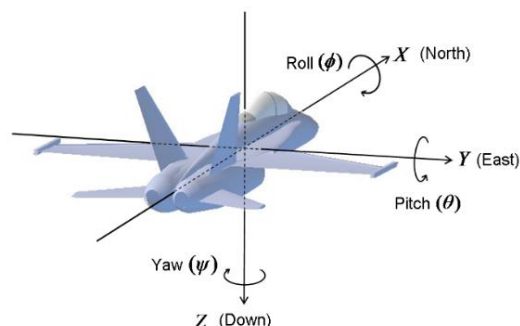


Figura 2-1 – Referenciais inercial e do objeto alinhados na posição inicial. Imagem obtida em [14].

Um ângulo de rotação em torno de determinado eixo é denominado por ângulo de Euler. Euler definiu que quaisquer dois sistemas de coordenadas independentes podem ser relacionados por uma sequência de rotações – não mais de três – em que duas rotações sucessivas não podem ser em torno do mesmo eixo [15]. Esta restrição determina doze (12) combinações possíveis de sequências de rotação[15]:

XYZ YZX ZXY

XZY YXZ ZYX

XYX YZY ZXZ

XZX YXY ZYZ

Estas sequências são lidas da esquerda para a direita, por exemplo, a sequência XYZ indica uma rotação em torno do eixo X, seguida de uma rotação em torno do eixo Y, seguida de uma rotação em torno do eixo Z. A sequência utilizada neste trabalho é a sequência aeroespacial - ZYX – por ser a sequência que foi identificada como sendo a mais utilizada na bibliografia. Esta sequência define a seguinte ordem de rotações: primeiro a rotação *yaw* (ψ) em torno do eixo Z, segundo a rotação *pitch* (θ) em torno de Y e finalmente *roll* (ϕ) em torno de X [15].

Os ângulos de Euler são o método mais intuitivo para representar a orientação relativa entre dois referenciais distintos. No entanto, é necessária a computação de matrizes de rotação para serem efetuados os cálculos da transformação dos vetores. Estas matrizes são denominadas de DCM – *Direction Cosine Matrices* – e implementam a transformação de vetores por meio de multiplicações sucessivas de matrizes [16]. É possível identificar uma matriz de rotação para cada ângulo de Euler, cuja função é executar a rotação de um vetor em torno de um único eixo. Como são necessárias três rotações, a definição do referencial inercial e do objeto não é suficiente. Por essa razão são definidos outros dois referenciais intermédios, denominados por *Vehicle-1* e *Vehicle-2* [14]. O ângulo de rotação *yaw* representa a rotação em torno do eixo Z do referencial inercial, transformando-o num novo sistema de coordenadas - *Vehicle-1* – em que o seu eixo Z é alinhado com o eixo Z do primeiro. Os eixos X e Y são desfasados de um ângulo ψ , tal como se ilustra na Figura 2-2 [14].

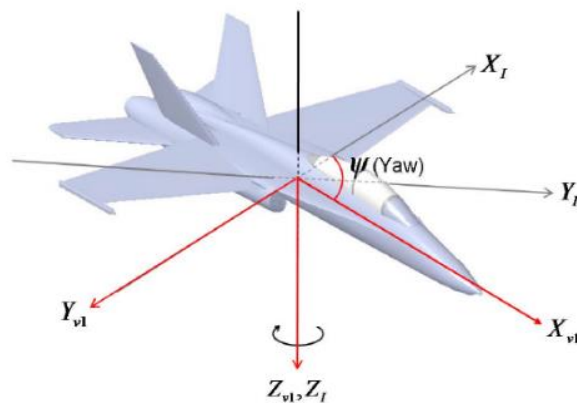


Figura 2-2 – Referencial *Vehicle-1*. Os subscritos I e v1 denotam referencial inercial e *Vehicle-1*, respetivamente. Imagem obtida em [14].

A descrição matemática de uma rotação de um vetor em R^3 por um ângulo *yaw*, é feita multiplicando a matriz $R(\psi)$ (2.1) por esse vetor. Em que *cos* é a função trigonométrica cosseno e *sen* a função seno.

$$R(\psi) = \begin{bmatrix} \cos(\psi) & \text{sen}(\psi) & 0 \\ -\text{sen}(\psi) & \cos(\psi) & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (2.1)$$

O ângulo de rotação *pitch* representa a rotação do sistema de coordenadas *Vehicle-1* em torno do eixo Y. Após essa rotação é definido o sistema de coordenadas *Vehicle-2*, cujo eixo Y é alinhado com o eixo Y anterior, enquanto os eixos X e Z estão desfasados de um ângulo θ , tal como ilustrado na Figura 2-3 [14].

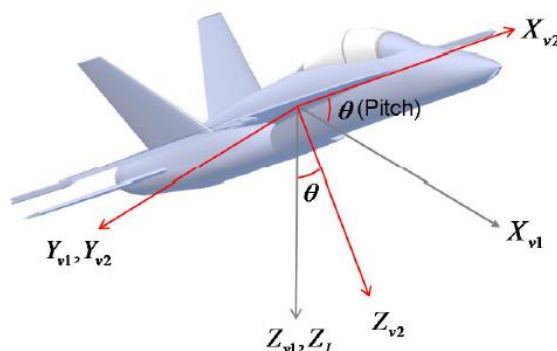


Figura 2-3 - Referencial *Vehicle-2*. Os subscritos v1 e v2 denotam referencial *Vehicle-1* e *Vehicle-2*, respectivamente. Imagem obtida em [14]

Similarmente, a rotação de um vetor por um ângulo *pitch* é feita multiplicando a matriz $R(\theta)$ (2.2) por esse vetor.

$$R(\theta) = \begin{bmatrix} \cos(\theta) & 0 & -\text{sen}(\theta) \\ 0 & 1 & 0 \\ \text{sen}(\theta) & 0 & \cos(\theta) \end{bmatrix} \quad (2.2)$$

Por fim, o ângulo de rotação *roll* representa a rotação do sistema de coordenadas *Vehicle-2* em torno do eixo X. Após esta rotação é obtido o referencial do objeto, em que os eixos X são coincidentes e os eixos Y e Z são desfasados de um ângulo ϕ , como se ilustra na Figura 2-4 [14].

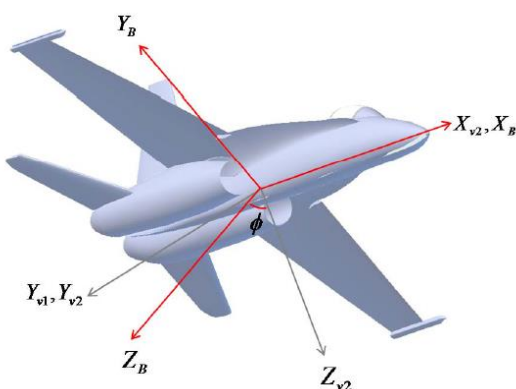


Figura 2-4 - Referencial do objeto obtido após a última rotação - *roll*. Os subscritos B indicam referencial do objeto. Imagem obtida em [14].

A rotação de um vetor por um ângulo *roll* é feita multiplicando a matriz $R(\phi)$ (2.3) por esse vetor.

$$R(\phi) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\phi) & \text{sen}(\phi) \\ 0 & -\text{sen}(\phi) & \cos(\phi) \end{bmatrix} \quad (2.3)$$

A rotação completa que indica a movimentação do referencial inercial até ao referencial do objeto é dada por (2.4) [14], [17]. De acordo com as propriedades da multiplicação de matrizes, o seu produto é associativo mas não comutativo, portanto a ordem de multiplicação das matrizes é importante. Deste modo, embora a rotação ψ aconteça em primeiro lugar, ela aparece em último na ordem das multiplicações.

$$R_I^B(\phi, \theta, \psi) = R(\phi) * R(\theta) * R(\psi) = \begin{bmatrix} \cos(\psi) \cos(\theta) & \text{sen}(\psi) \cos(\theta) & -\text{sen}(\theta) \\ \cos(\psi) \text{sen}(\theta) \text{sen}(\phi) - \text{sen}(\psi) \cos(\phi) & \text{sen}(\psi) \text{sen}(\theta) \text{sen}(\phi) + \cos(\psi) \cos(\phi) & \cos(\theta) \text{sen}(\phi) \\ \cos(\psi) \text{sen}(\theta) \cos(\phi) + \text{sen}(\psi) \text{sen}(\phi) & \text{sen}(\psi) \text{sen}(\theta) \cos(\phi) - \cos(\psi) \text{sen}(\phi) & \cos(\theta) \cos(\phi) \end{bmatrix} \quad (2.4)$$

A matriz R_I^B é uma matriz de rotação DCM que transforma o sistema de coordenadas de referência no referencial do objeto. Também é possível fazer o movimento inverso, ou seja, partir do referencial do objeto até ao inercial. Para o efeito, troca-se o sentido das rotações invertendo-se a ordem das multiplicações. No entanto, a matriz R_B^I é equivalente à inversa da matriz R_I^B , e como esta é uma matriz ortogonal, a resultante é a sua transposta (2.5) [14], [17].

$$R_B^I(\phi, \theta, \psi) = R(-\psi) * R(-\theta) * R(-\phi) = R_I^B{}^{-1} = R_I^B{}^T \begin{bmatrix} \cos(\psi) \cos(\theta) & \cos(\psi) \text{sen}(\theta) \text{sen}(\phi) - \text{sen}(\psi) \cos(\phi) & \cos(\psi) \text{sen}(\theta) \cos(\phi) + \text{sen}(\psi) \text{sen}(\phi) \\ \text{sen}(\psi) \cos(\theta) & \text{sen}(\psi) \text{sen}(\theta) \text{sen}(\phi) + \cos(\psi) \cos(\phi) & \text{sen}(\psi) \text{sen}(\theta) \cos(\phi) - \cos(\psi) \text{sen}(\phi) \\ -\text{sen}(\theta) & \cos(\theta) \text{sen}(\phi) & \cos(\theta) \cos(\phi) \end{bmatrix} \quad (2.5)$$

2.2 QUATERNIÕES

Os quaterniões foram propostos em 1843 por William Hamilton e são uma ferramenta alternativa aos ângulos de Euler para a representação e determinação da orientação de um corpo rígido em R^3 . Tal como o seu nome indica, estes não pertencem ao espaço tridimensional mas sim a R^4 e podem portanto ser representados pelo vetor de 4 elementos (2.6) [15].

$$q = q_0 + i * q_1 + j * q_2 + k * q_3 \quad (2.6)$$

Onde q_0, q_1, q_2 e q_3 são números reais e i, j e k números imaginários. Outra forma de representar um quaternião passa pela divisão na sua parte escalar (q_0) e na sua parte vetorial (\mathbf{q}) (2.7) [15].

$$q = q_0 + \mathbf{q} \quad (2.7)$$

$$\mathbf{q} = i * q_1 + j * q_2 + k * q_3$$

O uso do quaternião em cinemática deve-se ao facto deste descrever uma rotação θ (i.e., parte escalar), em torno de uma direção definida pelo vetor \mathbf{q} (i.e., parte vetorial), como se ilustra na Figura 2-5, sendo (2.8) uma forma comum de representação dos quaterniões. Este exige apenas

a definição dos referenciais inercial e do objeto, não precisando de intermédios, uma vez que apenas efetua uma única rotação para representar as três necessárias com os ângulos de Euler [15].

$$\begin{aligned}
 q_0 &= \cos\left(\frac{\theta}{2}\right) \\
 q_1 &= q_2 = q_3 = \text{sen}\left(\frac{\theta}{2}\right) \\
 q &= \cos\left(\frac{\theta}{2}\right) + i * \text{sen}\left(\frac{\theta}{2}\right) + j * \text{sen}\left(\frac{\theta}{2}\right) + k * \text{sen}\left(\frac{\theta}{2}\right)
 \end{aligned} \tag{2.8}$$

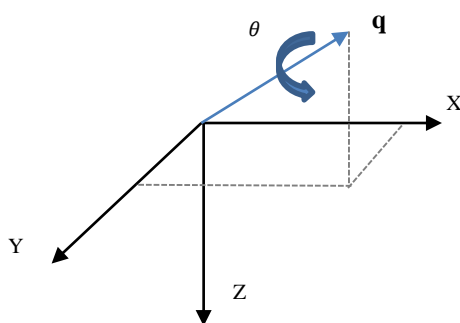


Figura 2-5 - Rotação com quatérniões.

Os quatérniões são utilizados com base numa álgebra própria, apresentada nas secções seguintes, que permite a computação da orientação de um objeto no espaço tridimensional.

2.2.1 IGUALDADE E SOMA

Dois quatérniões são iguais se todos os seus componentes forem iguais. Considerando dois quatérniões p e q , estes são iguais se as relações (2.9) se verificarem [15].

$$q_0 = p_0; q_1 = p_1; q_2 = p_2; q_3 = p_3 \tag{2.9}$$

Por sua vez, a soma de dois quatérniões é dada pela soma das componentes correspondentes (2.10) [15].

$$q + p = (q_0 + p_0) + i * (q_1 + p_1) + j * (q_2 + p_2) + k * (q_3 + p_3) \tag{2.10}$$

2.2.2 MULTIPLICAÇÃO

A multiplicação entre um quatérnião q e um número escalar c , é dada pela expressão (2.11) [15].

$$c * q = c * q_0 + i * c * q_1 + j * c * q_2 + k * c * q_3 \tag{2.11}$$

Para multiplicar dois quatérniões é necessário lembrar algumas propriedades dos números imaginários, entre as quais, a expressão que Hamilton apresentou quando definiu o conceito de quatérnião (2.12) [15].

$$\begin{aligned}
i^2 = j^2 = k^2 = ijk = -1 \\
ij = k = -ji \\
jk = i = -kj \\
ki = j = -ik
\end{aligned} \tag{2.12}$$

Estas propriedades são utilizadas para simplificar os cálculos entre as componentes vetoriais, sendo a multiplicação entre quaterniões determinada por (2.13), em que a resultante é também um quaterniões [15]. Esta operação é representada pelo símbolo \otimes .

$$\begin{aligned}
r = q \otimes p \Leftrightarrow \begin{bmatrix} r_0 \\ r_1 \\ r_2 \\ r_3 \end{bmatrix} &= \begin{bmatrix} p_0 & -p_1 & -p_2 & -p_3 \\ p_1 & p_0 & p_3 & -p_2 \\ p_2 & -p_3 & p_0 & p_1 \\ p_3 & p_2 & -p_1 & p_0 \end{bmatrix} * \begin{bmatrix} q_0 \\ q_1 \\ q_2 \\ q_3 \end{bmatrix} \Leftrightarrow \\
\begin{bmatrix} r_0 \\ r_1 \\ r_2 \\ r_3 \end{bmatrix} &= \begin{bmatrix} q_0 p_0 - q_1 p_1 - q_2 p_2 - q_3 p_3 \\ q_0 p_1 + q_1 p_0 + q_2 p_3 - q_3 p_2 \\ q_0 p_2 - q_1 p_3 + q_2 p_0 + q_3 p_1 \\ q_0 p_3 + q_1 p_2 - q_2 p_1 + q_3 p_0 \end{bmatrix}
\end{aligned} \tag{2.13}$$

2.2.3 NORMA

A norma de um quaterniões é encontrada da mesma forma que a de um vetor pertencente a R^3 (2.14) [15].

$$\|q\| = \sqrt{q_0^2 + q_1^2 + q_2^2 + q_3^2} \tag{2.14}$$

Os quaterniões utilizados para representar a orientação de um objeto no espaço tridimensional são denominados por quaterniões unitários ou normalizados, por possuírem norma igual a 1. A multiplicação de dois quaterniões unitários produz também um quaterniões unitário [15]. Para normalizar um quaterniões, este é dividido pela sua norma.

2.2.4 CONJUGADO

Para encontrar o conjugado de um quaterniões troca-se o sinal à sua parte vetorial (2.15). Sendo q^* o conjugado do quaterniões q [15].

$$\begin{aligned}
q &= q_0 + i * q_1 + j * q_2 + k * q_3 \\
q^* &= q_0 - i * q_1 - j * q_2 - k * q_3
\end{aligned} \tag{2.15}$$

2.2.5 ROTAÇÕES COM QUATERNIÕES

As coordenadas de um vetor v , no referencial inercial, podem ser transformadas num vetor w , no referencial do objeto, usando um quaterniões q . Matematicamente, essa transformação é descrita pela expressão (2.16) [15].

$$w = q^* \otimes v \otimes q \tag{2.16}$$

Em que w e v representam quaterniões puros, ou seja, contêm parte escalar nula (2.17). Logo, um vetor $v \in R^3$, indicando por exemplo as leituras de um acelerómetro triaxial ou um ponto no espaço, pode ser tratado como um quaterniões $v \in R^4$, contendo parte real nula [15].

$$\begin{aligned} w &= 0 + \mathbf{w} \\ v &= 0 + \mathbf{v} \end{aligned} \quad (2.17)$$

Resolvendo a equação e descartando a parte escalar, obtêm-se os seguintes resultados em formato matricial (2.18).

$$w = \begin{bmatrix} q_0^2 + q_1^2 - q_2^2 - q_3^2 & 2q_1q_2 + 2q_0q_3 & 2q_1q_3 - 2q_0q_2 \\ 2q_1q_2 - 2q_0q_3 & q_0^2 - q_1^2 + q_2^2 - q_3^2 & 2q_2q_3 + 2q_0q_1 \\ 2q_1q_3 + 2q_0q_2 & 2q_2q_3 - 2q_0q_1 & q_0^2 - q_1^2 - q_2^2 + q_3^2 \end{bmatrix} * \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix} \quad (2.18)$$

Esta matriz corresponde à matriz de rotação R_I^B , ilustrada em (2.4), sendo esta também uma matriz DCM. A simplificação desta matriz é estudada em [2], [15] devendo-se essa simplificação ao facto de se estar na presença de um quaternião unitário (2.19).

$$q_0^2 + q_1^2 + q_2^2 + q_3^2 = 1 \quad (2.19)$$

De (2.19) podem ser extraídas as seguintes relações (2.20).

$$\begin{aligned} q_0^2 + q_1^2 &= 1 - q_2^2 - q_3^2 \text{ ou } -q_2^2 - q_3^2 = q_0^2 + q_1^2 - 1 \\ q_0^2 + q_2^2 &= 1 - q_1^2 - q_3^2 \text{ ou } -q_1^2 - q_3^2 = q_0^2 + q_2^2 - 1 \\ q_0^2 + q_3^2 &= 1 - q_1^2 - q_2^2 \text{ ou } -q_1^2 - q_2^2 = q_0^2 + q_3^2 - 1 \end{aligned} \quad (2.20)$$

Simplificando a matriz R_I^B para uma das seguintes formas (2.21).

$$\begin{aligned} R_I^B &= \begin{bmatrix} 1 - 2q_2^2 - 2q_3^2 & 2q_1q_2 + 2q_0q_3 & 2q_1q_3 - 2q_0q_2 \\ 2q_1q_2 - 2q_0q_3 & 1 - 2q_1^2 - 2q_3^2 & 2q_2q_3 + 2q_0q_1 \\ 2q_1q_3 + 2q_0q_2 & 2q_2q_3 - 2q_0q_1 & 1 - 2q_1^2 - 2q_2^2 \end{bmatrix} \\ R_I^B &= \begin{bmatrix} 2q_0^2 + 2q_1^2 - 1 & 2q_1q_2 + 2q_0q_3 & 2q_1q_3 - 2q_0q_2 \\ 2q_1q_2 - 2q_0q_3 & 2q_0^2 + 2q_2^2 - 1 & 2q_2q_3 + 2q_0q_1 \\ 2q_1q_3 + 2q_0q_2 & 2q_2q_3 - 2q_0q_1 & 2q_0^2 + 2q_3^2 - 1 \end{bmatrix} \end{aligned} \quad (2.21)$$

2.2.6 CONVERSÃO DE ÂNGULOS DE EULER PARA QUATERNIÕES

Conhecendo os ângulos de Euler é possível encontrar o quaternião correspondente à matriz de rotação para uma determinada sequência [15]. Neste caso, são apresentadas as expressões para derivar o quaternião correspondente à sequência aeroespacial (2.22). É importante notar que, ao contrário das matrizes DCM, os quaterniões não exigem uma ordem de multiplicação inversa à ordem de rotação. Em que q_z , q_y e q_x representam os quaterniões que implementam a rotação ψ , θ e ϕ , respetivamente, r_0 a parte escalar do quaternião resultante r , r_1 a componente em X, r_2 a componente em Y e r_3 a componente em Z.

$$\begin{aligned}
q_z &= \cos\left(\frac{\psi}{2}\right) + i * 0 + j * 0 + k * \text{sen}\left(\frac{\psi}{2}\right) \\
q_y &= \cos\left(\frac{\theta}{2}\right) + i * 0 + j * \text{sen}\left(\frac{\theta}{2}\right) + k * 0 \\
q_x &= \cos\left(\frac{\phi}{2}\right) + i * \text{sen}\left(\frac{\phi}{2}\right) + j * 0 + k * 0 \\
r &= q_z \otimes q_y \otimes q_x \\
r_0 &= \cos\left(\frac{\psi}{2}\right) \cos\left(\frac{\theta}{2}\right) \cos\left(\frac{\phi}{2}\right) + \text{sen}\left(\frac{\psi}{2}\right) \text{sen}\left(\frac{\theta}{2}\right) \text{sen}\left(\frac{\phi}{2}\right) \\
r_1 &= \cos\left(\frac{\psi}{2}\right) \cos\left(\frac{\theta}{2}\right) \text{sen}\left(\frac{\phi}{2}\right) - \text{sen}\left(\frac{\psi}{2}\right) \text{sen}\left(\frac{\theta}{2}\right) \cos\left(\frac{\phi}{2}\right) \\
r_2 &= \cos\left(\frac{\psi}{2}\right) \text{sen}\left(\frac{\theta}{2}\right) \cos\left(\frac{\phi}{2}\right) + \text{sen}\left(\frac{\psi}{2}\right) \cos\left(\frac{\theta}{2}\right) \text{sen}\left(\frac{\phi}{2}\right) \\
r_3 &= \text{sen}\left(\frac{\psi}{2}\right) \cos\left(\frac{\theta}{2}\right) \cos\left(\frac{\phi}{2}\right) - \cos\left(\frac{\psi}{2}\right) \text{sen}\left(\frac{\theta}{2}\right) \text{sen}\left(\frac{\phi}{2}\right)
\end{aligned} \tag{2.22}$$

2.2.7 CONVERSÃO DE QUATERNIÕES PARA MATRIZES DE ROTAÇÃO

Como foi referido na secção 2.2.5, é possível determinar a matriz de rotação R_I^B para a sequência aeroespacial partindo do quaternião que a define (2.23) [15].

$$\begin{aligned}
R_I^B &= R(\phi) * R(\theta) * R(\psi) = \\
&\begin{bmatrix} q_0^2 + q_1^2 - q_2^2 - q_3^2 & 2q_1q_2 + 2q_0q_3 & 2q_1q_3 - 2q_0q_2 \\ 2q_1q_2 - 2q_0q_3 & q_0^2 - q_1^2 + q_2^2 - q_3^2 & 2q_2q_3 + 2q_0q_1 \\ 2q_1q_3 + 2q_0q_2 & 2q_2q_3 - 2q_0q_1 & q_0^2 - q_1^2 - q_2^2 + q_3^2 \end{bmatrix} * v \\
&= q^* \otimes v \otimes q
\end{aligned} \tag{2.23}$$

Esta matriz faz a transformação de um vetor no referencial inercial para o do objeto. Da mesma forma, é possível determinar a matriz de rotação que realiza a transformação inversa, ou seja, a matriz R_B^I (2.24). Esta é a matriz transposta de R_I^B , e corresponde igualmente, à transposta da matriz (2.23).

$$\begin{aligned}
R_B^I &= q \otimes v \otimes q^* = \\
&\begin{bmatrix} q_0^2 + q_1^2 - q_2^2 - q_3^2 & 2q_1q_2 - 2q_0q_3 & 2q_1q_3 + 2q_0q_2 \\ 2q_1q_2 + 2q_0q_3 & q_0^2 - q_1^2 + q_2^2 - q_3^2 & 2q_2q_3 - 2q_0q_1 \\ 2q_1q_3 - 2q_0q_2 & 2q_2q_3 + 2q_0q_1 & q_0^2 - q_1^2 - q_2^2 + q_3^2 \end{bmatrix} * v
\end{aligned} \tag{2.24}$$

2.2.8 CONVERSÃO DE QUATERNIÕES PARA ÂNGULOS DE EULER

Partindo de um quaternião q , são definidas as expressões (2.25) para derivar os ângulos de Euler, em que $\tan(\psi)$ é a tangente do ângulo *yaw*, $\tan(\phi)$ a tangente de *roll* e $\text{sen}(\theta)$ o seno de *pitch* [15].

$$\begin{aligned}\tan(\psi) &= \frac{2q_1q_2 + 2q_0q_3}{q_0^2 + q_1^2 - q_2^2 - q_3^2} \\ \text{sen}(\theta) &= -(2q_1q_3 - 2q_0q_2) \\ \tan(\phi) &= \frac{2q_2q_3 + 2q_0q_1}{q_0^2 - q_1^2 - q_2^2 + q_3^2}\end{aligned}\quad (2.25)$$

2.3 ACELERÓMETRO

O acelerómetro é um dispositivo que mede a aceleração de um objeto, sendo sensível às forças a que está a ser submetido, incluindo a gravidade. Recorrendo à Segunda Lei de Newton é possível transformar estas forças em acelerações [1], [18], [19] e [20]. Na sua forma mais simples, um acelerómetro com um único eixo pode ser representado como uma massa em suspensão. Qualquer tipo de aceleração provoca um deslocamento no material que é proporcional à aceleração sofrida, sendo este medido por transdutores (capacitivos, resistivos ou piezoelétricos), transformando-o num sinal elétrico. Na Figura 2-6 é apresentado um esquema simplificado em que o deslocamento do material altera a capacitância entre as placas de um condensador. Uma variação deste valor produz uma tensão que é digitalizada com um conversor analógico-digital [21].

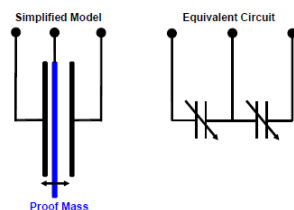


Figura 2-6 - Esquema simplificado de um acelerómetro e o seu circuito equivalente. Imagem obtida em [17].

A saída de um acelerómetro é descrita como a soma de dois tipos distintos de aceleração: aceleração linear e aceleração gravítica. A primeira refere-se à aceleração que é provocada pelo movimento do próprio acelerómetro, enquanto a segunda se deve à aceleração provocada pela força gravítica [5]. Considerando um acelerómetro triaxial que se encontre em repouso em cima de uma superfície perfeitamente horizontal, a sua saída é composta apenas por aceleração gravítica [17], medida no seu eixo vertical. Neste caso, a magnitude da aceleração total do objeto, a_m , é dada pela expressão (2.26).

$$a_m = \sqrt{a_x^2 + a_y^2 + a_z^2} = 1 g = 9,8 \text{ ms}^{-2} \quad (2.26)$$

Em que a_x corresponde à aceleração no eixo X do acelerómetro, a_y a aceleração no eixo Y e a_z a aceleração no eixo Z. O acelerómetro não mede diretamente a aceleração gravítica mas sim uma aceleração causada por uma força normal e oposta à da gravidade, que impede que este acelere em direção ao centro da Terra. Da mesma forma, se o acelerómetro se encontrar

em queda livre nenhuma aceleração é registada [20]. Este fenómeno permite o cálculo da orientação do dispositivo em relação ao vetor da força da gravidade, ou seja, apenas os ângulos *pitch* e *roll*, uma vez que qualquer rotação em torno do seu eixo vertical não altera o valor de aceleração gravítica sentida pelo acelerómetro [17], [18] e [19].

2.3.1 ESTIMAR ÂNGULOS DE ROTAÇÃO COM O ACELERÓMETRO

Os acelerómetros são insensíveis a qualquer tipo de rotação em torno do vetor da força da gravidade, não permitindo o cálculo do ângulo *yaw*. Para tal ser possível é necessária a adição de outros sensores como o giroscópio ou o magnetómetro [1], [17] e [19]. Multiplicando a matriz DCM R_I^B (2.4) pelas componentes de aceleração gravítica quando o acelerómetro está alinhado com o sistema de coordenadas de referência, são obtidos os valores dos ângulos de *pitch* e *roll* calculados por (2.27), em que a_x , a_y e a_z representam os valores de aceleração em X, Y e Z, respetivamente, e $g = 1g$ [14] [17].

$$R_I^B(\phi, \theta, \psi) * \begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix} = \begin{bmatrix} -\text{sen}(\theta) \\ \cos(\theta) \text{sen}(\phi) \\ \cos(\theta) \cos(\phi) \end{bmatrix} = \begin{bmatrix} a_x \\ a_y \\ a_z \end{bmatrix}$$

$$\tan(\phi) = \left(\frac{a_y}{a_z} \right) \quad (2.27)$$

$$\tan(\theta) = \frac{-a_x}{\sqrt{a_y^2 + a_z^2}}$$

Para eliminar duplicações de soluções é comumente estabelecido um intervalo para o ângulo *roll* de $[-\pi; \pi]$ e para o *pitch* $[-\pi/2; \pi/2]$ [14], [17].

2.3.2 REMOÇÃO DA ACELERAÇÃO GRAVÍTICA

Num acelerómetro, as componentes de aceleração gravítica presentes nas amostras de cada eixo são descritas como função da orientação do objeto. Em [22] é proposto um método para distinguir a aceleração gravítica da linear que pressupõe, numa primeira fase, saber os ângulos de rotação *roll*, *pitch* e *yaw* para fazer o objeto regressar virtualmente à posição horizontal e de seguida retirar a componente gravítica no eixo Z. Após a remoção é possível fazer regressar as mesmas amostras ao referencial do objeto, já sem as componentes de gravidade (2.28).

$$a_{linear_inercial} = R_B^I(\phi, \theta, \psi) * \begin{bmatrix} a_x \\ a_y \\ a_z \end{bmatrix} - \begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix} \quad (2.28)$$

$$a_{linear_obj} = R_I^B(\phi, \theta, \psi) * a_{linear_inercial}$$

Em que $a_{linear_inercial}$ é o vetor de aceleração linear no referencial inercial, a_{linear_obj} é o vetor de aceleração linear no referencial do objeto e g o valor da aceleração gravítica quando o acelerómetro se encontra alinhado com o sistema de referência $-1g$. Por fim, as mesmas equações são utilizadas para se obter apenas as componentes de aceleração gravítica (2.29) [22].

$$a_{gravítica} = \begin{bmatrix} a_x \\ a_y \\ a_z \end{bmatrix} - a_{linear_obj} \quad (2.29)$$

Quando são utilizados quaterniões, não é necessária a transformação em ângulos de Euler para implementar este método. Relembrando a expressão (2.24), que define a multiplicação entre quaterniões necessária para determinar a matriz R_B^I , assumindo a como um quaternião puro com parte vetorial correspondente ao vetor de aceleração lido pelo acelerómetro e q o quaternião que define a orientação do objeto nesse instante, o método anterior é implementado pelas equações (2.30). Descartando a parte escalar ao quaternião final, obtém-se o vetor de aceleração linear no referencial pretendido.

$$a = \begin{bmatrix} 0 \\ a_x \\ a_y \\ a_z \end{bmatrix} \quad (2.30)$$

$$a_{linear_inercial} = q \otimes a \otimes q^* - \begin{bmatrix} 0 \\ 0 \\ 0 \\ g \end{bmatrix}$$

$$a_{linear_obj} = q^* \otimes a_{linear_inercial} \otimes q$$

2.3.3 ESTIMAR VELOCIDADE E POSIÇÃO

Após a obtenção das acelerações lineares, podem ser estimadas velocidades e deslocamentos do objeto. Recordando os conceitos de aceleração, velocidade e posição, os dados de aceleração, \vec{a} , são integrados uma vez para a obtenção da velocidade, \vec{v} , e duas vezes para posição, \vec{p} (2.31). A utilização da dupla integração para o cálculo da posição leva a um acumular de erros devido ao ruído das amostras do acelerómetro ou de acelerações gravíticas que não são corretamente eliminadas com a técnica descrita na secção 2.3.2, no entanto este método é válido em situações em que a precisão do deslocamento calculado não é crítica, e pode ser melhorada considerando restrições da própria aplicação [21], [22].

$$\vec{v} = \int \vec{a} dt \quad (2.31)$$

$$\vec{p} = \int \vec{v} dt$$

É importante salientar que o cálculo da velocidade e posição é feito no referencial inercial e é necessária uma estimativa da orientação do sensor, não só para a remoção das acelerações gravíticas, mas também para o cálculo de acelerações lineares no sistema de coordenadas de referência [22]. Para esse efeito, deve ser usado o vetor de aceleração no referencial inercial, $a_{linear_inercial}$, calculado mediante (2.28) ou (2.30).

2.4 GIROSCÓPIO

O giroscópio é um dispositivo que mede velocidades angulares, ou seja, a velocidade com que é feita uma rotação em torno de determinado eixo. Quando é feita uma rotação em torno de um dos eixos do giroscópio, o efeito de Coriolis provoca uma vibração medida por transdutores capacitivos que a transforma num sinal elétrico, que por sua vez é digitalizado com um conversor analógico-digital [7]. O valor de velocidade angular fornecido pelo giroscópio pode ser modelado por (2.32), em que $\dot{\theta}_{total}$ é o valor fornecido pelo giroscópio em cada eixo, $n_{ruído}$ é o ruído da amostra e $\dot{\theta}$ é o valor real de velocidade angular [18], [19], [20] e [23].

$$\dot{\theta}_{total} = \dot{\theta} + n_{ruído} \quad (2.32)$$

2.4.1 ESTIMAR ÂNGULOS DE ROTAÇÃO COM O GIROSCÓPIO

A integração da velocidade angular permite obter o ângulo de rotação em torno de cada eixo do giroscópio, em que θ é o ângulo de rotação, $\dot{\theta}$ o valor de velocidade angular fornecido pelo giroscópio, Δt o intervalo de tempo e os subscritos t e $t-1$ a iteração atual e a passada, respetivamente (2.33) [1], [18], [19] e [23].

$$\theta_t = \theta_{t-1} + \dot{\theta}_t * \Delta t \quad (2.33)$$

Nesta integração pode observar-se um fenómeno caracterizado pela deriva do sinal para o infinito, que se deve ao facto do ruído presente nas amostras do giroscópio ser também integrado. À medida que o tempo de integração vai avançando, este fenómeno conduz a um acumular de erros e à perda de credibilidade do giroscópio na estimativa da orientação [1].

Os ângulos estimados por este método são ângulos de rotação relativos ao referencial do objeto e não são portanto ângulos de Euler. Relembrando os referenciais intermédios descritos na secção 2.1 - Referencial inercial, *Vehicle-1*, *Vehicle-2* e objeto - é necessário fazer uma transformação de cada valor reportado pelo giroscópio para o sistema de coordenadas correto para estes reportarem ângulos de Euler [14]. A expressão (2.34) define como os valores de velocidade angular lidos pelo giroscópio se relacionam com esses referenciais, em que ω_{obj} é o vetor de velocidade angular no referencial do objeto (medido pelo giroscópio); ω_{iX} , ω_{iY} e ω_{iZ} são os valores de velocidade angular no referencial inercial e $R(\phi)$ e $R(\theta)$ as matrizes DCM que implementam as rotações *roll* e *pitch*, respetivamente.

$$\omega_{obj} = R(\phi)R(\theta) \begin{bmatrix} 0 \\ 0 \\ \omega_{iZ} \end{bmatrix} + R(\phi) \begin{bmatrix} 0 \\ \omega_{iY} \\ 0 \end{bmatrix} + \begin{bmatrix} \omega_{iX} \\ 0 \\ 0 \end{bmatrix} \quad (2.34)$$

Esta expressão indica que, para a sequência aeroespacial, a rotação *roll* acontece em torno do eixo X do referencial *Vehicle-2* que é coincidente com o eixo X do referencial do objeto. Então, nenhuma transformação é necessária ao valor de velocidade angular nesse eixo. A rotação *pitch* acontece em torno do eixo Y do *Vehicle-1* e, depois dessa rotação ainda se verifica o *roll*. Por essa razão, o valor de velocidade angular em Y necessita de ser transformado mediante a multiplicação pela matriz $R(\phi)$. De forma semelhante, o valor de velocidade angular em Z

necessita de ser transformado pela resultante da multiplicação das matrizes $R(\phi)$ e $R(\theta)$. No entanto, é necessária a relação inversa, ou seja, saber ω_i em função de ω_{obj} (2.35) [24].

$$\omega_i = \begin{bmatrix} 1 & 0 & -\text{sen}(\theta) \\ 0 & \text{cos}(\phi) & \text{sen}(\phi)\text{cos}(\theta) \\ 0 & -\text{sen}(\phi) & \text{cos}(\phi)\text{cos}(\theta) \end{bmatrix}^{-1} * \omega_{obj} \quad (2.35)$$

Deste modo, a transformação necessária aos dados do giroscópio para que este reporte corretamente ângulos de Euler é dada pela matriz (2.36) [14].

$$\omega_i = \begin{bmatrix} 1 & \text{sen}(\phi)\text{tan}(\theta) & \text{cos}(\phi)\text{tan}(\theta) \\ 0 & \text{cos}(\phi) & -\text{sen}(\phi) \\ 0 & \frac{\text{sen}(\phi)}{\text{cos}(\theta)} & \frac{\text{cos}(\phi)}{\text{cos}(\theta)} \end{bmatrix} * \omega_{obj} \quad (2.36)$$

Em que ϕ e θ são os ângulos de Euler *pitch* e *roll* calculados no instante de tempo anterior.

2.5 MAGNETÓMETRO

O magnetómetro é um dispositivo sensível às diferentes componentes do campo magnético que atravessam os seus eixos. O seu princípio de funcionamento baseia-se no efeito de Hall, ou seja, sob a influência de um campo magnético produz uma tensão na sua saída que pode ser amplificada e digitalizada. Um magnetómetro triaxial paralelo à superfície terrestre permite o cálculo da orientação do próprio dispositivo em relação ao norte magnético – ângulo *yaw*. Na aplicação que se pretende usar o magnetómetro, e na grande maioria das aplicações de sensores inerciais, estes encontram-se anexados ao objeto em estudo e saem fora do plano horizontal à medida que este se movimenta – inclinação. Além disso, outros campos magnéticos, que não o terrestre, influenciam as medições do magnetómetro, sendo a influência destes denominada de *Hard-Iron* ou *Soft-Iron*, dependendo da sua origem. Para um cálculo correto do ângulo *yaw* é necessária a calibração do magnetómetro para reduzir os três efeitos descritos [25], [26].

2.5.1 COMPENSAÇÃO DA INCLINAÇÃO

Qualquer orientação do sensor pode ser modelada como a resultante das rotações *yaw*, *pitch* e *roll* aplicadas a uma posição inicial, em que o referencial do objeto se encontra alinhado com o referencial inercial. No caso particular do magnetómetro, esta condição inicial é dada pela expressão (2.37), em que B_i é o campo magnético de referência, B a magnitude do campo magnético terrestre e α a sua inclinação [26].

$$B_i = \begin{bmatrix} B\text{cos}(\alpha) \\ 0 \\ B\text{sen}(\alpha) \end{bmatrix} \quad (2.37)$$

As leituras do magnetómetro, B_{obj} , são dadas por (2.38).

$$B_{obj} = R(\phi)R(\theta)R(\psi)B_i \quad (2.38)$$

O objetivo da compensação da inclinação é fazer regressar o magnetómetro ao plano horizontal, situação em que os ângulos *pitch* e *roll* são nulos, usando esses ângulos estimados com o auxílio do acelerómetro e do giroscópio [1], [26]. Da expressão (2.38) retira-se a relação entre o cálculo do ângulo *yaw* do magnetómetro com os valores deste já calibrados para a inclinação (2.39).

$$R(\theta)^{-1}R(\phi)^{-1}B_{obj} = R(\psi)B_i \quad (2.39)$$

Então, sendo B_{tilt} os dados após a compensação, estes dados podem ser calculados mediante a equação (2.40).

$$B_{tilt} = R(\theta)^{-1}R(\phi)^{-1}B_{obj}$$

$$B_{tilt} = \begin{bmatrix} \cos(\theta) & \text{sen}(\theta)\text{sen}(\phi) & \text{sen}(\theta)\cos(\phi) \\ 0 & \cos(\phi) & -\text{sen}(\phi) \\ -\text{sen}(\theta) & \cos(\theta)\text{sen}(\phi) & \cos(\theta)\cos(\phi) \end{bmatrix} * \begin{bmatrix} Bx_{obj} \\ By_{obj} \\ Bz_{obj} \end{bmatrix} \quad (2.40)$$

Em que Bx_{obj} é a leitura no eixo X do magnetómetro, By_{obj} a leitura no eixo Y, e Bz_{obj} a leitura no eixo Z. É importante referir que não é necessário o conhecimento do valor da magnitude do campo magnético terrestre ou a sua inclinação [26]. Apesar destas variáveis irem variando ao longo da superfície terrestre, estas anulam-se ao longo dos cálculos, permitindo o cálculo de *yaw* (2.41).

$$\begin{bmatrix} Bx_{tilt} \\ By_{tilt} \\ Bz_{tilt} \end{bmatrix} = R(\theta)^{-1}R(\phi)^{-1}B_{obj} = \begin{bmatrix} \cos(\psi) B\cos(\alpha) \\ -\text{sen}(\psi) B\cos(\alpha) \\ \text{sen}(\alpha) \end{bmatrix}$$

$$\frac{By_{tilt}}{Bx_{tilt}} = \frac{-\text{sen}(\psi) B\cos(\alpha)}{\cos(\psi) B\cos(\alpha)} \quad (2.41)$$

$$\tan(\psi) = \left(\frac{-By_{tilt}}{Bx_{tilt}} \right)$$

Em que Bx_{tilt} é a leitura no eixo X após a compensação, By_{tilt} no eixo Y e Bz_{tilt} no eixo Z. Da mesma forma, a compensação da inclinação do magnetómetro pode ser feita com quaterniões. Do quaternião estimado para esse instante de tempo, são calculados os ângulos *pitch* e *roll*. Com esses ângulos é construído um novo quaternião q , considerando o ângulo *yaw* nulo, sendo este que vai fazer a rotação do vetor de dados do magnetómetro para o plano horizontal, descartando a parte escalar (2.42).

$$\begin{bmatrix} Bx_{tilt} \\ By_{tilt} \\ Bz_{tilt} \end{bmatrix} = q \otimes B_{obj} \otimes q^* \quad (2.42)$$

2.5.2 COMPENSAÇÃO DE INTERFERÊNCIAS MAGNÉTICAS

Adquirindo um elevado número de amostras do magnetómetro, orientando-o no máximo de direções possíveis e construindo um gráfico tridimensional com as leituras dos três eixos, é possível obter uma esfera centrada na origem caso essas leituras se encontrassem livres de interferências. No entanto, na presença de efeitos de *Hard-Iron* a esfera apresenta um

deslocamento em relação à origem, enquanto na presença de efeitos de *Soft-Iron* a esfera é deformada numa elipsoide. Na Figura 2-7 são ilustrados esses dois efeitos, a vermelho encontra-se um conjunto de dados que define a superfície de uma elipsoide que tem o seu centro afastado da origem, enquanto a azul é visível um conjunto de amostras que define a superfície de uma esfera centrada na origem.

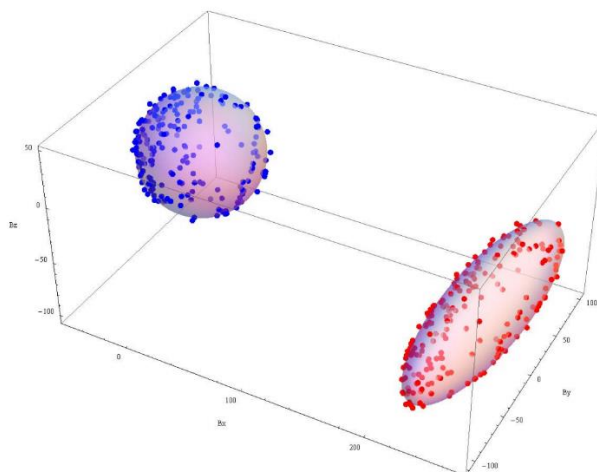


Figura 2-7 - Efeitos *Hard-Iron* e *Soft-Iron* nas medições de um magnetómetro. Imagem obtida em [25].

Os efeitos de *Hard-Iron* têm origem em materiais ferromagnéticos da própria placa de circuito impresso do magnetómetro, que se encontram permanentemente magnetizados. Como estes materiais se encontram no mesmo referencial que o magnetómetro - o referencial do objeto - estes contribuem com um efeito constante nas leituras de cada eixo do sensor [25]. A redução deste efeito baseia-se na determinação dessa constante. O método proposto pressupõe a aquisição de um conjunto de amostras no máximo de direções possíveis e a determinação do valor médio de cada eixo, que corresponde ao deslocamento ou desvio em relação à origem (2.43).

$$\begin{aligned}\bar{x} &= \frac{x_{max} + x_{min}}{2} \\ \bar{y} &= \frac{y_{max} + y_{min}}{2} \\ \bar{z} &= \frac{z_{max} + z_{min}}{2}\end{aligned}\tag{2.43}$$

Em que x_{max} , y_{max} e z_{max} são os valores máximos nos eixos X, Y e Z, respetivamente, x_{min} , y_{min} e z_{min} os mínimos nos eixos X, Y e Z, respetivamente, e \bar{x} , \bar{y} e \bar{z} os valores médios determinados para o eixo X, Y e Z, respetivamente. A diferença entre os efeitos *Soft-Iron* e *Hard-Iron* reside na dependência da orientação do sensor com o campo magnético e não pode portanto ser descrita como uma única constante. Estes efeitos são criados quando o campo

magnético terrestre induz um campo magnético temporário em componentes ferromagnéticos que normalmente não se encontram magnetizados. A solução final da calibração do magnetómetro é dada pela equação (2.44), em que S é denominada de matriz *Soft-Iron* e O o vetor de dados de desvio para cada eixo do magnetómetro. Neste trabalho, a matriz S foi assumida como sendo uma matriz identidade 3×3 , isto é, foi considerado que os efeitos de *Hard-Iron* são dominantes e os de *Soft-Iron* foram ignorados [25].

$$B_{cal} = R(\theta)^{-1}R(\phi)^{-1}S^{-1}(B_{obj} - O) \quad (2.44)$$

2.6 SINGULARIDADES NA UTILIZAÇÃO DOS ÂNGULOS DE EULER

Este capítulo tem o objetivo de demonstrar graficamente e matematicamente algumas das singularidades na utilização dos ângulos de Euler. Estes são momentos em que os ângulos de Euler não são capazes de definir unicamente a orientação de um objeto no espaço tridimensional. Uma situação inerente a qualquer sequência de rotações com ângulos de Euler ocorre quando dois eixos de rotação se alinham, o que leva a que as rotações em torno destes produzam o mesmo movimento [27].

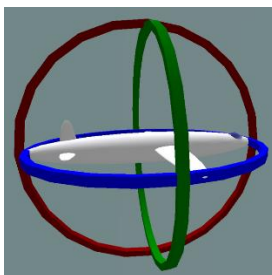


Figura 2-8 – Representação em anéis de três rotações sucessivas. Imagem obtida em [27].

Na Figura 2-8 considera-se o anel azul como o anel exterior, o vermelho o intermédio e o verde o interior. Considerando a sequência aeroespacial – ZYX – atribui-se o anel azul ao ângulo *yaw*, o vermelho ao ângulo *pitch* e o azul ao *roll*, de tal forma que uma rotação *yaw* roda os três anéis, uma rotação *pitch* roda o anel vermelho e verde e uma rotação *roll* apenas o verde. Deste modo, quando é efetuada uma rotação *pitch* de $\pm 90^\circ$, os anéis verde e azul ficam alinhados o que leva a que qualquer rotação em torno destes produza o mesmo movimento.

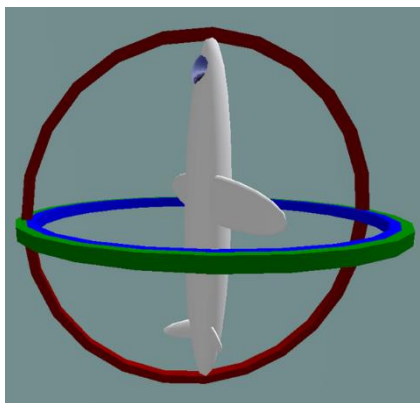


Figura 2-9 – Alinhamento de dois eixos de rotação. Imagem obtida em [27].

Na Figura 2-9 é perceptível que quando se pretende fazer uma rotação *yaw* ou *roll* numa situação em que *pitch* é igual a $\pm 90^\circ$, esta produz a mesma rotação. É importante referir que este fenómeno não desaparece ao trocar a ordem das rotações, este apenas acontece numa situação diferente. Isto é, considerando a rotação XZY, em que X representa o anel exterior, Z o intermédio e Y o interior, a singularidade ocorrerá quando *yaw* é igual a $\pm 90^\circ$.

Matematicamente, esta ocorrência é justificada de diversas formas. Por exemplo, durante o cálculo do *roll* com o acelerómetro, na situação em que o eixo X aponta na vertical (*pitch* igual a $\pm 90^\circ$), os valores de aceleração em Y e Z, a_y e a_z respetivamente, são zero. Mesmo que o eixo não aponte exatamente na vertical, o cálculo da inversa da tangente será dominado por ruído, tanto no numerador, como no denominador, produzindo uma estimativa do *roll* instável. Esta singularidade pode ser atenuada, modificando a equação do *roll* ao adicionar uma fração μ de a_z^2 no denominador, de forma a prevenir que este seja zero (2.45) [17].

$$\begin{aligned}\tan(\phi) &= \frac{a_y}{\text{sign}(a_z)\sqrt{a_z^2 + \mu a_x^2}} \\ \tan(\theta) &= \frac{-a_x}{\sqrt{a_y^2 + a_z^2}}\end{aligned}\quad (2.45)$$

Em que $\text{sign}(a_z)$ representa a recuperação do sinal de a_z pela utilização da raiz quadrada. No entanto, a singularidade continuará a acontecer mas mais próximo de $\pm 90^\circ$ [17].

Ao calcular os ângulos *roll*, *pitch* e *yaw* usando os dados do giroscópio, estes têm de ser convertidos para os sistemas de coordenadas corretos. Na matriz utilizada na transformação (2.46), são visíveis duas divisões por $\cos(\theta)$, portanto, quando o ângulo *pitch* se aproxima de $\pm 90^\circ$ o denominador fica zero, divergindo os elementos da matriz para o infinito, causando a falha da conversão.

$$\omega_i = \begin{bmatrix} 1 & \text{sen}(\phi)\tan(\theta) & \text{cos}(\phi)\tan(\theta) \\ 0 & \text{cos}(\phi) & -\text{sen}(\phi) \\ 0 & \frac{\text{sen}(\phi)}{\text{cos}(\theta)} & \frac{\text{cos}(\phi)}{\text{cos}(\theta)} \end{bmatrix} * \omega_{obj} \quad (2.46)$$

Este fenómeno também acontece durante a utilização das matrizes DCM. No caso de R_I^B apresentada em (2.47), substituindo θ por $\pm 90^\circ$, esta apenas determina uma única rotação dada por $(\phi \mp \psi)$, e não pode portanto ser usada para calcular os ângulos *roll* e *yaw*.

$$\begin{aligned}R_I^B &= \\ &= \begin{bmatrix} \text{cos}(\psi)\text{cos}(\theta) & \text{sen}(\psi)\text{cos}(\theta) & -\text{sen}(\theta) \\ \text{cos}(\psi)\text{sen}(\theta)\text{sen}(\phi) - \text{sen}(\psi)\text{cos}(\phi) & \text{sen}(\psi)\text{sen}(\theta)\text{sen}(\phi) + \text{cos}(\psi)\text{cos}(\phi) & \text{cos}(\theta)\text{sen}(\phi) \\ \text{cos}(\psi)\text{sen}(\theta)\text{cos}(\phi) + \text{sen}(\psi)\text{sen}(\phi) & \text{sen}(\psi)\text{sen}(\theta)\text{cos}(\phi) - \text{cos}(\psi)\text{sen}(\phi) & \text{cos}(\theta)\text{cos}(\phi) \end{bmatrix} \\ &= \begin{bmatrix} 0 & 0 & -1 \\ \pm \text{sen}(\phi \pm \psi) & \pm \text{cos}(\phi \mp \psi) & 0 \\ \pm \text{cos}(\phi \mp \psi) & \pm \text{sen}(\phi \pm \psi) & 0 \end{bmatrix}\end{aligned}\quad (2.47)$$

Esta condição leva a que os ângulos de Euler sejam preteridos em relação aos quatérnios para representação da orientação de um objeto no espaço tridimensional [14], [15] e [28]. Ainda assim, a utilização dos ângulos de Euler permite uma melhor percepção da orientação do objeto e continuam a ser usados numa fase final, em que o objetivo é apenas a disponibilização da informação a um utilizador, sendo os cálculos feitos com base em quatérnios. No entanto, as singularidades voltam a surgir quando os quatérnios são transformados nos respetivos ângulos de Euler (2.48). Na situação em que $\theta = \pm 90^\circ$, os denominadores das equações de ψ e ϕ são zero, não permitindo o cálculo dos ângulos *yaw* e *roll*, respetivamente [24].

$$\begin{aligned}\tan(\psi) &= \frac{2q_1q_2 + 2q_0q_3}{q_0^2 + q_1^2 - q_2^2 - q_3^2} \\ \text{sen}(\theta) &= -(2q_1q_3 - 2q_0q_2) \\ \tan(\phi) &= \frac{2q_2q_3 + 2q_0q_1}{q_0^2 - q_1^2 - q_2^2 + q_3^2}\end{aligned}\tag{2.48}$$

Conclui-se então que a utilização isolada de cada um destes sensores não é suficiente para definir a orientação em R^3 de um objeto. O acelerómetro é capaz de oferecer uma estimativa da orientação em relação ao vetor da força de gravidade, ou seja, apenas *pitch* e *roll*. No entanto, estas estimativas são bastante instáveis quando este se encontra sujeito a elevadas acelerações lineares. O giroscópio, para além de não reportar diretamente ângulos de Euler perde credibilidade ao longo do tempo de aquisição. O magnetómetro oferece uma estimativa da orientação em relação ao norte magnético, *yaw*. No entanto, necessita de receber uma boa estimativa da inclinação do sensor para que possa reportar um valor correto desse ângulo e admite vários tipos de interferências magnéticas nos seus dados. A utilização destes sensores em conjunto é realizada para poderem ser obtidas estimativas mais fiáveis da orientação.

Um IMU é um sistema que reúne um acelerómetro e giroscópio para obter uma estimativa da orientação com 6 graus de liberdade. No entanto, a falta de uma referência para corrigir o ângulo *yaw* leva a que esta rotação não seja calculada com eficácia. Um sistema AHRS introduz um magnetómetro que permite obter resultados com 9 graus de liberdade, oferecendo uma estimativa da orientação em relação ao norte magnético. A fusão dos dados destes sensores para estimar uma orientação mais fiável é denominada de fusão sensorial. No capítulo seguinte são estudados vários algoritmos de fusão sensorial baseados em quatérnios. Todavia, os ângulos de Euler voltarão a ser usados neste trabalho, nomeadamente para a disponibilização de informação da posição angular dos sensores, uma vez que são mais intuitivos.

3 FUSÃO SENSORIAL

No Capítulo 2 foi justificada a presença de três sensores distintos nos sistemas AHRS, tendo-se concluído que a utilização isolada destes não fornece uma estimativa estável da orientação. Por essa razão, a informação destes tem que ser fundida de modo a estimarem informação mais eficaz e fiável – fusão sensorial. Na literatura o algoritmo de fusão sensorial mais utilizado é o Filtro de Kalman que apesar da sua complexidade e carga computacional que impõe, revela resultados bastante satisfatórios. No entanto têm sido conduzidos diversos estudos com o objetivo de desenvolver algoritmos mais simples e menos dispendiosos computacionalmente. Neste capítulo são apresentados vários algoritmos de fusão sensorial baseados em quatérniões, nomeadamente o Filtro de Kalman Estendido [12], [13], o Filtro Complementar Explícito [29] e os métodos do Gradiente Descendente [2] e Gauss-Newton [30].

3.1 FILTRO DE KALMAN ESTENDIDO

O Filtro de Kalman é um conjunto de equações matemáticas que implementam um algoritmo capaz de produzir uma estimativa estatisticamente ótima do estado do sistema em estudo. Este é um algoritmo recursivo, uma vez que estima o estado atual tendo em consideração o anterior. O Filtro de Kalman Estendido - versão não linear do Filtro de Kalman - produz estimativas estatisticamente ótimas do sistema, mesmo na presença de relações não-lineares [31], [32]. Por essa razão, este é o algoritmo mais utilizado para fazer a fusão de dados em aplicações com sensores inerciais [2]. Em [13] é proposto um Filtro de Kalman Estendido que utiliza dados de velocidade angular como entrada e as medidas do acelerómetro como referência. O objetivo é estimar um quatérnio unitário que traduz a orientação de um IMU no espaço tridimensional, como ilustrado na Figura 3-1.



Figura 3-1 - Diagrama de implementação do Filtro de Kalman Estendido num IMU.

As equações que constituem um Filtro de Kalman Estendido dividem-se em duas categorias: preditivas (3.1) e corretivas (3.2). As primeiras são responsáveis por projetar o estado atual do sistema – estimativa *a priori*. Por sua vez, as corretivas estão encarregues de integrar na primeira estimativa um fator de correção de forma a obter uma estimativa melhorada – estimativa *a posteriori*.

$$\begin{aligned} \bar{q}_t &= A * q_{t-1} \\ \bar{p}_t &= A * p_{t-1} * A^t + Q \end{aligned} \quad (3.1)$$

$$\begin{aligned}
K &= \bar{p}_t * H^t * (H * \bar{p}_t * H^t + V * R * V^t)^{-1} \\
q_t &= \bar{q}_t + K * (z - h(\bar{q}_t)) \\
p_t &= (I - K * H) * \bar{p}_t
\end{aligned} \tag{3.2}$$

Em que \bar{q}_t é a estimativa *a priori*, \bar{p}_t a matriz de covariância do erro da estimativa *a priori*, K o ganho do filtro, q_t a estimativa *a posteriori*, p_t a matriz de covariância do erro da estimativa *a posteriori*, Q a matriz de covariância do ruído de transição de estado, R a matriz de covariância do ruído da medida de referência, A a matriz de transição de estado, $h(\bar{q}_t)$ o vetor expectável da medida de referência calculada com a estimativa *a priori*, H a matriz Jacobiano de derivadas parciais de $h(\bar{q}_t)$, V a matriz Jacobiano de derivadas parciais do ruído da equação não linear $h(\bar{q}_t)$ e z o vetor das leituras do sensor, neste caso o acelerómetro [13].

A matriz de transição A é calculada segundo a equação (3.3), em que Ω é a matriz que determina a evolução do sistema partindo da iteração anterior, I uma matriz identidade 4x4, dt o intervalo de tempo e ω_x , ω_y e ω_z são os valores de velocidade angular reportados pelo giroscópio em X, Y e Z, respetivamente [15]. O objetivo desta matriz é determinar a variação que o quaternião sofre de uma iteração para a seguinte, com respeito aos dados do giroscópio, que origina a estimativa *a priori*, \bar{q}_t .

$$\begin{aligned}
A &= I + \frac{1}{2} * \Omega * dt \\
\Omega &= \begin{bmatrix} 0 & -\omega_x & -\omega_y & -\omega_z \\ \omega_x & 0 & \omega_z & -\omega_y \\ \omega_y & -\omega_z & 0 & \omega_x \\ \omega_z & \omega_y & -\omega_x & 0 \end{bmatrix}
\end{aligned} \tag{3.3}$$

A matriz $h(t)$ é calculada utilizando a matriz de rotação apresentada em (2.4), considerando o vetor de gravidade constante no eixo Z e o quaternião estimado com os dados do giroscópio, \bar{q}_t . Estas considerações implicam que seja necessário ter conhecimento da posição inicial do sensor, mesmo que a posição indicada não seja a correta o filtro converge para o valor suposto. A equação (3.4) tem como objetivo fazer uma estimativa dos valores que o acelerómetro deve reportar. Estimativa essa que é comparada com os valores reais do acelerómetro, determinando assim um fator de correção. Este valor será posteriormente atenuado ou enfatizado por um ganho K [13].

$$\begin{aligned}
h(\bar{q}_t) &= \begin{bmatrix} 2\bar{q}_{t0}^2 - 1 + 2\bar{q}_{t1}^2 & 2\bar{q}_{t1}\bar{q}_{t2} + 2\bar{q}_{t0}\bar{q}_{t3} & 2\bar{q}_{t1}\bar{q}_{t3} - 2\bar{q}_{t0}\bar{q}_{t2} \\ 2\bar{q}_{t1}\bar{q}_{t3} - 2\bar{q}_{t0}\bar{q}_{t2} & 2\bar{q}_{t0}^2 - 1 + 2\bar{q}_{t2}^2 & 2\bar{q}_{t2}\bar{q}_{t3} + 2\bar{q}_{t0}\bar{q}_{t1} \\ 2\bar{q}_{t1}\bar{q}_{t3} + 2\bar{q}_{t0}\bar{q}_{t2} & 2\bar{q}_{t2}\bar{q}_{t3} - 2\bar{q}_{t0}\bar{q}_{t1} & 2\bar{q}_{t0}^2 - 1 + 2\bar{q}_{t3}^2 \end{bmatrix} * \begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix} \\
h(\bar{q}_t) &= \begin{bmatrix} 2\bar{q}_{t1}\bar{q}_{t3} - 2\bar{q}_{t0}\bar{q}_{t2} \\ 2\bar{q}_{t2}\bar{q}_{t3} + 2\bar{q}_{t0}\bar{q}_{t1} \\ 2\bar{q}_{t0}^2 - 1 + 2\bar{q}_{t3}^2 \end{bmatrix}
\end{aligned} \tag{3.4}$$

A matriz H , indicada na equação (3.5), é a matriz Jacobiano de derivadas parciais da matriz $h(\bar{q}_t)$.

$$H = \begin{bmatrix} -2\bar{q}_{t2} & 2\bar{q}_{t3} & -2\bar{q}_{t0} & 2\bar{q}_{t1} \\ 2\bar{q}_{t1} & 2\bar{q}_{t0} & 2\bar{q}_{t3} & 2\bar{q}_{t2} \\ 4\bar{q}_{t0} & 0 & 0 & 4\bar{q}_{t3} \end{bmatrix} \quad (3.5)$$

Este algoritmo é resumido na Figura 3-2.

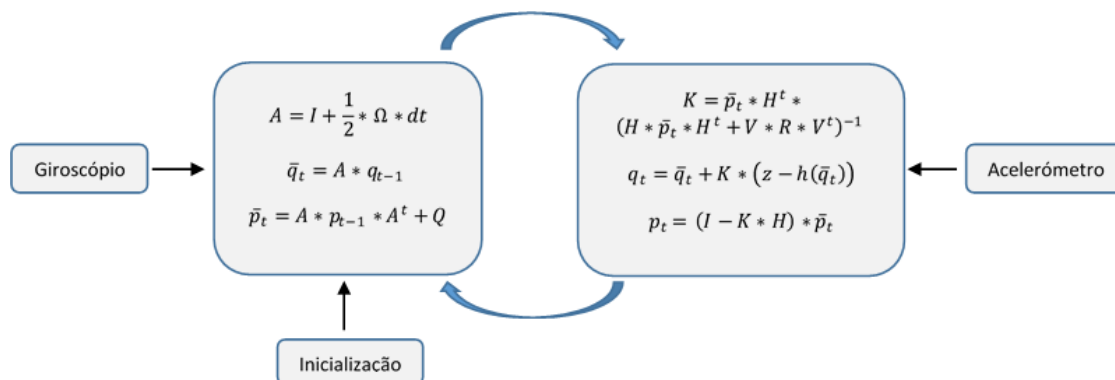


Figura 3-2 - Algoritmo do Filtro de Kalman Estendido.

O Filtro de Kalman é o algoritmo mais utilizado em fusão sensorial uma vez que, apesar de algumas desvantagens (e.g., a sua complexidade e carga computacional imposta), se revela capaz de produzir resultados satisfatórios. Vários estudos têm sido conduzidos com o intuito de desenvolver algoritmos computacionalmente menos dispendiosos que o filtro apresentado.

3.2 GRADIENTE DESCENDENTE

Em [2] é apresentado um algoritmo alternativo ao Filtro de Kalman, implementado em sistemas AHRS, que se baseia no método do Gradiente Descendente. Este método consiste em minimizar uma função objetivo na direção da maior taxa de decrescimento, a partir de uma solução inicial. Neste caso, o método é utilizado para minimizar a função que alinha as medidas de observação dos sensores com a direção do campo de referência, definido pelo vetor de gravidade e norte magnético [2]. Na Figura 3-3 ilustra-se o diagrama geral do algoritmo em questão.

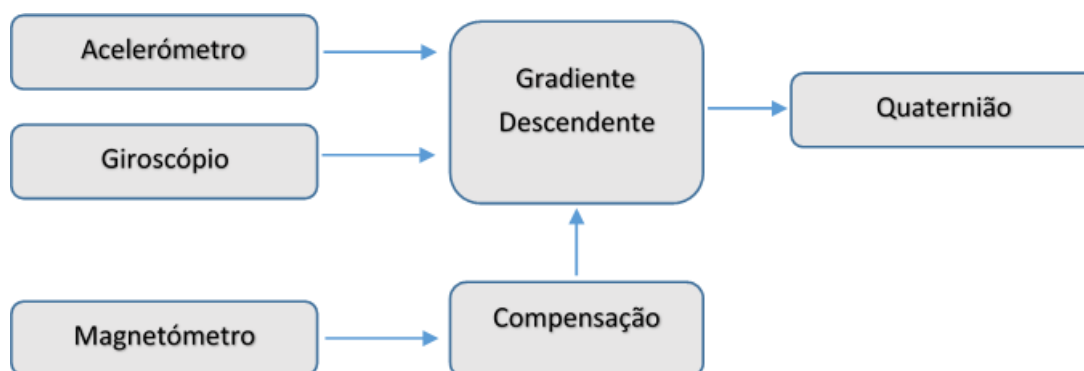


Figura 3-3 – Diagrama de implementação do algoritmo Gradiente Descendente num sistema AHRS.

Numa primeira fase deste algoritmo é necessária a determinação de uma estimativa da orientação tendo em conta os dados do giroscópio. A expressão (3.6) descreve a taxa de variação entre dois quatérnios, em que $\dot{q}_{\omega t}$ é a taxa de variação do quatérnio, \hat{q}_{t-1} o quatérnio estimado na iteração anterior e ω o vetor de dados do giroscópio [2].

$$\dot{q}_{\omega t} = \frac{1}{2} * \hat{q}_{t-1} \otimes \omega_t \quad (3.6)$$

$$\omega = [0 \quad \omega_x \quad \omega_y \quad \omega_z]$$

Em que ω_x é a velocidade angular no eixo X do giroscópio, ω_y no eixo Y, e ω_z no eixo Z. A orientação do sensor dada pelos dados do giroscópio, $q_{\omega t}$, é calculada com a expressão (3.7), integrando numericamente a derivada do quatérnio obtida em (3.6) e tendo conhecimento das condições iniciais [2].

$$q_{\omega t} = \hat{q}_{t-1} + \dot{q}_{\omega t} * \Delta t \quad (3.7)$$

É inicialmente assumido que o acelerómetro mede apenas gravidade e que o magnetómetro mede apenas o campo magnético terrestre. Se a direção do campo definido por estas variáveis for conhecido, é possível calcular uma orientação do sensor em relação a esse plano terrestre. Então, é formalizado um problema de otimização em que a orientação do sensor é encontrada quando a função alinha as medidas de observação com o campo definido como referência [2]. Este problema é formalizado em (3.8), em que \hat{q} é o quatérnio estimado na iteração anterior, \hat{d} o campo de referência e \hat{s} o campo definido pelas leituras dos sensores.

$$\min_{\hat{q} \in R^4} f(\hat{q}, \hat{d}, \hat{s}) \quad (3.8)$$

$$f(\hat{q}, \hat{d}, \hat{s}) = \hat{q}^* \otimes \hat{d} \otimes \hat{q} - \hat{s}$$

Para a definição do campo de referência é utilizada a convenção que a direção do vetor da gravidade é definida na vertical [2]. Esta convenção é ilustrada em (3.9), onde é indicada direção do vetor de gravidade, $f(\hat{q}, \hat{a})$, e a respetiva matriz Jacobiano, $J(\hat{q})$, em que \hat{q} corresponde ao quatérnio calculado na iteração anterior e \hat{a} o vetor de dados do acelerómetro.

$$f(\hat{q}, \hat{a}) = \begin{bmatrix} 2(\hat{q}_1\hat{q}_3 - \hat{q}_0\hat{q}_2) - a_x \\ 2(\hat{q}_0\hat{q}_1 + \hat{q}_2\hat{q}_3) - a_y \\ 2\left(\frac{1}{2} - \hat{q}_1^2 - \hat{q}_2^2\right) - a_z \end{bmatrix} \quad (3.9)$$

$$J(\hat{q}) = \begin{bmatrix} -2\hat{q}_2 & 2\hat{q}_3 & -2\hat{q}_0 & 2\hat{q}_1 \\ 2\hat{q}_1 & 2\hat{q}_0 & 2\hat{q}_3 & 2\hat{q}_2 \\ 0 & -4\hat{q}_1 & -4\hat{q}_2 & 0 \end{bmatrix}$$

Este algoritmo pressupõe que seja feita uma compensação ao magnetómetro, sendo, para o efeito, realizada a rotação do vetor do campo magnético, m_t , para o referencial inercial, \hat{h}_t . É também feita a consideração de que o campo magnético é normalizado para ter uma componente horizontal e outra vertical, \hat{b}_t . Este método, indicado em (3.10), não necessita da

definição de um valor constante para o campo magnético de referência, sendo esta uma das vantagens em relação a outros algoritmos [2].

$$\begin{aligned} \hat{h}_t &= \hat{q}_{t-1} \otimes m_t \otimes \hat{q}_{t-1}^* \\ \hat{b}_t &= \begin{bmatrix} 0 & \sqrt{h_x^2 + h_y^2} & 0 & h_z \end{bmatrix} = [0 \quad b_x \quad 0 \quad b_z] \end{aligned} \quad (3.10)$$

Em (3.11) é definida a direção do campo magnético terrestre, $f(\hat{q}, \hat{b}, \hat{m})$, e a respetiva matriz Jacobiano, $J(\hat{q}, \hat{b})$, em que \hat{q} corresponde ao quaternião calculado na iteração anterior, \hat{b} o vetor de dados do magnetómetro após a compensação e \hat{m} o vetor de dados que chegam do magnetómetro.

$$\begin{aligned} \hat{m} &= [0 \quad m_x \quad m_y \quad m_z] \\ f(\hat{q}, \hat{b}, \hat{m}) &= \begin{bmatrix} 2b_x \left(\frac{1}{2} - \hat{q}_2^2 - \hat{q}_3^2 \right) + 2b_z(\hat{q}_1\hat{q}_3 - \hat{q}_0\hat{q}_2) - m_x \\ 2b_x(\hat{q}_1\hat{q}_2 - \hat{q}_0\hat{q}_3) + 2b_z(\hat{q}_0\hat{q}_1 + \hat{q}_2\hat{q}_3) - m_y \\ 2b_x(\hat{q}_0\hat{q}_2 + \hat{q}_1\hat{q}_3) + 2b_z \left(\frac{1}{2} - \hat{q}_1^2 - \hat{q}_2^2 \right) - m_z \end{bmatrix} \\ J(\hat{q}, \hat{b}) &= \begin{bmatrix} -2b_z\hat{q}_2 & 2b_z\hat{q}_3 & -4b_x\hat{q}_2 - 2b_z\hat{q}_0 & -4b_x\hat{q}_3 + 2b_z\hat{q}_1 \\ -2b_x\hat{q}_3 + 2b_z\hat{q}_1 & 2b_x\hat{q}_2 + 2b_z\hat{q}_0 & 2b_x\hat{q}_1 + 2b_z\hat{q}_3 & -2b_x\hat{q}_0 + 2b_z\hat{q}_2 \\ 2b_x\hat{q}_2 & 2b_x\hat{q}_3 - 4b_z\hat{q}_1 & 2b_x\hat{q}_0 - 4b_z\hat{q}_2 & 2b_x\hat{q}_1 \end{bmatrix} \end{aligned} \quad (3.11)$$

Por forma a combinar as medições de ambos os sensores e as direções de ambos os campos, as equações finais da função objetivo e o seu jacobiano são dadas pelas expressões (3.12).

$$\begin{aligned} f_{g,b}(\hat{q}, \hat{a}, \hat{b}, \hat{m}) &= \begin{bmatrix} f(\hat{q}, \hat{a}) \\ f(\hat{q}, \hat{b}, \hat{m}) \end{bmatrix} \\ J_{g,b}(\hat{q}, \hat{b}) &= \begin{bmatrix} J(\hat{q}) \\ J(\hat{q}, \hat{b}) \end{bmatrix} \end{aligned} \quad (3.12)$$

Um método convencional de otimização requer múltiplas iterações para cada nova orientação e medição dos sensores. No entanto, é concluído em [2] que é aceitável um único passo de otimização. O modelo que devolve a orientação do objeto num determinado instante t , definida pelas medições do acelerómetro e do magnetómetro, $q_{\nabla t}$, é dada pela equação (3.13). Em que ∇f é um vetor que indica o erro da função objetivo e $\|\nabla f\|$ a sua norma.

$$\begin{aligned} q_{\nabla t} &= \hat{q}_{t-1} - \beta * \frac{\nabla f}{\|\nabla f\|} \\ \nabla f &= J_{g,b}(\hat{q}, \hat{b})^t * f_{g,b}(\hat{q}, \hat{a}, \hat{b}, \hat{m}) \end{aligned} \quad (3.13)$$

É necessária a fusão de $q_{\nabla t}$ e $q_{\omega t}$ por forma a obter uma estimativa estável da orientação. Na prática, $q_{\omega t}$ deriva para o infinito devido ao ruído nas amostras do giroscópio e $q_{\nabla t}$ devolve estimativas incorretas quando o acelerómetro não se encontra estacionário ou o magnetómetro sofre de interferências. O objetivo desta fusão passa por utilizar $q_{\omega t}$ para filtrar ruído de alta

frequência em $q_{\nabla t}$ e este último para compensar a deriva para o infinito de $q_{\omega t}$, resultante da integração numérica, assim como permitir a convergência de valores iniciais incorretos [2]. Por fim, as fórmulas finais dessa fusão são dadas pelas equações (3.14)

$$\begin{aligned}\dot{q}_t &= \dot{q}_{\omega t} - \beta * \frac{\nabla f}{\|\nabla f\|} \\ \hat{q}_t &= \hat{q}_{t-1} + \dot{q}_t * \Delta t\end{aligned}\quad (3.14)$$

Em que $\dot{q}_{\omega t}$ é a taxa de variação do quaternião, β o erro do giroscópio, \hat{q}_t o quaternião final, \hat{q}_{t-1} o quaternião estimado na iteração anterior e Δt o intervalo de tempo. A variável \dot{q}_t define a taxa de variação do quaternião estimado com o giroscópio, $\dot{q}_{\omega t}$, otimizado com o método do Gradiente Descendente. O único parâmetro ajustável do filtro é a variável β , que representa o erro do giroscópio expresso como a magnitude da derivada do quaternião (3.15). É conveniente utilizar ω_{max} como o erro máximo do giroscópio em cada eixo [2].

$$\beta = \left\| \frac{1}{2} \hat{q} \otimes [0 \quad \omega_{max} \quad \omega_{max} \quad \omega_{max}] \right\| = \sqrt{\frac{3}{4}} * \omega_{max} \quad (3.15)$$

Na Figura 3-4 é ilustrado um esquema simplificado do algoritmo exposto.

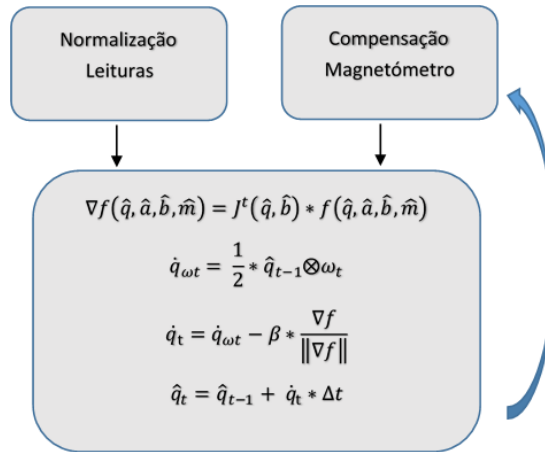


Figura 3-4 - Algoritmo do Gradiente Descendente.

Em [2] conclui-se que o algoritmo de compensação do magnetómetro é uma vantagem em relação a outros algoritmos de fusão sensorial e que pode ser facilmente incorporado nestes.

3.3 FILTRO DE KALMAN ESTENDIDO COM DUPLA FASE DE CORREÇÃO

Em [12] é proposto um algoritmo baseado no Filtro de Kalman Estendido que implementa uma segunda fase corretiva para introduzir a referência do campo magnético no algoritmo descrito na secção 3.1. No entanto, o método proposto para a compensação do magnetómetro considera que o campo magnético terrestre normalizado, \hat{b} , está direcionado apenas no eixo Y do sensor e a sua componente vertical não é considerada. Esta consideração é formalizada na expressão (3.16).

$$\hat{b} = R_I^B * \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} \quad (3.16)$$

A definição de um valor constante para o campo magnético é uma desvantagem deste algoritmo quando comparado com o algoritmo apresentado na secção 3.2. É apresentado em [2] um algoritmo de compensação do magnetómetro que não necessita da definição de um valor constante para o campo magnético de referência e é concluído que este pode ser incorporado em outros algoritmos de fusão sensorial. É proposto nesta secção um algoritmo baseado no Filtro de Kalman Estendido com Dupla Fase de Correção apresentado em [12], implementando o método de compensação do magnetómetro apresentado em [2]. Resumidamente, a primeira fase corresponde ao algoritmo apresentado na secção 3.1, e é implementada uma segunda fase de correção para retificar a estimativa *a priori* com as medidas de referência do magnetómetro.

O vetor da gravidade apenas corrige os ângulos *pitch* e *roll*. Para garantir que a correção da estimativa *yaw* é feita apenas na segunda fase corretiva, quando é calculada a estimativa *a posteriori* da primeira fase, q_{t1} , a sua terceira componente vetorial é colocada a zero. Sendo q_1 a estimativa final da primeira fase corretiva e \bar{q}_t a estimativa *a priori* (3.17).

$$\begin{aligned} q_{t1} &= q_{t1,0} + q_{t1,1} + q_{t1,2} + 0 * q_{t1,3} \\ q_1 &= \bar{q}_t + q_{t1} \end{aligned} \quad (3.17)$$

O campo magnético de referência é obtido mediante a expressão (3.18), em que m_t é o vetor de dados do magnetómetro.

$$\begin{aligned} \hat{h}_t &= \bar{q}_t \otimes m_t \otimes \bar{q}_t^* \\ \hat{b}_t &= \begin{bmatrix} 0 & \sqrt{h_x^2 + h_y^2} & 0 & h_z \end{bmatrix} = \begin{bmatrix} 0 & b_x & 0 & b_z \end{bmatrix} \end{aligned} \quad (3.18)$$

Os valores expetáveis do campo magnético calculados pelo giroscópio são dados por $h_2(\bar{q}_t)$. Para diferenciar as variáveis utilizadas nas fases de correção 1 e 2, são utilizados os subscritos 1 e 2, respetivamente (3.19).

$$\begin{aligned} h_2(\bar{q}_t) &= \begin{bmatrix} 2b_x \left(\frac{1}{2} - \bar{q}_{t2}^2 - \bar{q}_{t3}^2 \right) + 2b_z (\bar{q}_{t1} \bar{q}_{t3} - \bar{q}_{t0} \bar{q}_{t2}) \\ 2b_x (\bar{q}_{t1} \bar{q}_{t2} - \bar{q}_{t0} \bar{q}_{t3}) + 2b_z (\bar{q}_{t0} \bar{q}_{t1} + \bar{q}_{t2} \bar{q}_{t3}) \\ 2b_x (\bar{q}_{t0} \bar{q}_{t2} + \bar{q}_{t1} \bar{q}_{t3}) + 2b_z \left(\frac{1}{2} - \bar{q}_{t1}^2 - \bar{q}_{t2}^2 \right) \end{bmatrix} \\ H_2 &= \begin{bmatrix} -2b_x \bar{q}_{t2} & 2b_z \bar{q}_{t3} & -4b_x \bar{q}_{t2} - 2b_z \bar{q}_{t0} & -4b_x \bar{q}_{t3} + 2b_z \bar{q}_{t1} \\ -2b_x \bar{q}_{t3} + 2b_z \bar{q}_{t1} & 2b_x \bar{q}_{t2} + 2b_z \bar{q}_{t0} & 2b_x \bar{q}_{t1} + 2b_z \bar{q}_{t3} & -2b_x \bar{q}_{t0} + 2b_z \bar{q}_{t2} \\ 2b_x \bar{q}_{t2} & 2b_x \bar{q}_{t3} - 4b_z \bar{q}_{t1} & 2b_x \bar{q}_{t0} - 4b_z \bar{q}_{t2} & 2b_x \bar{q}_{t1} \end{bmatrix} \end{aligned} \quad (3.19)$$

O campo magnético de referência consegue apenas corrigir a estimativa *yaw*. Por essa razão, a primeira e segunda componente vetorial do quaternião da estimativa *a posteriori* da segunda fase corretiva, q_{t2} , são colocadas a zero. A estimativa final do algoritmo, q_t , é dada pela soma das estimativas obtidas em cada fase (3.20).

$$q_{t2} = q_{t2,0} + 0 * q_{t2,1} + 0 * q_{t2,2} + q_{t1,3} \quad (3.20)$$

$$q_t = q_1 + q_{t2}$$

O algoritmo em questão encontra-se resumido na Figura 3-5.

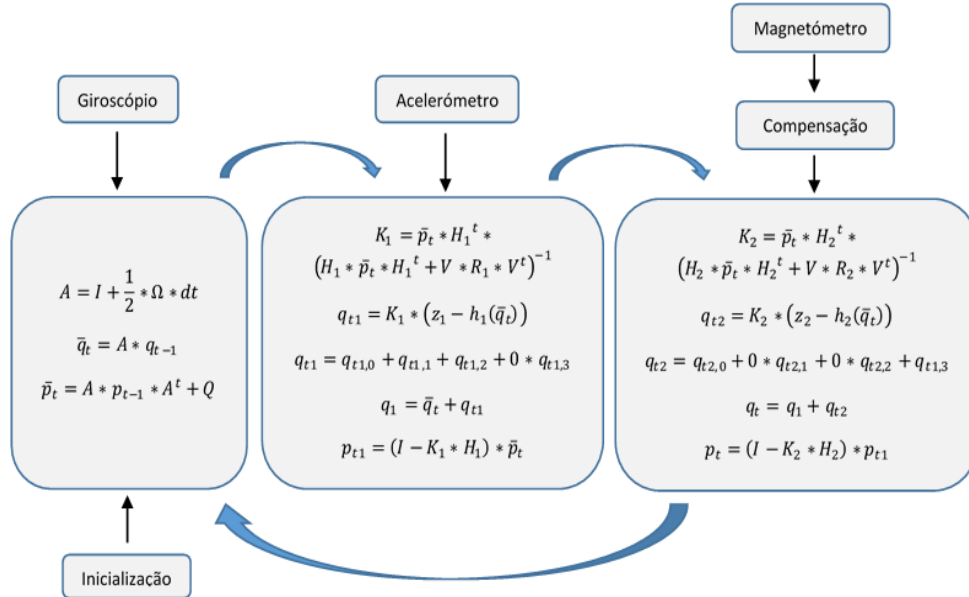


Figura 3-5 - Algoritmo do Filtro de Kalman Estendido com dupla fase corretiva.

A principal desvantagem deste filtro reside no acréscimo da sua complexidade, uma vez que a fase corretiva acontece em duas fases e, como tal, as equações necessárias nessa fase são duplicadas. No entanto este apresenta alguma robustez perante perturbações do campo magnético, uma vez que estas afetam somente a estimativa *yaw* da orientação dos sensores [12].

3.4 FILTRO COMPLEMENTAR EXPLÍCITO

Em [29] é proposto um algoritmo baseado num filtro complementar, denominado Filtro Complementar Explícito, para fazer a fusão dos dados de um IMU. No entanto, este algoritmo pode ser modificado para uma implementação em sistemas AHRS. Nas secções seguintes são apresentadas duas implementações deste algoritmo.

3.4.1 IMU

A ideia principal deste método é corrigir o vetor de velocidade angular, ω , com um fator de correção, δ , calculado por meio de um controlador proporcional integral (3.21). Os únicos parâmetros de afinação deste filtro são as variáveis Kp e Ki , ganho proporcional e integral, respetivamente [29].

$$\delta = Kp * e + Ki \int e \quad (3.21)$$

Este algoritmo utiliza a rotação relativa entre a direção da gravidade indicada pelo acelerómetro e a orientação estimada anteriormente, denotada por e , calculada pelo produto vetorial entre esses dois vetores, \bar{v} e \hat{v} , respetivamente (3.22). Em que a_x , a_y e a_z são os valores de aceleração em X, Y e Z, e q o quaternião estimado na iteração anterior [31].

$$e = \bar{v} \times \hat{v}$$

$$\hat{v} = \begin{bmatrix} 2(q_1q_3 - q_0q_2) \\ 2(q_0q_1 + q_2q_3) \\ 2\left(\frac{1}{2} - q_1^2 - q_2^2\right) \end{bmatrix} \quad (3.22)$$

$$\bar{v} = \begin{bmatrix} a_x \\ a_y \\ a_z \end{bmatrix}$$

Sendo δ a inovação do filtro, o vetor de velocidade angular ω pode ser corrigido somando ambas as componentes [29]. Assim, a taxa de variação do quaternião na iteração atual, \dot{q}_t , é calculada mediante a expressão (3.23).

$$\dot{q}_t = \frac{1}{2} * \hat{q}_{t-1} \otimes (\omega_t + \delta) \quad (3.23)$$

O quaternião final que traduz a orientação do sensor na iteração atual \hat{q}_t é dada pela integração numérica da taxa de variação calculada, \dot{q}_t (3.24).

$$\hat{q}_t = \hat{q}_{t-1} + \dot{q}_t * \Delta t \quad (3.24)$$

Na Figura 3-6 encontra-se um resumo do algoritmo apresentado.

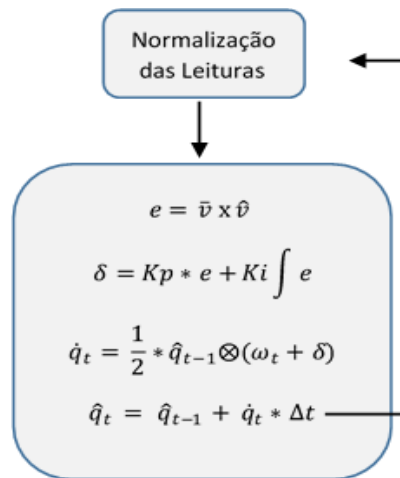


Figura 3-6 - Algoritmo do Filtro Complementar Explícito implementado num IMU.

3.4.2 AHRS

Este algoritmo pode ser aplicado a sistemas AHRS para produzir uma estimativa da orientação com 9 graus de liberdade. Para esse efeito é necessário definir a direção do campo magnético de referência utilizando o algoritmo apresentado em [2]. Similarmente ao algoritmo anterior, é também calculado o produto vetorial entre o vetor de dados do magnetómetro, \bar{m} , e a referência do campo magnético, \hat{m} . Os dois produtos vetoriais efetuados para o acelerómetro e

magnetómetro são então somados para providenciar um fator de correção. Na Figura 3-7 encontra-se um esquema resumido do algoritmo exposto.

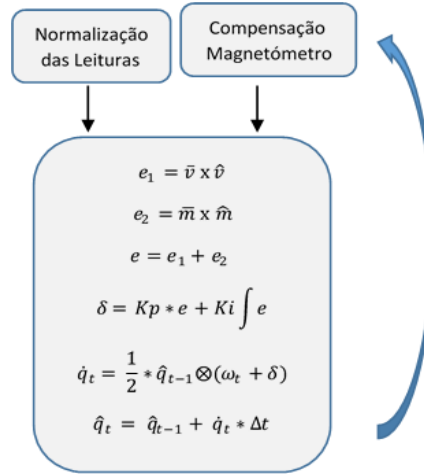


Figura 3-7 - Algoritmo do Filtro Complementar Explícito implementado num sistema AHRS.

3.5 GAUSS-NEWTON

O método de Gauss-Newton é um algoritmo de otimização alternativo ao método do Gradiente Descendente apresentado em 3.2 [30]. Este algoritmo é utilizado para encontrar o quaternião que define a orientação do sensor em relação aos dados do acelerómetro e magnetómetro, tendo uma etapa de otimização, com k passos, definida pela expressão (3.25), em que x é o estado do sistema.

$$x_{k+1} = x_k - [J_k^T J_k]^{-1} \cdot J_k^T \cdot f(\hat{q}_{t-1}, z_0, z_t) \quad (3.25)$$

J_k é o Jacobiano da função objetivo, f , definida em (3.26).

$$f(\hat{q}_{t-1}, z_0, z_t) = z_0 - M_t * z_t \quad (3.26)$$

Em que z_0 é o campo de referência definido pelo campo gravítico e magnético terrestre, z_t é o vetor de leituras do acelerómetro e magnetómetro, M_t a matriz DCM equivalente a R_B^I , b_x a componente horizontal do campo magnético de referência e b_z a componente vertical, a_x , a_y e a_z os valores das leituras do acelerómetro e m_x , m_y e m_z os valores das leituras do magnetómetro (3.27).

$$M_t = \begin{bmatrix} 2q_0^2 - 1 + 2q_1^2 & 2q_1q_2 - 2q_0q_3 & 2q_1q_3 + 2q_0q_2 \\ 2q_1q_2 + 2q_0q_3 & 2q_0^2 - 1 + 2q_2^2 & 2q_2q_3 - 2q_0q_1 \\ 2q_1q_3 - 2q_0q_2 & 2q_2q_3 + 2q_0q_1 & 2q_0^2 - 1 + 2q_3^2 \end{bmatrix} \quad (3.27)$$

$$z_0 = [0 \quad 0 \quad 1 \quad b_x \quad 0 \quad b_z]$$

$$z_t = [a_x \quad a_y \quad a_z \quad m_x \quad m_y \quad m_z]$$

De facto, como estão a ser consideradas duas contribuições diferentes para estimar a orientação, a matriz M_t faz a rotação dos dois vetores (3.28).

$$M_t = \begin{bmatrix} M & 0 \\ 0 & M \end{bmatrix}$$

$$J_t = \begin{bmatrix} 2q_1a_x + 2q_2a_y + 2q_3a_z & -2q_2a_x + 2q_1a_y + 2q_0a_z & -2q_3a_x - 2q_0a_y + 2q_2a_z & 2q_0a_x - 2q_3a_y + 2q_2a_z \\ 2q_2a_x - 2q_1a_y - 2q_0a_z & 2q_1a_x + 2q_2a_y + 2q_3a_z & 2q_0a_x - 2q_3a_y + 2q_2a_z & 2q_3a_x + 2q_0a_y - 2q_1a_z \\ 2q_3a_x + 2q_0a_y - 2q_1a_z & -2q_0a_x + 2q_3a_y - 2q_2a_z & 2q_1a_x + 2q_2a_y + 2q_3a_z & -2q_2a_x + 2q_1a_y + 2q_0a_z \\ 2q_1m_x + 2q_2m_y + 2q_3m_z & -2q_2m_x + 2q_1m_y + 2q_0m_z & -2q_3m_x - 2q_0m_y + 2q_2m_z & 2q_0m_x - 2q_3m_y + 2q_2m_z \\ 2q_2m_x - 2q_1m_y - 2q_0m_z & 2q_1m_x + 2q_2m_y + 2q_3m_z & 2q_0m_x - 2q_3m_y + 2q_2m_z & 2q_3m_x + 2q_0m_y - 2q_2m_z \\ 2q_3m_x + 2q_0m_y - 2q_2m_z & -2q_2m_x + 2q_1m_y + 2q_0m_z & 2q_1m_x + 2q_2m_y + 2q_3m_z & -2q_2m_x + 2q_1m_y + 2q_0m_z \end{bmatrix} \quad (3.28)$$

O quaternião do passo de otimização $k+1$ pode ser calculado pela expressão (3.29). O objetivo deste algoritmo é determinar um quaternião, no fim dos k passos de otimização, que defina a orientação estimada apenas com as medidas do acelerómetro e magnetómetro, denominado de quaternião de observação, \hat{q}_{obs} . É este quaternião que volta a entrar no primeiro passo do algoritmo de Gauss-Newton na iteração seguinte [30].

$$q_{k+1} = q_k - [J_k^T J_k]^{-1} \cdot J_k^T \cdot (z_0 - M_t * z_t) \quad (3.29)$$

Neste algoritmo também é necessária a compensação do campo magnético (3.30), que é feita seguindo o mesmo procedimento implementado em [2].

$$\hat{h}_t = \hat{q}_{t-1} \otimes m_t \otimes \hat{q}_{t-1}^*$$

$$\hat{b}_t = \begin{bmatrix} 0 & \sqrt{h_x^2 + h_y^2} & 0 & h_z \end{bmatrix} = \begin{bmatrix} 0 & b_x & 0 & b_z \end{bmatrix} \quad (3.30)$$

Na Figura 3-8 encontra-se um esquema simplificado do algoritmo em questão.

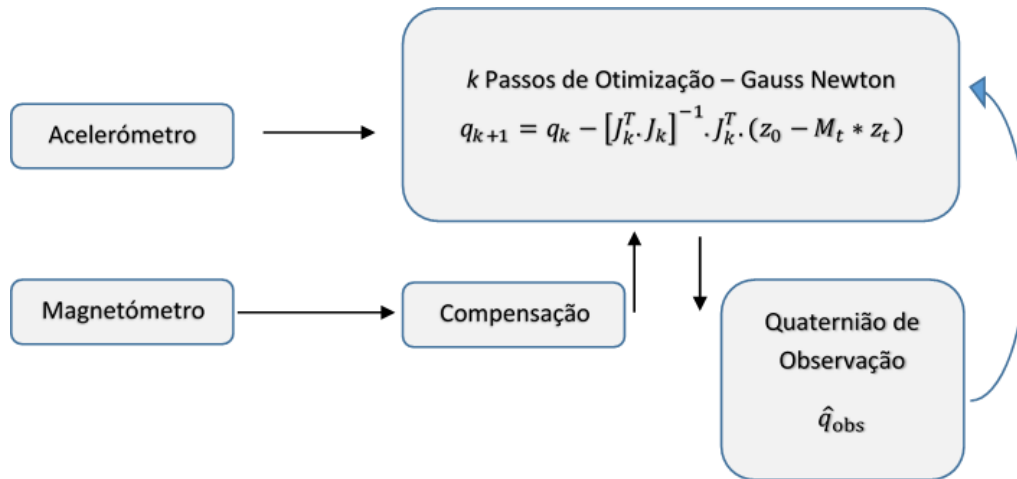


Figura 3-8 - Algoritmo Gauss-Newton.

A fusão entre a orientação estimada pelo giroscópio e a orientação estimada pelas leituras do magnetómetro e acelerómetro mediante o uso do método de Gauss-Newton pode ser feita utilizando um Filtro Complementar ou um Filtro de Kalman (Figura 3-9).

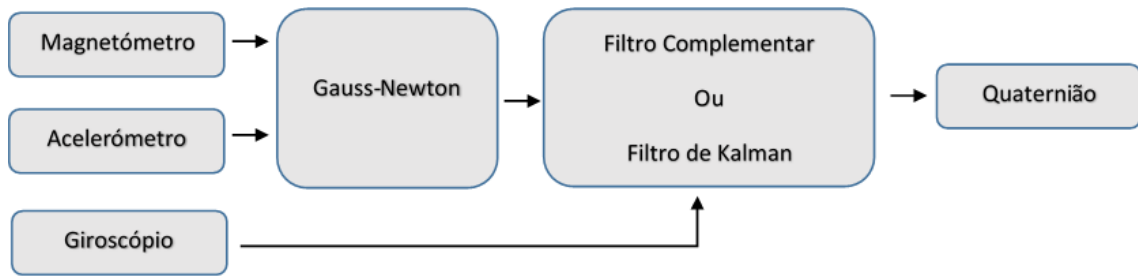


Figura 3-9 - Esquema de funcionamento do método de Gauss-Newton.

Na secção seguinte são apresentados os passos necessários para implementar a fusão do quaternião de observação com o estimado pelo giroscópio.

3.5.1 FILTRO COMPLEMENTAR

Depois de realizados os passos do método de Gauss-Newton e encontrado o quaternião de observação, este pode ser fundido com o quaternião estimado com os dados do giroscópio mediante um Filtro Complementar. Esta fusão é dada pelas expressões (3.31), em que \hat{q}_t é o quaternião final, \hat{q}_{gt} o quaternião obtido do giroscópio, \hat{q}_{obs} o quaternião de observação, k o parâmetro de afinação do filtro com um valor entre 0 e 1, \dot{q}_t a taxa de variação do quaternião, ω_t o quaternião puro construído com os dados do giroscópio e Δt o período de amostragem.

$$\begin{aligned}\hat{q}_t &= \hat{q}_{gt} * k + \hat{q}_{obs}(1 - k) \\ \dot{q}_t &= \frac{1}{2} * \hat{q}_{t-1} \otimes (\omega_t) \\ \omega_t &= [0 \quad \omega_x \quad \omega_y \quad \omega_z] \\ \hat{q}_{gt} &= \hat{q}_{t-1} + \dot{q}_t * \Delta t\end{aligned}\tag{3.31}$$

Na Figura 3-10 encontra-se um esquema simplificado da fusão descrita.

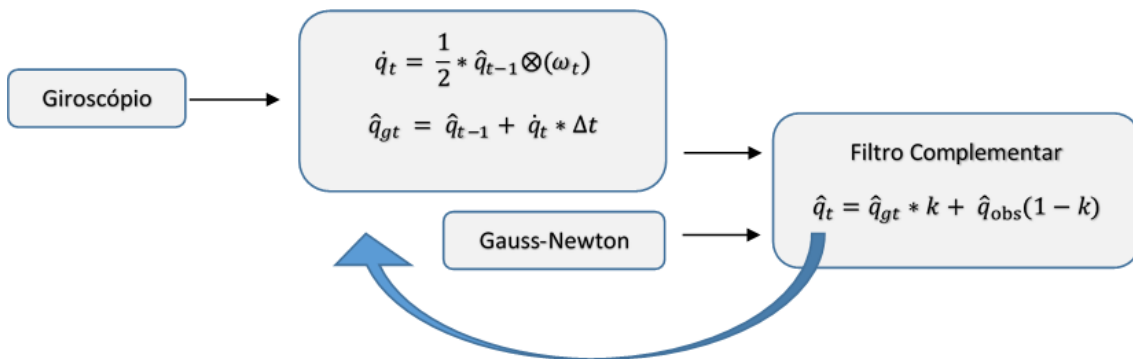


Figura 3-10 – Algoritmo do filtro complementar para fusão de dados do método de Gauss-Newton.

3.5.2 FILTRO DE KALMAN

A fusão mediante o uso de Filtro de Kalman é feita de forma semelhante ao algoritmo apresentado na secção 3.1. Numa primeira fase é feita uma estimativa da orientação – estimativa *a priori* – determinada mediante as expressões (3.32).

$$\Omega = \begin{bmatrix} 0 & -\omega_1 & -\omega_2 & -\omega_3 \\ \omega_1 & 0 & \omega_3 & -\omega_2 \\ \omega_2 & -\omega_3 & 0 & \omega_1 \\ \omega_3 & \omega_2 & -\omega_1 & 0 \end{bmatrix} \quad (3.32)$$

$$A = I + \frac{1}{2} * \Omega * dt$$

$$\bar{q}_t = A * q_{t-1}$$

De seguida é calculada a matriz de covariância do erro da estimativa *a priori* \bar{p}_t , segundo a expressão (3.33).

$$\bar{p}_t = A * p_{t-1} * A^t + Q \quad (3.33)$$

Também é necessário calcular o ganho do filtro, K , mediante a expressão (3.34). Neste caso a matriz H é uma matriz identidade, uma vez que a estimativa *a priori*, \bar{q}_t , e a referência, \hat{q}_{obs} , são um quaternião que representa a orientação no espaço tridimensional [30].

$$K = \bar{p}_t * H^t * (H * \bar{p}_t * H^t + R)^{-1} \quad (3.34)$$

Por fim, é calculada a estimativa *a posteriori*, q_t , e a sua matriz de covariância, p_t , mediante as expressões (3.35).

$$\begin{aligned} q_t &= \bar{q}_t + K * (\hat{q}_{obs} - H * \bar{q}_t) \\ p_t &= (I - K * H) * \bar{p}_t \end{aligned} \quad (3.35)$$

Este algoritmo é resumido pela Figura 3-11.

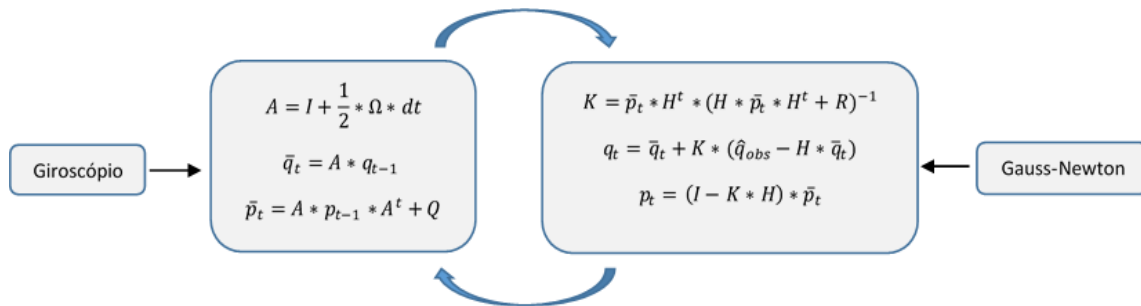


Figura 3-11 – Filtro de Kalman para fusão de dados do método de Gauss-Newton.

Em [30], compara-se este método com o do Gradiente Descendente. É concluído que o Gauss-Newton é mais rápido a convergir e requer apenas 3 iterações para chegar ao resultado ótimo, em contraste com o do Gradiente Descendente – 10 iterações. No entanto, o algoritmo do Gradiente Descendente proposto no capítulo 3.2 requer apenas um único passo de otimização. Entre a fusão com o Filtro Complementar e de Kalman, o último tem um tempo resposta mais rápido quando o sensor é submetido a elevadas acelerações devido ao seu ganho variável. Apesar disso, a complexidade de implementação do filtro e a elevada carga computacional mantêm-se, enquanto o Filtro Complementar é mais simples de implementar.

4 INFRAESTRUTURA PARA O TESTE E DESENVOLVIMENTO DE ALGORITMOS DE FUSÃO SENSORIAL

Neste capítulo é apresentado o protocolo de comunicação BLE, são apresentados os dispositivos que foram utilizados ao longo deste trabalho, bem como o protótipo desenvolvido. São também apresentadas as aplicações desenvolvidas para cada um dos dispositivos utilizados. Por fim é apresentado o método de obtenção de resultados e de avaliação dos algoritmos desenvolvidos ao longo desta dissertação.

4.1 BLUETOOTH LOW ENERGY

O Bluetooth Low Energy ou Bluetooth 4.0 é a versão do Bluetooth focada no consumo de energia, que permite o desenvolvimento de dispositivos sem-fios, com grande autonomia. O BLE é desenvolvido pelo Bluetooth SIG (*Special Interest Group*) com o objetivo de integrar os mercados de aplicações em saúde e desporto, onde os requisitos são o baixo consumo de energia, baixo custo, tamanho reduzido e a compatibilidade com diversos *smartphones*, *tablets* e computadores existentes no mercado. Por esta razão foi este o protocolo escolhido para implementar as soluções em estudo ao longo do trabalho e segundo o qual foi desenvolvida uma solução de *hardware*.

O protocolo Bluetooth é definido de acordo com o modelo OSI (*Open Systems Interconnection*) dividido em camadas. Estas camadas são agrupadas em duas secções distintas: controlador (*controller*) e hospedeiro (*host*). O controlador opera as camadas inferiores do protocolo como a camada LL (*Link Layer*) e PHY (*Physical Layer*), responsáveis pela gestão das restrições de tempo, assim como pela formatação dos dados. O hospedeiro define os perfis que armazenam dados e enviam / recebem comandos. As camadas BLE estão ilustradas na Figura 4-1.

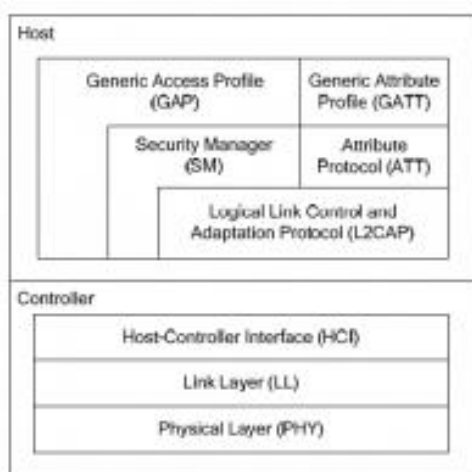


Figura 4-1 – Camadas do protocolo de comunicação BLE. Imagem obtida em [33].

O desenvolvimento de aplicações é normalmente restrito às camadas GAP (*Generic Application Profile*) e GATT (*Generic Attribute Profile*). O GATT implementa perfis que armazenam os dados do dispositivo numa estrutura semelhante a uma base de dados, sendo o acesso feito num

modo de operação de cliente-servidor. Tal como ilustrado na Figura 4-2, a base de dados é constituída por estruturas denominadas por características. Estas estruturas são agrupadas em serviços para descrever os pontos de acesso aos dados disponíveis do dispositivo. Por exemplo, uma característica pode ser o ponto de acesso às medições de um sensor como um acelerómetro. O serviço contendo esses dados pode também apresentar outras características que armazenam as restantes informações do sensor como a identificação, o formato ou a taxa de amostragem. Usando o modo de operação cliente / servidor significa que o cliente pede ao servidor dados ao qual este responde com os valores apropriados. O cliente também pode escrever para uma característica para configurar o modo de funcionamento do servidor.

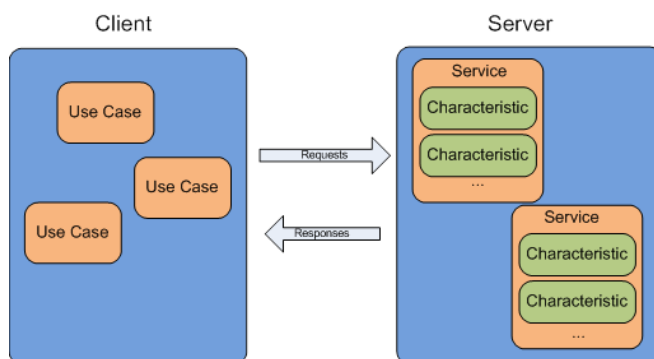


Figura 4-2 – Características de um dispositivo BLE num modo de operação cliente/servidor. Imagem obtida em [33].

4.2 SENSORTAG

O SensorTag da Texas Instruments (TI), ilustrado na Figura 4-3, é um dispositivo Bluetooth Low Energy que permite a utilização de vários sensores, entre os quais o acelerómetro triaxial KXTJ9 da Kionix, o giroscópio triaxial IMU-3000 da Invensense e o magnetómetro triaxial MAG3110 da Freescale. O seu microprocessador é o CC2541 (*core* 8051) da TI, proprietário do protocolo BLE, que juntamente com o número de sensores que tem à sua disposição torna o SensorTag um *kit* de desenvolvimento de aplicações sem-fios em diversas áreas [34].



Figura 4-3 - SensorTag da Texas Instruments. Imagem obtida em [34].

Em paralelo com o SensorTag, foi utilizado o CC2540 USB Evaluation Module Kit, Figura 4-4, que para além de habilitar a versão BLE no computador utilizado, permite estudar o protocolo de comunicação verificando os pacotes de dados enviados e recebidos pelo SensorTag [35]. Para esse efeito existem vários *softwares* disponibilizados pela TI, como o BTool e o BLE Device Monitor. O primeiro é uma aplicação que comunica com o CC2540 permitindo o

controlo do SensorTag enviando comandos por porta série [36]. O último é uma aplicação que apresenta os serviços, características e atributos de qualquer dispositivo BLE [37]. Neste trabalho o SensorTag foi utilizado com o intuito de fazer um primeiro estudo sobre os dados que chegam dos sensores inerciais.



Figura 4-4 - CC2540 USB Evaluation Module Kit.

O Matlab foi a plataforma escolhida para o desenvolvimento de uma GUI (*Graphical User Interface*) que permite a análise em tempo-real dos dados de cada sensor. Esta interface tem a capacidade de estabelecer uma comunicação por porta série com o CC2540, emparelhar o SensorTag, configurar as notificações e guardar dados num ficheiro de texto. Apenas a gama do acelerómetro é possível alterar através do BLE durante a configuração, tendo sido escolhido o intervalo de ± 2 g. No caso do giroscópio e magnetómetro os valores são fixos de ± 250 °/s e ± 1000 μ T, respetivamente. Os dados enviados pelos sensores vêm representados em complemento para dois pelo que têm de ser convertidos para os valores correspondentes - positivos ou negativos. É importante salientar que o acelerómetro tem uma resolução de 8 bits, pelo que pode assumir 256 valores, enquanto o giroscópio e magnetómetro são de 16 bits, podendo adotar 65536 valores. A variável *Range* corresponde ao tamanho do intervalo de valores que os sensores podem reportar, sendo no caso do acelerómetro igual a 4, do giroscópio igual a 500 e magnetómetro igual a 2000. A conversão dos dados do acelerómetro é dada pelas equações (4.1).

$$\begin{aligned} Valor[g] &= \frac{Valor}{256} \cdot \frac{Range}{256}, Valor < 128 \\ Valor[g] &= \frac{(Valor - 256)}{256} \cdot \frac{Range}{256}, Valor \geq 128 \end{aligned} \quad (4.1)$$

Para o caso do giroscópio e magnetómetro, a conversão é feita pelas equações (4.2).

$$\begin{aligned} Valor[^\circ/s, \mu T] &= \frac{Valor}{65536} \cdot \frac{Range}{65536}, Valor < 32768 \\ Valor[g^\circ/s, \mu T] &= \frac{(Valor - 65536)}{65536} \cdot \frac{Range}{65536}, Valor \geq 32768 \end{aligned} \quad (4.2)$$

Os dados dos sensores, após a conversão, são colocados em gráficos distintos e sujeitos a algoritmos de fusão sensorial, produzindo uma orientação do SensorTag que é representada por uma animação em três dimensões. Apesar dos cálculos serem realizados com base em quaterniões, são disponibilizados graficamente os ângulos de Euler resultantes. Também é disponibilizada uma bússola para fazer a representação do ângulo do magnetômetro em relação ao Norte magnético. Foram ainda implementados os algoritmos necessários ao cálculo dos ângulos de Euler partindo dos dados individuais de cada sensor, sendo esses resultados colocados nas caixas de texto respectivas. A Figura 4-5 ilustra a organização da aplicação desenvolvida acompanhada de uma legenda dos seus vários constituintes.

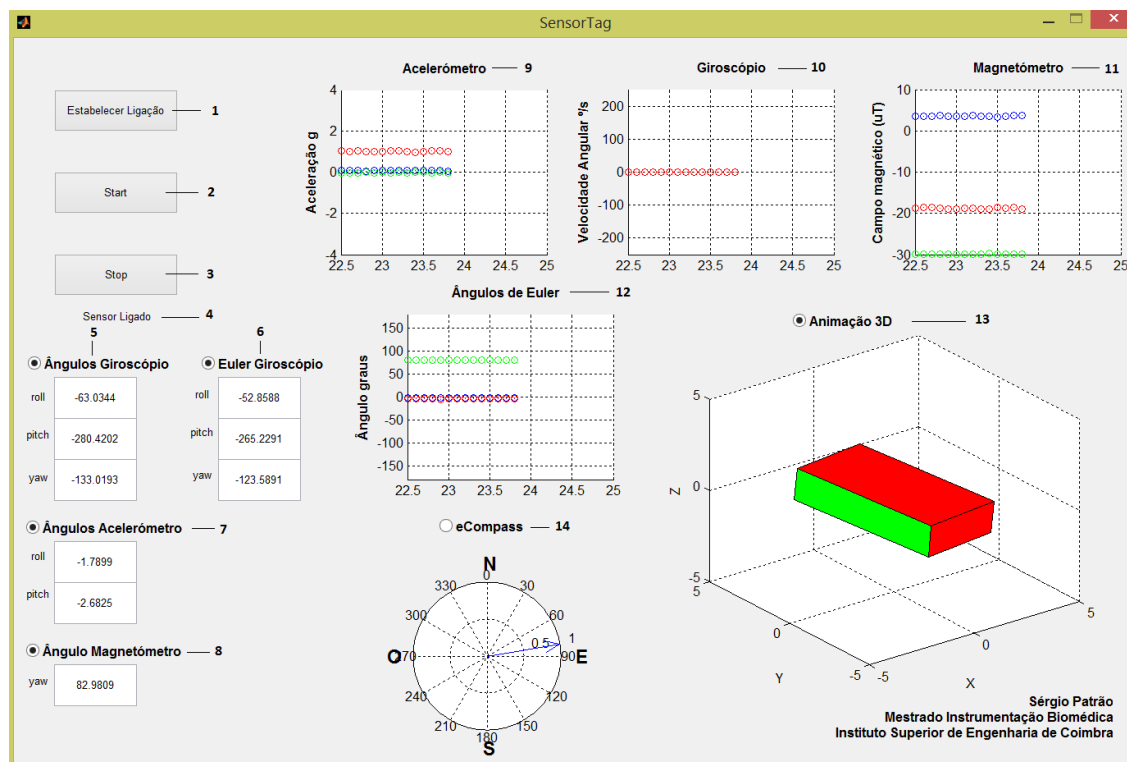


Figura 4-5 - Organização da aplicação desenvolvida em Matlab.

1. Botão que estabelece a comunicação entre o SensorTag e o CC2540.
2. Botão para iniciar a aquisição dos dados.
3. Botão para parar a aquisição. Ao pressionar este botão, é perguntado ao utilizador se quer gravar os dados adquiridos em formato de texto, concedendo a possibilidade de criar um novo ficheiro de dados ou reescrever um já existente.
4. Notificação que indica o estado da ligação – “Sensor Desligado”, “A ligar...” e “Sensor Ligado”.
5. Botão para iniciar ou parar o cálculo de ângulos com os dados do giroscópio.
6. Botão para iniciar ou parar o cálculo de ângulos de Euler com os dados do giroscópio.
7. Botão para iniciar ou parar o cálculo de ângulos de Euler com os dados do acelerômetro.
8. Botão para iniciar ou parar os cálculos do ângulo em relação ao Norte magnético com os dados do magnetômetro.

9. Gráfico que disponibiliza os dados que chegam do acelerómetro. Sendo a azul os dados relativos ao eixo X, a verde os Y e vermelho os Z.
10. Gráfico que disponibiliza os dados que chegam do giroscópio. Sendo a azul os dados relativos ao eixo X, a verde os Y e vermelho os Z.
11. Gráfico que disponibiliza os dados que chegam do magnetómetro. Sendo a azul os dados relativos ao eixo X, a verde os Y e vermelho os Z.
12. Gráfico que disponibiliza os ângulos de Euler calculados com o algoritmo de fusão sensorial. Sendo a azul o ângulo *roll*, a verde o *yaw* e vermelho o *pitch*.
13. Animação 3D que representa a orientação calculada com os algoritmos de fusão sensorial.
14. Bússola que representa o ângulo em relação ao Norte magnético calculado com o algoritmo de fusão sensorial.

O estudo do SensorTag serviu para perceber como os dados dos sensores inerciais se relacionam com a realidade dos movimentos realizados e que condições devem ser garantidas para que estes possam integrar um algoritmo de fusão sensorial. Concluiu-se que os dados reportados devem satisfazer um determinado referencial, Figura 4-6, mesmo que a disposição dos sensores na placa de circuito impresso não seja coerente. Tendo em conta a informação disponível na folha de especificações de cada sensor, foram determinadas as transformações necessárias para alinhar a orientação dos sensores com o referencial definido (Tabela 4-1). Em que x_i , y_i e z_i são os dados que chegam inicialmente de cada sensor, e x_f , y_f e z_f os dados depois da transformação.

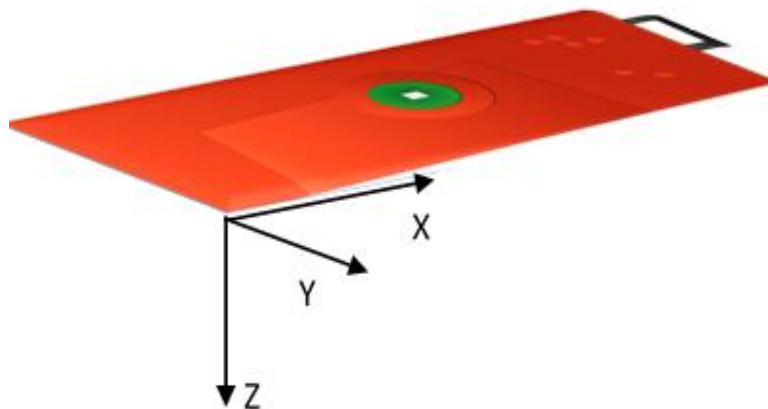


Figura 4-6 - Sistema de eixos definido para o SensorTag. Imagem do SensorTag obtida em [38].

Sensores	Transformações
Acelerómetro	$x_f = -y_i$ $y_f = x_i$ $z_f = z_i$
Giroscópio	$x_f = -x_i$ $y_f = -y_i$ $z_f = z_i$
Magnetómetro	$x_f = -x_i$ $y_f = y_i$ $z_f = z_i$

Tabela 4-1 - Transformações necessárias aos dados do SensorTag.

Outra conclusão retirada foi a necessidade de se inverter as saídas do acelerómetro, estabelecendo a sua orientação como a contrária do giroscópio e magnetómetro. Esta é uma prática comum para fazer com que o acelerómetro reporte 1g quando os seus eixos se encontram alinhados com o vetor de gravidade, simplificando os cálculos nos algoritmos [26]. Também, todos os algoritmos de fusão sensorial estudados recebem os dados do giroscópio em radianos por segundo, pelo que tem de haver uma conversão, uma vez que os dados do giroscópio são normalmente enviados em graus por segundo. Finalmente, os algoritmos baseados no método de Gauss-Newton recebem os dados do magnetómetro em Gauss, pelo que também tem de haver uma conversão, uma vez que os dados do magnetómetro são normalmente enviados em micro Tesla.

A máxima taxa de amostragem obtida com o SensorTag foi de 10Hz, no entanto esse valor não foi considerado como suficiente para a realização de testes que permitam estimar a orientação dos sensores. Também, o alcance fixo de ± 250 °/s do giroscópio foi visto como uma limitação para rotações mais rápidas, pelo que este satura com facilidade. Por essas razões foi encontrada outra solução por forma a melhorar a eficácia dos algoritmos desenvolvidos. Para o efeito foi utilizado o MPU-9150 da Invensense.

4.3 MPU-9150

O MPU-9150 é um dispositivo desenvolvido pela InvenSense que combina dois circuitos integrados no mesmo módulo: o MPU-6050, da InvenSense, e o AK8975 da Asahi Kasei Microdevices Corporation. O MPU-6050 é constituído por um acelerómetro triaxial, um giroscópio triaxial e uma solução de *hardware* denominada DMP (*Digital Motion Processor*); enquanto o AK8975 contém um magnetómetro triaxial [7]. O DMP é uma solução de *hardware* que implementa um algoritmo de fusão sensorial proprietário da InvenSense, processando os dados do acelerómetro e giroscópio, e devolvendo a orientação do sensor em formato de quaternião. Apesar de utilizar apenas o acelerómetro e o giroscópio, a estimativa do correspondente ao ângulo de Euler *yaw* apresenta resultados satisfatórios, devido aos algoritmos de deteção de movimento incluídos no DMP [7]. Estes analisam n amostras do giroscópio e concluem se o sinal recolhido é Gaussiano ou não. Se o sinal for determinado como

não Gaussiano conclui-se que o giroscópio está em movimento, por outro lado, se for determinado como Gaussiano, conclui-se que o giroscópio está imóvel [9]. A comunicação com os registos do dispositivo é feita através do protocolo *I2C (Inter-Integrated Circuit)* e para esse efeito foi utilizado um Arduino Leonardo para estabelecer a comunicação com uma placa de teste da SparkFun Electronics. Esta placa foi utilizada por permitir uma prototipagem rápida com o MPU-9150. As ligações necessárias encontram-se descritas na Figura 4-7, respeitando as ligações indicadas na Tabela 4-2.

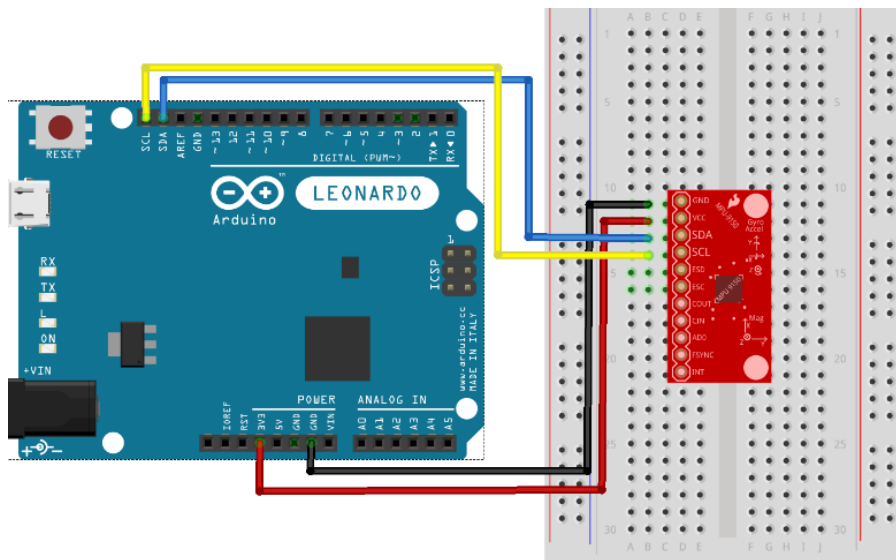


Figura 4-7 - Ligações entre Arduino Leonardo e a placa de teste MPU-9150.

MPU-9150	Arduino Leonardo
GND	GND
VCC	3.3V
SDA	SDA
SCL	SCL

Tabela 4-2 - Correspondências entre os pinos do MPU-9150 e entradas do Arduino Leonardo.

A utilização de um Arduino neste trabalho deve-se principalmente à existência de bibliotecas de *software* livre que garantem o acesso a vários dispositivos existentes no mercado. O MPU-9150 é um desses exemplos e por essa razão o código desenvolvido para configurar o sensor e aceder ao DMP foi baseado na biblioteca reportada em [39]. O MPU-9150 é configurado com uma frequência de amostragem de 40Hz, ou seja, a enviar dados de 25 em 25 milissegundos. Em termos de gama de valores foi estabelecido ± 2 g para o acelerómetro, ± 2000 °/s para o giroscópio e ± 1200 μ T para o magnetómetro. Os dados que chegam de cada sensor têm de ser convertidos para as unidades corretas, sendo essa conversão dada por (4.3). Em que n é o número de bits de resolução do sensor – 16 bits para o acelerómetro/giroscópio e 13 bits para o magnetómetro; e o *Range* o dobro do valor da gama – 4 para o acelerómetro, 4000 para o giroscópio e 2400 para o magnetómetro.

$$Unidade[\mu T, ^\circ/s, g] = \frac{Valor}{Escala} \quad (4.3)$$

$$Escala = \frac{2^n}{Range}$$

O conhecimento da orientação dos diversos sensores dentro da placa de circuito impresso permite definir o sistema de eixos a utilizar e de que forma os dados se vão relacionar com este. As transformações aplicadas aos dados dos sensores para que estes satisfaçam o referencial escolhido, Figura 4-8, encontram-se definidas na Tabela 4-3 em que x_i , y_i e z_i são os dados que chegam inicialmente de cada sensor, e x_f , y_f e z_f os dados depois da transformação.

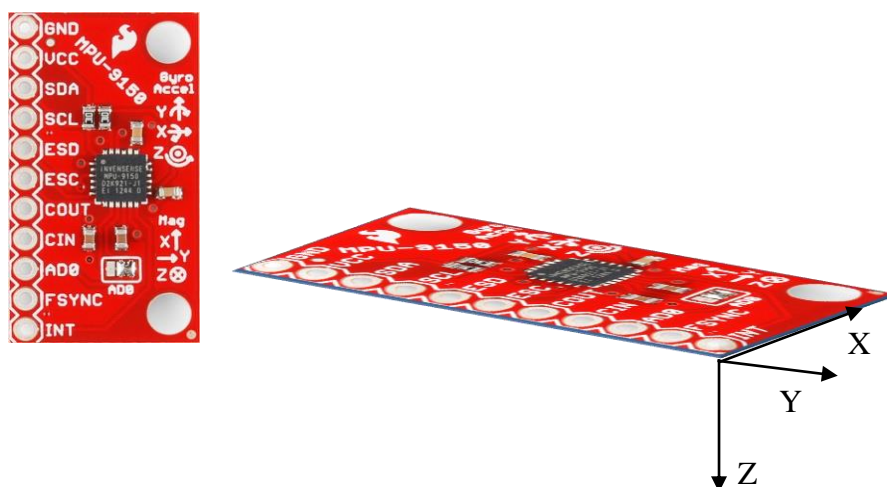


Figura 4-8 - Descrição da orientação dos sensores na placa de teste e referencial escolhido para o MPU-9150. Imagem retirada de [42]

Sensores	Transformações
Acelerómetro	$x_f = -x_i$ $y_f = y_i$ $z_f = z_i$
Giroscópio	$x_f = x_i$ $y_f = -y_i$ $z_f = -z_i$
Magnetómetro	$x_f = -x_i$ $y_f = y_i$ $z_f = z_i$

Tabela 4-3 - Transformações necessárias aos dados do MPU-9150.

Para além dos dados dos sensores chegam também dados do DMP que consistem em 4 valores, um para cada elemento do quaternião. Internamente ao DMP, os cálculos são feitos tendo como

referência a orientação do giroscópio, é portanto necessária uma transformação a estes valores [7]. Na Tabela 4-4 são descritas essas transformações em que q_{wi} , q_{xi} , q_{yi} e q_{zi} referem-se ao quaternião inicial, e q_{wf} , q_{xf} , q_{yf} e q_{zf} referem-se ao quaternião final.

Sensor	Transformações
DMP	$q_{wf} = q_{wi}$ $q_{xf} = q_{xi}$ $q_{yf} = -q_{yi}$ $q_{zf} = -q_{zi}$

Tabela 4-4 - Transformações de alinhamento dos dados do DMP com o referencial definido.

À semelhança da conversão feita para os dados dos sensores, também os dados do DMP necessitam de ser convertidos para as unidades corretas. Essa conversão é feita por (4.4), em que cada constituinte do quaternião tem uma resolução n de 32 bit e um *Range* de 4.

$$q[w, x, y, z] = \frac{Valor}{Escala} \quad (4.4)$$

$$Escala = \frac{2^n}{Range}$$

Para estabelecer a comunicação com o Arduino Leonardo e adquirir os dados do MPU-9150 em tempo-real foi desenvolvida uma aplicação para Windows com o Microsoft Visual Studio, nomeadamente uma Windows Forms Application em C#, por não ter sido possível obter do Matlab a performance desejada para os tempos de aquisição mais curtos conseguidos com esta interface – evolução de 10Hz (SensorTag/Matlab) para 40Hz (MPU-9150/Arduino). No desenvolvimento desta aplicação foi construída uma classe denominada por SensorFusion que reúne as funções necessárias à utilização de sensores inerciais, entre as quais: algoritmos de fusão sensorial, cálculos com quaterniões, cálculos da orientação com os dados individuais de cada sensor, remoção da gravidade nas leituras do acelerómetro, cálculo de velocidades e deslocamentos, correção da inclinação do magnetómetro e, finalmente, a avaliação do estado de movimento dos sensores baseado no algoritmo presente no DMP. Foi construída uma animação 3D através da biblioteca OpenTK. Esta biblioteca de baixo nível permite o acesso à API (*Application Programming Interface*) OpenGL que permite a construção de animações 2D e 3D [40]. As restantes funcionalidades da aplicação passam por disponibilizar em gráficos os dados dos sensores, os ângulos de Euler resultantes dos algoritmos implementados e as velocidades e posições calculadas. Também foram disponibilizadas caixas de texto onde é possível conferir o quaternião que chega do DMP, o quaternião calculado com os algoritmos e os respetivos ângulos de Euler. Por forma a testar os vários algoritmos e verificar os seus resultados, é disponibilizada uma lista na qual se pode selecionar o algoritmo de fusão sensorial que irá comandar a animação. A aplicação, Figura 4-9, permite ainda a gravação dos dados recolhidos num ficheiro de texto para posterior processamento.

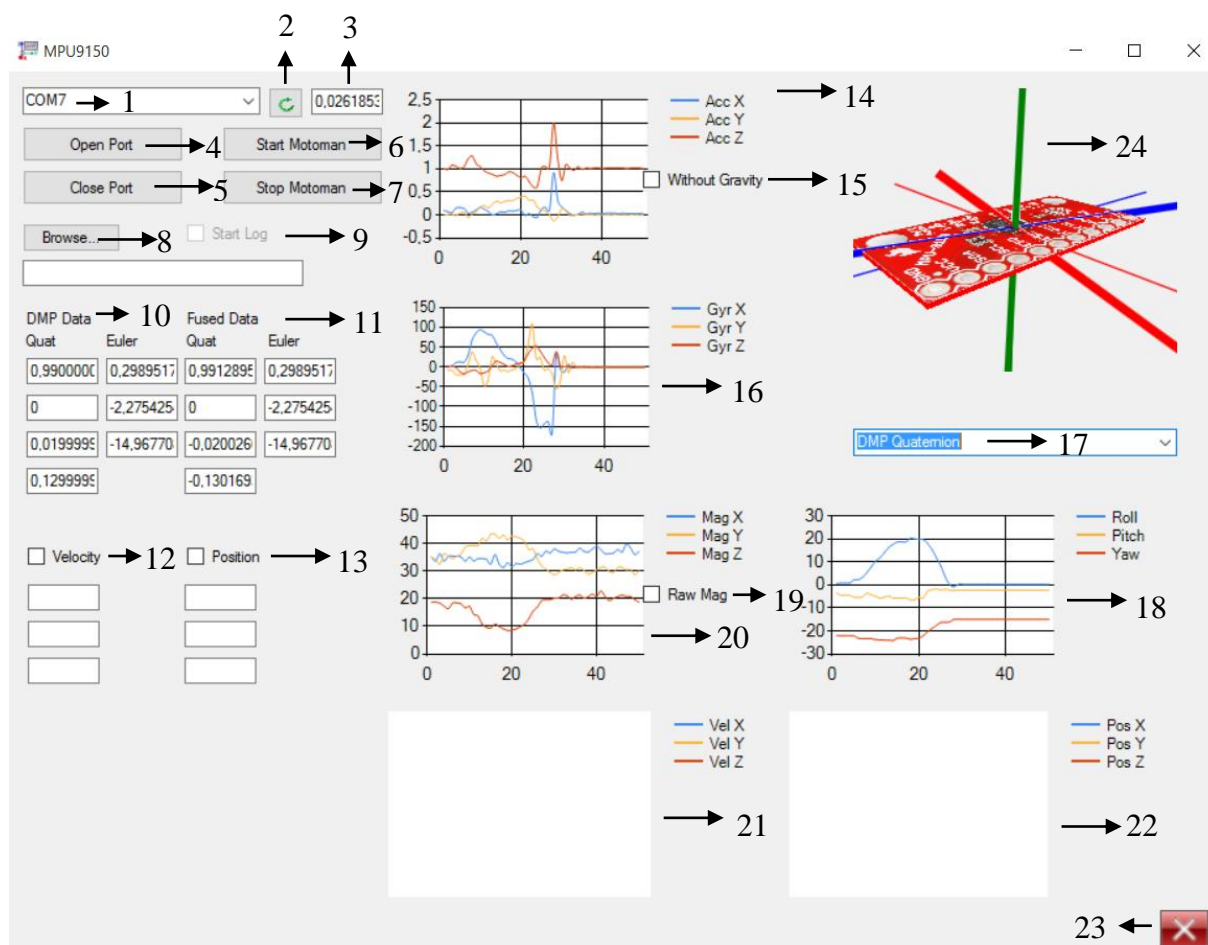


Figura 4-9 - Aplicação desenvolvida para receber dados da interface MPU-9150/Arduino.

1. Indicação das portas série disponíveis.
2. Botão para atualização das portas série.
3. Caixa de texto que indica o tempo passado entre cada notificação do Arduino.
4. Botão para inicializar porta série.
5. Botão para terminar a comunicação com a porta série.
6. Inicializar a comunicação com o robot.
7. Terminar comunicação com o robot.
8. Botão para criar o ficheiro de texto onde são guardados os dados.
9. Botão para inicializar a escrita para o ficheiro de texto.
10. Caixas de texto que indicam a informação que chega do DMP, nomeadamente o quaternião e os respetivos ângulos de Euler.
11. Caixas de texto que indicam o resultado do algoritmo de fusão sensorial selecionado, em formato de quaternião e os respetivos ângulos de Euler.
12. Botão para iniciar o cálculo da velocidade.
13. Botão para iniciar o cálculo da posição.
14. Gráfico com as leituras do acelerómetro. Sendo a azul as leituras no eixo X, a amarelo as do eixo Y e vermelho o eixo Z.

15. Botão que aplica o algoritmo de remoção das componentes de aceleração gravítica às leituras do acelerómetro. O resultado passa ser apresentado no gráfico de acelerações.
16. Gráfico com as leituras do giroscópio. Sendo a azul as leituras no eixo X, a amarelo as do eixo Y e vermelho o eixo Z.
17. Lista para seleccionar o algoritmo de fusão sensorial.
18. Gráfico com os ângulos de Euler resultantes do algoritmo de fusão sensorial seleccionado. Sendo a azul o ângulo *roll*, a amarelo o *pitch* e vermelho *yaw*.
19. Botão que define que devem ser usadas as leituras do magnetómetro sem calibração.
20. Gráfico com as leituras do magnetómetro. Sendo a azul as leituras no eixo X, a amarelo as do eixo Y e vermelho o eixo Z.
21. Gráfico com as velocidades calculadas. Sendo a azul as velocidades no eixo X, a amarelo as do eixo Y e vermelho o eixo Z.
22. Gráfico com as posições calculadas. Sendo a azul os deslocamentos no eixo X, a amarelo os do eixo Y e vermelho o eixo Z.
23. Botão para fechar a aplicação. Responde ao botão Esc do teclado.
24. Animação 3D que reproduz o movimento do MPU-9150. A vermelho encontram-se os eixos X, a azul os eixos Y e a verde os Z. Sendo que os eixos mais grossos indicam o referencial inercial, e os mais finos o do objeto.

Para possibilitar a utilização do MPU-9150 com o protocolo de comunicação BLE, foi desenvolvida uma solução que permite uma comunicação sem-fios baseada no SensorTag, mas aproveitando as capacidades do MPU-9150, nomeadamente o DMP.

4.4 PROTÓTIPO DESENVOLVIDO

O *hardware* desenvolvido neste trabalho foi baseado no SensorTag e reúne numa placa de circuito impresso o processador CC2541 da TI e o MPU-9150, alimentados com uma pilha CR2032. A placa tem dois botões – um de *Reset* e outro para efetuar o emparelhamento do dispositivo - possuindo ainda um conjunto de indicadores LED que indicam o estado da ligação e uma ficha de programação para atualizar o *firmware* do CC2541. Na Figura 4-10 é ilustrado o resultado final do protótipo desenvolvido.

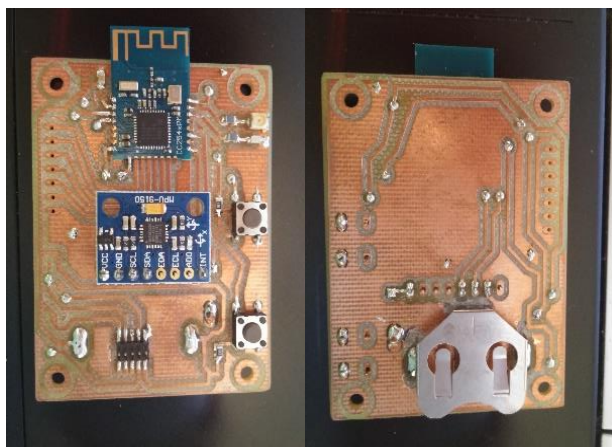


Figura 4-10 - Protótipo desenvolvido.

no SensorTag, por exemplo, em que os dados do acelerómetro, giroscópio e magnetómetro são enviados em pacotes de dados distintos.

Por sua vez o serviço DMP engloba as características DMP State Service, DMP Quat, DMP Sample Rate, DMP Enable e Mag Rate. A característica DMP State Service faz a declaração do serviço. O DMP Quat é a caraterística que guarda os dados do quaternião e que configura as notificações. O DMP Sample Rate permite a configuração da taxa de amostragem e o DMP Enable habilita o DMP. Como o DMP apenas utiliza dados do acelerómetro e giroscópio o magnetómetro tem que ser configurado em separado, por essa razão foi criada a característica Mag Rate.

Handle	Type	Mnemonic	Value	Description
1	0x2800	GATT Primary Service Declaration	00:18	Generic Access Service
2	0x2803	GATT Characteristic Declaration	02:03:00:00:2A	Device Name
4	0x2803	GATT Characteristic Declaration	02:05:00:01:2A	Appearance
6	0x2803	GATT Characteristic Declaration	0A:07:00:02:2A	Peripheral Privacy Flag
8	0x2803	GATT Characteristic Declaration	08:09:00:03:2A	Reconnection Address
10	0x2803	GATT Characteristic Declaration	02:08:00:04:2A	Peripheral Preferred Connection Parameters
12	0x2800	GATT Primary Service Declaration	01:18	Generic Attribute Service
16	0x2800	GATT Primary Service Declaration	0A:18	Device Information Service
35	0x2800	GATT Primary Service Declaration	F0:FF	Simple Profile Service
52	0x2800	GATT Primary Service Declaration	1A:BA	DMP Service
53	0x2803	GATT Characteristic Declaration	02:36:00:15:BA	DMP Status Service
54	0xBA15	DMP Status Service		MPUId:MAGId:DMPST
55	0x2901	Characteristic User Description		
56	0x2803	GATT Characteristic Declaration	12:39:00:16:BA	DMP QUAT
57	0xBA16	DMP QUAT	C6:40:00:2E:1D:66:F7:03:FF:E9:FF:D1:00:07:27:00:61:00:11:00	
58	0x2902	Client Characteristic Configuration		Write "01:00" to enable notifications, "00:00" to disable
59	0x2901	Characteristic User Description	44:69:67:69:74:61:6C:20:4D:6F:74:69:6F:6E:20:50:72:6F:63:6...	Digital Motion Process
60	0x2803	GATT Characteristic Declaration	0A:3D:00:18:BA	DMP Sample Rate
61	0xBA18	DMP Sample Rate	52	
62	0x2901	Characteristic User Description	44:6D:70:20:50:65:72:69:6F:64	Dmp Period
63	0x2803	GATT Characteristic Declaration	0A:40:00:17:BA	DMP enable
64	0xBA17	DMP enable	00	Write "01" to enable, "00" to disable
65	0x2901	Characteristic User Description		
66	0x2803	GATT Characteristic Declaration	0A:43:00:1C:BA	Mag Rate
67	0xBA1C	Mag Rate	44:6D:70:20:43:6F:6E:66:2E:20:45:6E:61:62:6C:65:2F:44:69:7...	Dmp Conf. Enable/Disab
68	0x2901	Characteristic User Description	4D:50:55:20:41:63:71:20:52:61:74:65	MPU Acq Rate
69			02:46:00:1D:BA	

Figura 4-12 - Inclusão do serviço DMP no protocolo BLE.

As funcionalidades da aplicação desenvolvida para comunicar com o protótipo desenvolvido são: guardar os dados recolhidos num ficheiro de texto, controlar uma animação 3D, escolher qual o algoritmo que controla a animação e ver os dados graficamente. Permite ainda fazer um registo dos pacotes de dados que são enviados e recebidos pelo CC2540 (baseado no Btool) e realizar algumas ações de configuração dos sensores. A classe SensorFusion desenvolvida foi também integrada nesta aplicação. Ainda foi integrada a capacidade de comunicar com o robot através de uma porta série, para obtenção das medidas de referência dos resultados dos algoritmos de fusão sensorial. A aplicação, Figura 4-13, foi desenvolvida com a capacidade de comunicar com o protótipo, estando este trabalhar com o serviço IMU ou o serviço DMP. Para ambos os casos existe uma zona de configuração do respetivo serviço, onde são configuradas as suas notificações, taxa de amostragem e estado ON/OFF.

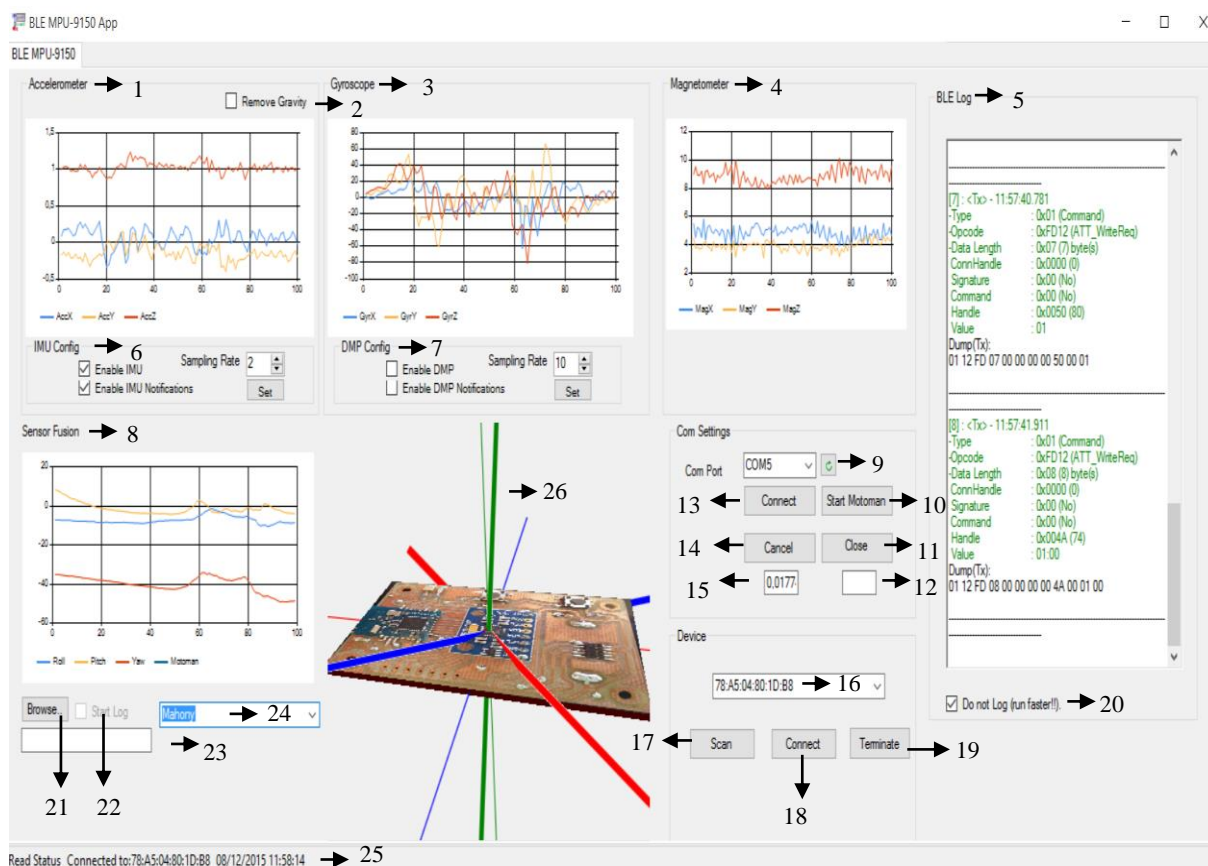


Figura 4-13 - Aplicação para receber dados do protótipo.

1. Gráfico com as leituras do acelerómetro, sendo a azul o eixo X, a amarelo o eixo Y e a vermelho o eixo Z.
2. Botão que aplica o algoritmo de remoção das componentes de aceleração gravítica nas leituras do acelerómetro.
3. Gráfico com as leituras do giroscópio, sendo a azul o eixo X, a amarelo o eixo Y e a vermelho o eixo Z.
4. Gráfico com as leituras do magnetómetro, sendo a azul o eixo X, a amarelo o eixo Y e a vermelho o eixo Z.
5. Caixa onde são mostrados os pacotes de dados enviados entre o PC e o protótipo. Sendo a verde os pacotes enviados pelo PC, e a azul os recebidos.
6. Zona de configuração do serviço IMU. Aqui são configuradas as notificações do IMU, a taxa de amostragem e o seu estado de ON/OFF.
7. Zona de configuração do serviço DMP. Aqui são configuradas as notificações do DMP, a taxa de amostragem e o seu estado de ON/OFF.
8. Gráfico com os resultados dos algoritmos de fusão sensorial. Sendo a azul-claro o ângulo *roll*, a amarelo o *pitch*, a vermelho o *yaw* e a azul-escuro a posição do robot.
9. Botão para atualizar as portas série disponíveis.
10. Botão para iniciar a comunicação com o robot.
11. Botão para terminar a comunicação com o robot.

12. Caixa de texto que mostra a notificação enviada pelo robot.
13. Botão para iniciar a comunicação com o CC2540 USB Evaluation Module Kit.
14. Botão para terminar a comunicação com o CC2540 USB Evaluation Module Kit.
15. Tempo decorrido entre cada receção de pacotes de dados.
16. Endereço do dispositivo BLE.
17. Procurar por dispositivos BLE.
18. Estabelecer comunicação com dispositivo BLE.
19. Terminar comunicação com dispositivo BLE.
20. Botão para desligar a apresentação dos pacotes enviados e recebidos.
21. Botão para criar o ficheiro de texto onde são guardados os dados.
22. Notão para iniciar a escrita para o ficheiro de dados.
23. Caminho do ficheiro de dados.
24. Escolha do algoritmo de fusão sensorial
25. Texto que indica o estado da ligação com o dispositivo BLE, o seu endereço e a data e hora.
26. Animação 3D que reproduz o movimento do MPU-9150. A vermelho encontram-se os eixos X, a azul os eixos Y e a verde os Z. Sendo que os eixos mais grossos indicam o referencial inercial, e os mais finos o do objeto.

O método adotado para a validação dos algoritmos desenvolvidos requer a utilização de um braço robótico que é programado para assumir orientações pré-definidas. De forma a sincronizar os dados dos sensores com os do robot, este comunica com a aplicação por porta série, funcionalidade para a qual a aplicação encontra-se também preparada.

4.5 MÉTODO DE OBTENÇÃO DE RESULTADOS

Por forma a validar os algoritmos de fusão sensorial estudados ao longo deste trabalho, foi utilizado um braço robótico, Figura 4-14, da família Motoman NX100 da Yaskawa para realizar movimentos pré-definidos e fornecer uma posição de referência para os resultados obtidos. O robot foi programado com recurso ao simulador MotoSim EG, um *software* para Windows que serve como uma ferramenta de ensino para robots da Motoman. Este permite a criação de um ficheiro denominado Job, que define as ações que o robot vai realizar por meio de instruções consecutivas, em que cada uma delas coloca o robot em determinada posição. Os Jobs criados foram editados para incluir as instruções de configuração da porta série e de envio de notificações uma vez que estas não se encontram disponíveis no simulador.



Figura 4-14 - Braço robótico da Motoman.

Assim que o robot atinge determinada posição, uma notificação é enviada por porta série para a aplicação que também recebe dados do MPU-9150. No momento em que essa notificação é recebida, a orientação do sensor é registada podendo assim ser comparada com o valor de referência do robot. A notificação enviada pelo robot consiste num byte que é incrementado à medida que cada instrução do robot é executada. Como se tratam de movimentos pré-definidos, a orientação do robot no fim de cada instrução é conhecida e pode ser feita uma correspondência entre o byte enviado e a orientação a que se encontra. A placa é colocada na garra do robot, isto porque, tanto a velocidade do centro da garra como a sua velocidade de rotação, são programadas por *software*. Esta localização permite realizar rotações em torno dos 3 eixos dos sensores assim como deslocamentos nessas direções. Uma das vantagens deste método é a repetibilidade de testes em que os movimentos são garantidamente os mesmos.

5 ANÁLISE COMPARATIVA DO COMPORTAMENTO DOS ALGORITMOS DE FUSÃO SENSORIAL

Com o objetivo de avaliar os resultados dos algoritmos de fusão sensorial foram realizados testes de orientação, secção 5.1, utilizando um braço robótico que permite colocar o protótipo desenvolvido em diversas posições distintas, fornecendo uma referência para comparação. Os testes consistiam em rodar a placa em torno dos seus três eixos de rotação, entre dois valores fixos de ângulos definidos com o robot. Foi também avaliada a capacidade dos algoritmos de corrigir uma posição inicial que não é a fornecida ao algoritmo, verificando quais os algoritmos mais rápidos e mais lentos a convergir para a posição correta, como também a capacidade dos algoritmos de recuperarem a sua orientação após submetidos a acelerações elevadas.

Um segundo conjunto de testes foi realizado com o intuito de avaliar o algoritmo de remoção das componentes de aceleração gravítica exposto na secção 2.3.2 e foram realizadas rotações em torno dos eixos X e Y do acelerómetro com o auxílio do robot, secção 5.2.

Estes testes foram realizados com o protótipo desenvolvido na secção 4.4 e com o auxílio da aplicação desenvolvida para receber os seus dados. Apesar de terem sido desenvolvidos dois serviços distintos, para funcionamento do MPU-9150 com o protocolo de comunicação BLE, em ambos os conjuntos de testes realizados foi utilizado o serviço IMU. Este serviço foi escolhido pois permite a configuração dos sensores inerciais fornecendo assim os dados para os algoritmos de fusão sensorial. A taxa de amostragem dos sensores definida foi de 50Hz, ou seja, de 20 em 20ms. No entanto, o valor de intervalo de tempo utilizado pelos algoritmos foi determinado com o auxílio de um cronómetro criado na aplicação, que determina o tempo (em segundos) que passa entre cada notificação BLE do protótipo. Também foi necessária a calibração do magnetómetro, sendo esta efetuada com o dispositivo colocado na garra do braço robótico para serem tidas em conta as interferências que os motores do robot têm nas leituras do magnetómetro.

Os algoritmos de fusão sensorial testados são: Gradiente Descendente, Filtro Complementar Explícito (AHRS), Filtro de Kalman Estendido com dupla fase de correção, Gauss-Newton (Filtro Complementar), Gauss-Newton (Filtro de Kalman), Filtro Complementar Explícito (IMU) e Filtro de Kalman Estendido. O algoritmo de remoção dos componentes de aceleração gravítica é baseado em quatérniões.

5.1 AVALIAÇÃO RELATIVA À PRESTAÇÃO DOS ALGORITMOS NA DETERMINAÇÃO DA ORIENTAÇÃO

Estes testes consistiam em fazer variar a orientação da placa entre dois valores distintos, 90° e -90°, registando a orientação dos sensores em orientações específicas ao longo do movimento – 0°, 45, 90°, -45 e -90°. No momento em que o robot atinge cada uma dessas posições ele envia uma notificação para a aplicação, que também se encontra a receber os dados dos sensores. Esta notificação é registada em conjunto com a posição angular indicada pelos algoritmos de fusão sensorial, podendo assim ser quantificado um desvio médio ao longo de todo o movimento

realizado. A notificação enviada é um byte que vai sendo incrementado sempre que o robot atinge cada uma das posições definidas. Como os movimentos são pré-definidos no simulador em computador, é feita uma correspondência entre o byte enviado e a respetiva posição.

Na Figura 5-1 ilustra-se o resultado para o caso em que o ângulo *roll* varia entre 90° e -90° com velocidade angular de $10^\circ/s$.

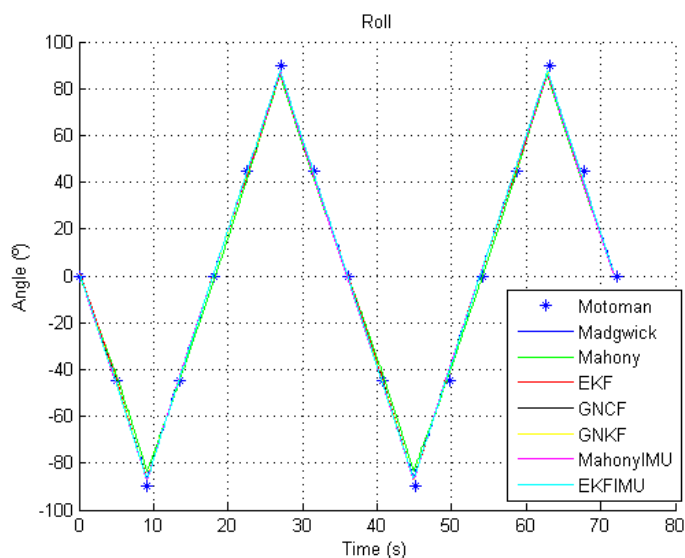


Figura 5-1 - Variação do ângulo *roll* em comparação com a variação da posição do robot.

Em que os pontos assinalados a azul-escuro indicam a posição do robot, a linha a azul-escuro indica o *roll* do algoritmo do Gradiente Descendente, a verde o resultado do algoritmo do Filtro Complementar Explícito para sistemas AHRS, a vermelho o Filtro de Kalman Estendido com dupla fase de correção, a preto o método de Gauss-Newton em parceria com o Filtro Complementar, a amarelo o método de Gauss-Newton em parceria com o Filtro de Kalman, a rosa o Filtro Complementar Explícito para IMU e a azul-claro o Filtro de Kalman Estendido. Na Tabela 5-1 são indicados os desvios médios (em graus), ao longo de todo o movimento, dos resultados para o ângulo *roll* com os algoritmos de fusão sensorial estudados ao longo deste trabalho.

Algoritmo	Desvio (°)
Gradiente Descendente	2.2385
Filtro Complementar Explícito (AHRS)	3.9531
Filtro de Kalman Estendido (Dupla Fase)	2.5976
Gauss-Newton (Filtro Complementar)	2.1814
Gauss-Newton (Filtro de Kalman)	2.1278
Filtro Complementar Explícito (IMU)	2.0353
Filtro de Kalman Estendido	2.1075

Tabela 5-1 - Desvio médio ao longo do movimento para o ângulo *roll*.

Consegue-se perceber que o algoritmo do Filtro Complementar Explícito (IMU) é o que tem o menor desvio médio, de 2.0353° (destacado a verde), enquanto o Filtro Complementar Explícito

para sistemas AHRS o mais elevado, com valor de 3.9531° (destacado a amarelo). A diferença de um filtro para o outro reside na inclusão das leituras do magnetómetro no algoritmo para sistemas AHRS, o que pode aumentar o desvio nas estimativas do ângulo *roll*. Ainda assim, muito próximos do algoritmo com menor desvio médio encontram-se os algoritmos de Gauss-Newton e o Filtro de Kalman Estendido.

Na Figura 5-2 ilustra-se o resultado para o caso em que o ângulo *pitch* varia entre 90° e -90° com velocidade angular $10^\circ/\text{s}$. Em que os pontos assinalados a azul-escuro indicam a posição do robot, a linha a azul-escuro o *pitch* do algoritmo do Gradiente Descendente, a verde o resultado do algoritmo do Filtro Complementar Explícito para sistemas AHRS, a vermelho o Filtro de Kalman Estendido com dupla fase de correção, a preto o método de Gauss-Newton em parceria com o Filtro Complementar, a amarelo o método de Gauss-Newton em parceria com o Filtro de Kalman, a rosa o Filtro Complementar Explícito para IMU e a azul-claro o Filtro de Kalman Estendido.

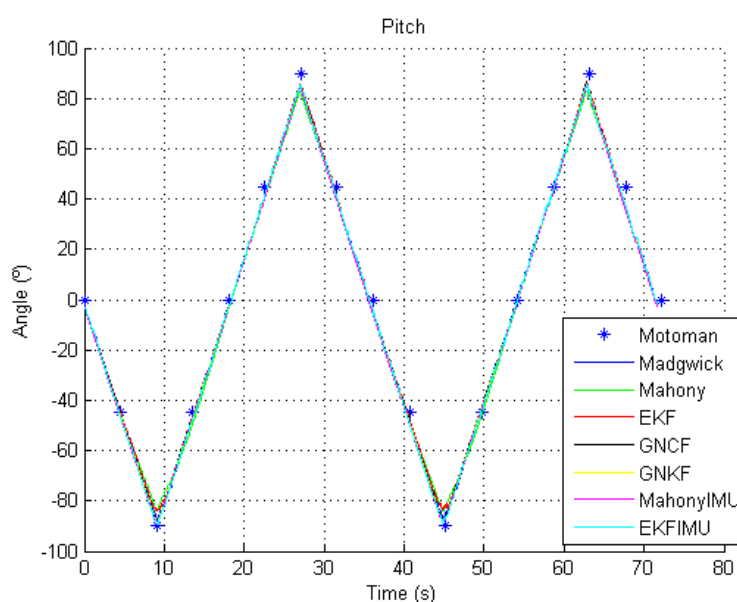


Figura 5-2 - Variação do ângulo *pitch* em comparação com a variação da posição do robot.

Na Tabela 5-2 são indicados os desvios médios (em graus) dos resultados para o ângulo *pitch* com os algoritmos de fusão sensorial estudados ao longo deste trabalho.

Algoritmo	Desvio (°)
Gradiente Descendente	3.5217
Filtro Complementar Explícito (AHRS)	4.8161
Filtro de Kalman Estendido (Dupla Fase)	3.6803
Gauss-Newton (Filtro Complementar)	3.3107
Gauss-Newton (Filtro de Kalman)	3.4959
Filtro Complementar Explícito (IMU)	3.8534
Filtro de Kalman Estendido	3.3303

Tabela 5-2 - Desvio médio ao longo do movimento para o ângulo *pitch*.

Consegue-se perceber que o método de Gauss-Newton (Filtro Complementar) é o que tem o menor desvio médio, 3.3107° (destacado a verde), enquanto o Filtro complementar Explícito (AHRS) apresenta o mais elevado, 4.8161° (destacado a amarelo). Perto do algoritmo com menor desvio médio encontra-se o algoritmo Filtro de Kalman Estendido, com valor de 3.3303° .

Na Figura 5-3 ilustra-se o resultado para o caso em que o ângulo *yaw* varia entre 90° e -90° com velocidade angular $20^\circ/s$. Em que os pontos assinalados a azul-escuro indicam a posição do robot, a linha a azul-escuro o *yaw* do algoritmo do Gradiente Descendente, a verde o resultado do algoritmo do Filtro Complementar Explícito para sistemas AHRS, a vermelho o Filtro de Kalman Estendido com dupla fase de correção, a preto o método de Gauss-Newton em parceria com o Filtro Complementar, a amarelo o método de Gauss-Newton em parceria com o Filtro de Kalman, a rosa o Filtro Complementar Explícito para IMU e a azul-claro o Filtro de Kalman Estendido.

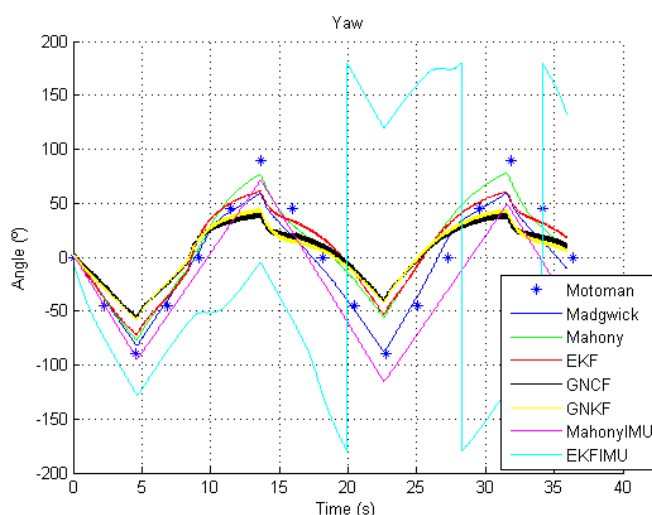


Figura 5-3 - Variação do ângulo *yaw* em comparação com a variação da posição do robot.

Na Tabela 5-3 são indicados os desvios médios (em graus) dos resultados para o ângulo *yaw* com os algoritmos de fusão sensorial estudados ao longo deste trabalho.

Algoritmo	Desvio ($^\circ$)
Gradiente Descendente	10.3265
Filtro Complementar Explícito (AHRS)	14.6532
Filtro de Kalman Estendido (Dupla Fase)	17.9846
Gauss-Newton (Filtro Complementar)	25.9522
Gauss-Newton (Filtro de Kalman)	24.3974
Filtro Complementar Explícito (IMU)	22.1535
Filtro de Kalman Estendido	126.6251

Tabela 5-3 - Desvio médio ao longo do movimento para o ângulo *yaw*.

Consegue-se perceber que o método do Gradiente Descendente é o que tem o menor desvio médio, 10.3265° (destacado a verde), enquanto o Filtro de Kalman Estendido apresenta o mais elevado, 126.6251° (destacado a cor de laranja). O desvio na estimativa do ângulo *yaw* é

superior aos desvios nas estimativas dos restantes ângulos de Euler. Atribui-se como razão o facto de o robot causar grandes interferências eletromagnéticas devido aos seus motores em funcionamento. Também, a calibração do magnetómetro representou um desafio. Para este tipo de calibração é necessário orientar os sensores no máximo de orientações possíveis e o robot não possui a liberdade suficiente de movimentos para atingir esse objetivo, sendo crucial esta ser feita com o dispositivo colocado no robot. O Filtro de Kalman Estendido apresenta um desvio médio tão elevado pela falta da referência do norte magnético indicada pelas leituras do magnetómetro. Por essa razão, a sua estimativa do ângulo *yaw*, apresenta uma deriva para o infinito pelo facto dela apenas ser estimada com recurso aos dados do giroscópio. As alterações entre -180° e 180° devem-se ao facto deste ângulo ter um intervalo entre -180° e 180° . O algoritmo do Filtro Complementar Explícito (IMU) também não inclui as leituras do magnetómetro, no entanto a deriva não é tão acentuada neste algoritmo, apesar de já se verificar um afastamento gradual em relação à linha de referência do robot. Dos algoritmos que possuem o Norte magnético como referência, o que apresenta desvio médio mais elevado é o método de Gauss-Newton (Filtro Complementar), com valor de 25.9522° (destacado a amarelo). Finalmente, as variações das estimativas não aparentam ter uma variação linear, como acontece para as estimativas *pitch* e *roll*, o que sugere a falta da calibração para os efeitos de *Soft-Iron*, nas leituras do magnetómetro.

Na Figura 5-4 é descrita a convergência das estimativas do ângulo *roll* para a orientação correta, partindo de uma posição inicial diferente da que é fornecida aos algoritmos (eixo X a apontar para Norte, Y para Este e Z para Baixo). A Figura 5-5 e Figura 5-6 indicam as variações dos ângulos *pitch* e *yaw*, respetivamente. Nos três casos, a linha a azul-escuro corresponde ao resultado do algoritmo do Gradiente Descendente, a verde o resultado do algoritmo do Filtro Complementar Explícito para sistemas AHRS, a vermelho o Filtro de Kalman Estendido com dupla fase de correção, a preto o método de Gauss-Newton em parceria com o Filtro Complementar, a amarelo o método de Gauss-Newton em parceria com o Filtro de Kalman, a rosa o Filtro Complementar Explícito para IMU e a azul-claro o Filtro de Kalman Estendido.

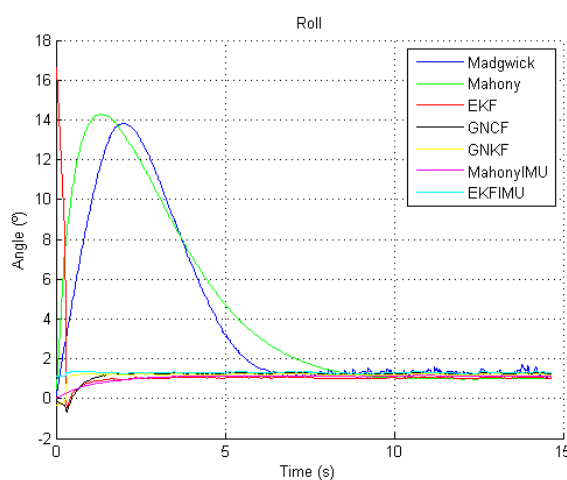


Figura 5-4 - Convergência das estimativas do ângulo *roll* partindo de uma posição inicial diferente.

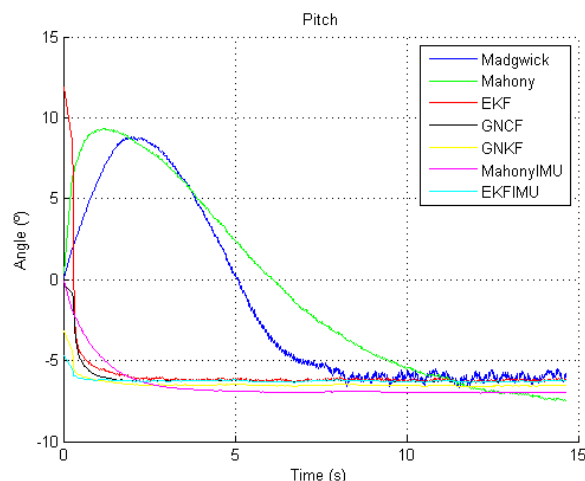


Figura 5-5 - Convergência das estimativas do ângulo *pitch* partindo de uma posição inicial diferente.

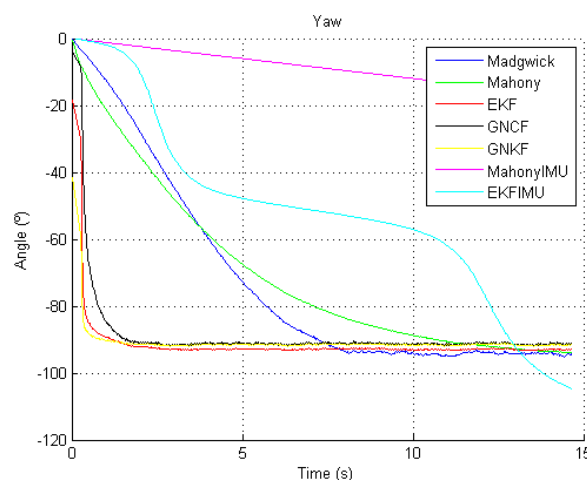


Figura 5-6 - Convergência das estimativas do ângulo *yaw* partindo de uma posição inicial diferente.

Verifica-se que os algoritmos que mais tempo demoram a convergir para a posição inicial correta são o método Gradiente Descendente e o Filtro Complementar Explícito para sistemas AHRS. Pelo contrário, os algoritmos com base em Filtros de Kalman são os mais rápidos a convergir, por possuírem um ganho adaptativo e variável. Na estimativa do ângulo *yaw* os algoritmos do Filtro Complementar Explícito (IMU) e o Filtro de Kalman Estendido não convergem para uma posição inicial porque não têm a referência do norte magnético. Apresentam apenas uma deriva da sua posição inicial pelo que não foram tidos em conta para este caso específico.

A Figura 5-7 apresenta a resposta dos algoritmos de fusão sensorial, para a estimativa *roll*, quando sujeitos a elevadas acelerações, indicando quais os algoritmos mais afetados por essas perturbações e o tempo que demoram a recuperar a sua orientação. Em que a linha a azul-escuro corresponde ao *roll* do algoritmo do Gradiente Descendente, a verde o resultado do algoritmo do Filtro Complementar Explícito para sistemas AHRS, a vermelho o Filtro de Kalman

Estendido com dupla fase de correção, a preto o método de Gauss-Newton em parceria com o Filtro Complementar, a amarelo o método de Gauss-Newton em parceria com o Filtro de Kalman, a rosa o Filtro Complementar Explícito para IMU e a azul-claro o Filtro de Kalman Estendido.

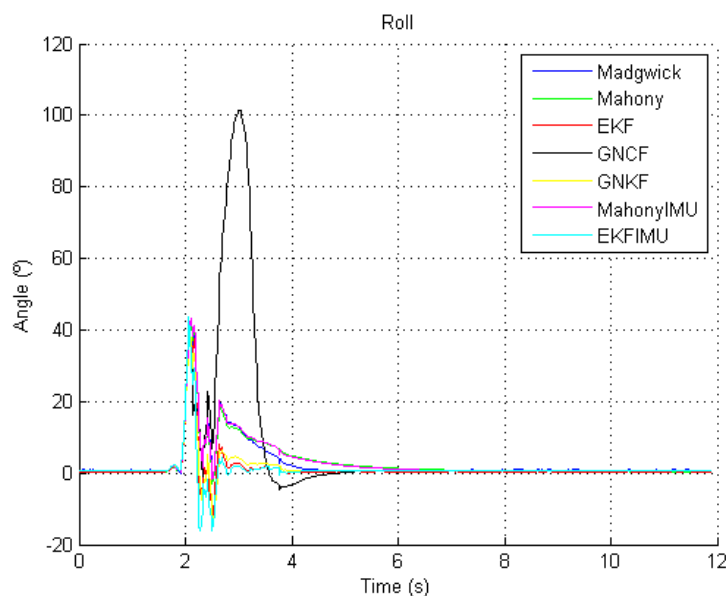


Figura 5-7 - Resposta dos algoritmos quando sujeitos a elevadas acelerações.

Verifica-se que o algoritmo que é mais afetado por acelerações lineares elevadas é o método de Gauss-Newton (Filtro Complementar). A ação do Filtro Complementar, uma vez que não possui um ganho variável, afeta o tempo de resposta do algoritmo. Apesar de todos sofrerem consideravelmente com a influência destas acelerações, algoritmos como o método do Gradiente Descendente e Filtro Complementar Explícito demoram algum tempo a recuperar a informação correta de posição. Por outro lado, algoritmos baseados em filtros de Kalman são os mais rápidos a produzir resultados.

5.2 AVALIAÇÃO DA PRESTAÇÃO DO ALGORITMO DE REMOÇÃO DA GRAVIDADE

Estes testes consistiam em realizar rotações entre 45° e -45° a uma velocidade angular de $10^\circ/\text{s}$ em torno dos eixos X e Y do acelerómetro. Foi imposto um movimento lento ao robot para garantir que as acelerações lineares do acelerómetro fossem próximas de zero, ou seja, as leituras do acelerómetro são predominantemente acelerações gravíticas. Assim, uma remoção da gravidade com sucesso é indicada pelas leituras dos três eixos do acelerómetro próximas de zero.

A Figura 5-8 ilustra o caso em que foram realizadas rotações entre 45° e -45° em torno do eixo X do sensor, situação que vai alterando as componentes gravíticas nos eixos Y e Z (azul). Também é visível a aceleração linear do sensor próxima de zero, sugerindo uma atenuação das componentes gravíticas com sucesso (vermelho).

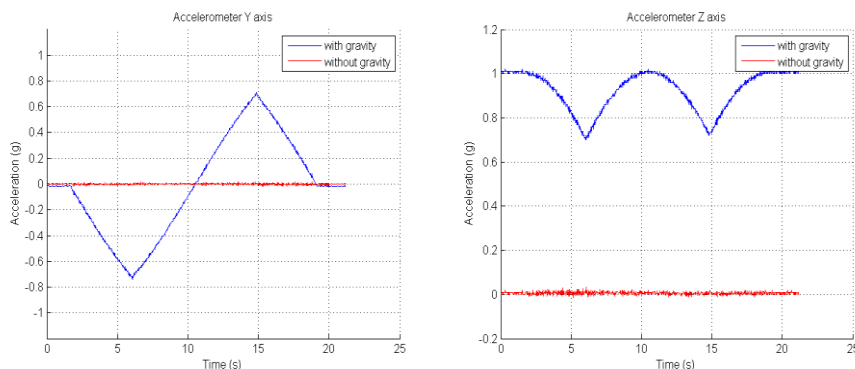


Figura 5-8 - Na esquerda é representada a variação da aceleração no eixo Y. Na direita, variação da aceleração no eixo Z.

Na Figura 5-9 ilustra-se a variação da posição angular dos sensores ao longo de todo o movimento realizado, com *roll* a azul, *pitch* a verde e *yaw* a vermelho.

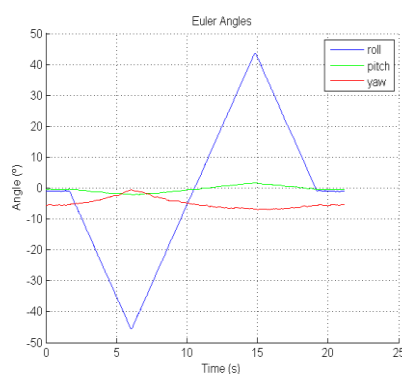


Figura 5-9 – Variação da posição angular dos sensores durante rotação em torno do eixo X.

A Figura 5-10 ilustra o caso em que foram realizadas rotações entre 45° e -45° em torno do eixo Y do sensor, situação que vai alterando as componentes gravíticas nos eixos X e Z (azul). Também é visível a aceleração linear do sensor próxima de zero, sugerindo uma atenuação das componentes gravíticas com sucesso (vermelho).

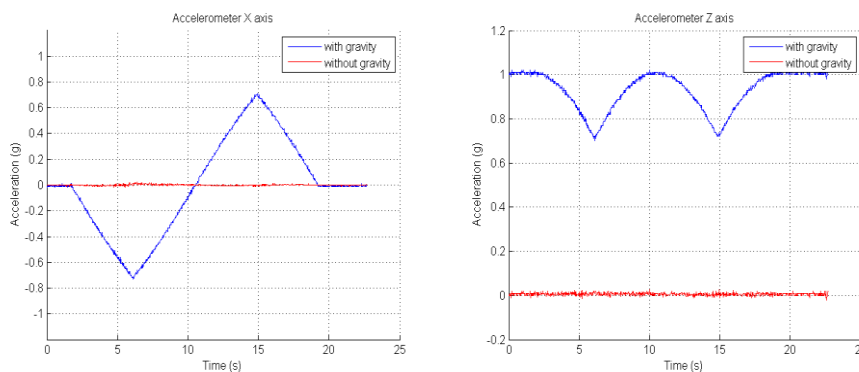


Figura 5-10 - Na esquerda é representada a variação da aceleração no eixo X. Na direita, variação da aceleração no eixo Z.

Na Figura 5-11 ilustra-se a variação da posição angular dos sensores ao longo de todo o movimento realizado, com *roll* a azul, *pitch* a verde e *yaw* a vermelho.

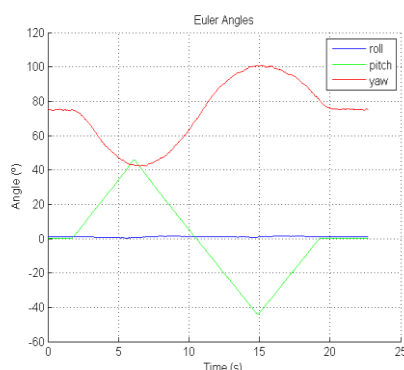


Figura 5-11 - Variação da posição angular dos sensores durante rotação em torno do eixo Y.

Para ambos os testes apresentados foi determinado o desvio na magnitude do vetor de gravidade removido, que deve ser igual a 1g. Foi calculado o vetor de gravidade removido sabendo a magnitude da aceleração obtida nas leituras do acelerómetro e as acelerações obtidas com a aplicação do algoritmo. Essas informações encontram-se disponíveis na Tabela 5-4.

Ângulo de Euler	Magnitude do Vetor de Gravidade Removido (g)	Desvio (%)
<i>Roll</i>	0.9965	0.3532
<i>Pitch</i>	0.9954	0.4557

Tabela 5-4 - Desvio na remoção as componentes de aceleração gravítica.

Um último teste foi realizado com o intuito de avaliar a eficácia do algoritmo quando o acelerómetro se encontra sujeito a acelerações lineares. Neste caso o dispositivo deslocou-se 15 cm no seu eixo Y e regressou à sua posição original, com uma velocidade de 93 mm/s. Na Figura 5-12 é representada a variação da orientação dos sensores em todo o movimento, que sugere que a placa não se encontra na horizontal. Este facto também é visível no gráfico da aceleração no eixo Y, em que as leituras do acelerómetro nesse eixo apresentam um desvio perto de 0.15 g, correspondente a aceleração gravítica. Nesse mesmo gráfico é visível uma remoção das componentes gravíticas com eficácia e sem atenuar acelerações lineares.

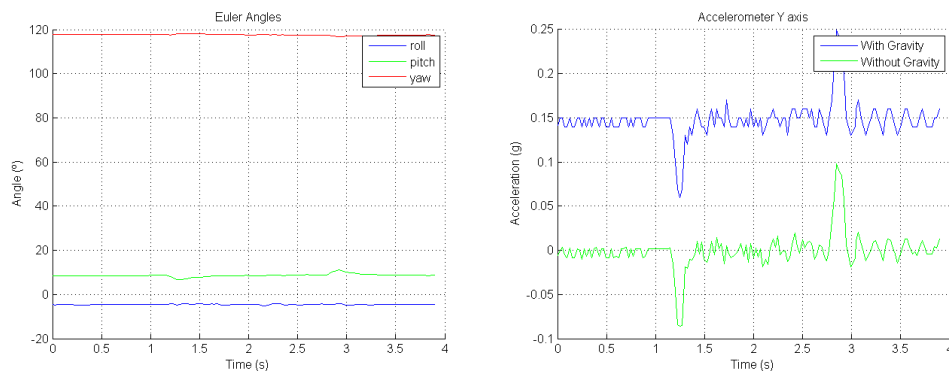


Figura 5-12 – Na esquerda é representada a variação da posição angular dos sensores (roll a azul, pitch a verde e yaw a vermelho). Na direita é exibida a aceleração do eixo Y com gravidade (azul) e sem gravidade (verde).

6 CONCLUSÃO

Esta dissertação teve como base o conceito de fusão sensorial aplicado a sensores inerciais, uma vez que estes cada vez mais são características imprescindíveis em dispositivos eletrônicos utilizados no dia-a-dia de milhões de pessoas – *smartphones* e *tablets*. Por outro lado, existe uma tendência crescente na aplicação deste tipo de tecnologia em áreas como a saúde ou o desporto.

O presente trabalho teve como objetivo comparar vários algoritmos de fusão sensorial e averiguar quais os mais eficazes. Para tal, foi proposto um método que consiste na utilização de um braço robótico para realizar movimentos pré-definidos e fornecer uma referência para comparação. Os testes foram realizados com um dispositivo que permite ao MPU-9150 da Invensense, comunicar através do protocolo de comunicação BLE com uma aplicação para Windows, dispositivo esse desenvolvido no decorrer deste trabalho. Em fases preliminares o MPU-9150 foi utilizado conjuntamente com uma plataforma Arduino para análise do seu funcionamento e foi usado o SensorTag, da Texas Instruments, para estudo do protocolo BLE.

Esta dissertação inclui uma investigação sobre sensores inerciais, meios de representação da orientação de um objeto no espaço tridimensional e alguns dos algoritmos de fusão sensorial mais utilizados atualmente. É ainda estudado um algoritmo de remoção das componentes de aceleração gravítica nas leituras do acelerómetro.

Toda a investigação realizada no decorrer desta dissertação serve como base à utilização dos sensores inerciais para estimar a orientação de um objeto no espaço tridimensional. Tendo como destinatários futuros utilizadores desta tecnologia, são documentados neste trabalho, através dos métodos de validação descritos, quais os algoritmos de fusão sensorial mais fiáveis e quais possuem menor carga computacional (caracterizada pelo menor número de operações matemáticas exigidas). São também descritos passo a passo todos os algoritmos de fusão sensorial, a informação que pode ser extraída dos sensores inerciais e os métodos de definição da orientação de um objeto no espaço tridimensional. Pelo que esta dissertação serve como um ponto de partida à utilização desta tecnologia.

6.1 PRINCIPAIS CONCLUSÕES

No âmbito desta dissertação foram realizados dois conjuntos distintos de testes: testes de orientação e testes de remoção da gravidade nas leituras do acelerómetro. Em ambos foi utilizado o dispositivo desenvolvido no decorrer deste trabalho, que consiste na utilização do MPU-9150 da Invensense a comunicar por BLE com uma aplicação para Windows.

O primeiro conjunto de testes permitiu concluir quais os algoritmos mais eficazes a estimar a orientação dos sensores durante os movimentos realizados com o braço robótico, quais os mais rápidos a convergir para a orientação correta partindo de uma posição inicial diferente do que a fornecida aos algoritmos e quais os mais rápidos a recuperar após submetidos a acelerações lineares elevadas. Na Tabela 6-1 ilustra-se um resumo dos resultados dos primeiros testes de orientação realizados, nomeadamente, os obtidos com os movimentos de rotação concretizados

pelo robot em torno dos três eixos de rotação dos sensores. São indicados quais os algoritmos com menor e maior desvio médio (em graus) para cada estimativa determinada: ângulo *roll*, *pitch* e *yaw*, e os respectivos desvios.

Ângulo de Euler	Menor Erro	Desvio (°)	Maior Erro	Desvio (°)
<i>Roll</i>	Filtro Complementar Explícito (IMU)	2.0353	Filtro Complementar Explícito (AHRS)	3.9531
<i>Pitch</i>	Gauss-Newton (Filtro Complementar)	3.3107	Filtro Complementar Explícito (AHRS)	4.8161
<i>Yaw</i>	Gradiente Descendente	10.3265	Gauss-Newton (Filtro Complementar)	25.9522

Tabela 6-1 - Síntese dos resultados para os testes de orientação.

Para o teste do ângulo *roll* foi determinado que o algoritmo mais eficaz foi o Filtro Complementar Explícito (IMU) e o de maior desvio médio o Filtro Complementar Explícito (AHRS). A diferença entre ambos os algoritmos reside apenas na inclusão do magnetômetro no algoritmo para sistemas AHRS. Neste algoritmo os dados de campo magnético afetam também as estimativas do ângulo *roll* e *pitch*, e daí a diferença de 2 graus no desvio médio de cada algoritmo. É importante mencionar que os algoritmos baseados no método de Gauss-Newton e o algoritmo do Filtro de Kalman Estendido apresentam erros médios próximos do apresentado pelo Filtro de Complementar Explícito (IMU).

Para o teste do ângulo *pitch* foi identificado como o algoritmo de menor desvio médio o método de Gauss-Newton (Filtro Complementar) e o de maior desvio médio o Filtro Complementar Explícito (AHRS), verificando-se um aumento no desvio médio em relação aos resultados para o ângulo *roll*. O Filtro Complementar Explícito (AHRS) volta a ser determinado como o algoritmo de maior desvio médio. O algoritmo do Filtro de Kalman Estendido apresentou um desvio médio próximo do menor obtido neste teste.

O teste relativamente ao ângulo *yaw* indica como algoritmo de menor desvio médio o método do Gradiente Descendente, e o de maior desvio médio o método de Gauss-Newton (Filtro Complementar). Na realidade, o Filtro de Kalman Estendido apresenta um desvio superior, no entanto este deve-se à deriva para o infinito que a sua estimativa sofre, por esta apenas ser calculada com recurso ao giroscópio. Este teste apresentou um desafio que inicialmente não era esperado, nomeadamente na calibração do magnetômetro. O funcionamento dos motores do braço robótico causa interferências nas leituras do magnetômetro e era importante a sua calibração na presença destas interferências. A calibração do magnetômetro é caracterizada pela

necessidade de o orientar no máximo de orientações possíveis, pelo que foi um desafio obter o número de orientações necessárias com o robot. No entanto, uma rotina de calibração foi realizada tendo em vista obter os melhores resultados possíveis. Também, a variação do ângulo *yaw* para a maior parte dos algoritmos não apresenta a linearidade apresentada para os restantes ângulos de Euler. Considera-se que este facto deve-se à necessidade de calibração do magnetómetro para os efeitos de *Soft-Iron*, que não foi realizado no âmbito deste trabalho. No entanto a variação do Filtro Complementar Explícito (IMU) apresenta uma variação linear, isto porque a sua estimativa é determinada exclusivamente com o giroscópio, apresentando ainda um afastamento gradual em relação ao movimento definido pelo robot, mas não tão acentuado como o apresentado pelo Filtro de Kalman Estendido.

Na Tabela 6-2 são identificados os algoritmos que apresentam melhor resposta quando sujeitos a uma posição inicial diferente da que lhes é fornecida e quando os sensores são sujeitos a elevadas acelerações.

Teste	Mais rápidos	Mais lentos
Posição inicial diferente	Algoritmos baseados em Filtros de Kalman	Gradiente Descendente e Filtro Complementar Explícito (AHRS e IMU)
Elevadas acelerações	Algoritmos baseados em Filtros de Kalman	Gauss-Newton (Filtro Complementar), Gradiente Descendente e Filtro Complementar Explícito (AHRS e IMU)

Tabela 6-2 - Resultados para os testes de resposta dos algoritmos.

Verifica-se que em ambos os testes todos os algoritmos baseados em Filtros de Kalman são rápidos a responder e a convergir para a correta. Isto é explicado pelo facto do Filtro de Kalman apresentar um ganho variável e ajustável às características das leituras do sensor. Por outro lado, os algoritmos Gradiente Descendente e ambos do Filtro Complementar Explícito são os mais lentos. Quando sujeitos a elevadas acelerações, o método de Gauss-Newton baseado no Filtro Complementar apresenta um erro muito grande em comparação com os restantes algoritmos, e por consequência demora um tempo considerável a recuperar a orientação inicial. Ao contrário do que acontece para o Filtro de Kalman, o Filtro Complementar apresenta um ganho fixo e não ajustável.

Por fim, foram realizados os testes de remoção das componentes de aceleração gravítica. O algoritmo utilizado para estimar a orientação dos sensores foi o do Filtro de Kalman Estendido com dupla fase de correção. Este apresentou um comportamento razoável para os testes de orientação, entenda-se por razoável o facto de nunca ter sido identificado como o algoritmo menos eficaz nem o mais eficaz, sendo ainda um dos algoritmos mais rápidos na sua resposta quando sujeitos a acelerações lineares. Foram realizados testes em que as leituras do acelerómetro eram predominantemente acelerações gravíticas, obtendo assim acelerações próximas de zero quando aplicado o algoritmo em questão. Foram obtidos desvios no vetor de

gravidade removido na ordem dos 0.3532% e 0.4557% para as rotações em torno do eixo X e Y, respetivamente. Também foi realizado um teste já na presença de acelerações lineares com o intuito de se remover as acelerações gravíticas mantendo as lineares o que mais uma vez se verificou.

No decorrer deste trabalho, nomeadamente na fase de estudo dos algoritmos de fusão sensorial, foi feito um levantamento do número de operações matemáticas necessárias para a implementação de cada um dos algoritmos. Os resultados encontram-se resumidos na Tabela 6-3.

Algoritmo	Somas	Subtrações	Multiplicações	Divisões	Raízes Quadradas
Gradiente Descendente	82	87	208	4	5
Filtro Complementar Explícito (AHRs)	43	29	102	3	4
Filtro de Kalman Estendido (Dupla Fase)	726	47	1059	15	4
Gauss-Newton (Filtro Complementar)	64	159	955	38	14
Gauss-Newton (Filtro de Kalman)	294	150	1417	30	12
Filtro Complementar Explícito (IMU)	23	11	58	2	2
Filtro de Kalman Estendido	422	13	573	11	2

Tabela 6-3 - Levantamento do número de operações matemáticas necessárias à implementação dos algoritmos de fusão sensorial.

Conclui-se que os algoritmos que mais operações matemáticas necessita é o método de Gauss-Newton (Filtro de Kalman) (destacado a amarelo). Isto deve-se maioritariamente ao facto do método de Gauss-Newton possuir uma etapa de otimização com 3 passos, triplicando o número de operações necessárias em cada um desses passos. Por outro lado, o algoritmo que necessita menos operações é o Filtro Complementar Explícito (IMU) (destacado a verde).

6.2 DESENVOLVIMENTOS FUTUROS

Ao longo da realização deste trabalho foram surgindo algumas sugestões para futuras contribuições no estudo de algoritmos de fusão sensorial, nomeadamente:

- Estudar a calibração para efeitos de *Soft-Iron* do magnetómetro.
- Considerar os efeitos de *Soft-Iron* na informação proveniente do magnetómetro e verificar as diferenças nos resultados realizando os mesmos testes.
- Estudar o impacto da variação da carga da pilha nos efeitos de *Hard-Iron* do magnetómetro, assim como a orientação desta no protótipo.
- Estudar um novo método, que não inclua o braço robótico, para a validação das estimativas do ângulo *yaw*.
- Estudar a carga computacional imposta que os algoritmos de fusão sensorial considerados impõem aos microprocessadores.
- Estudar o impacto do envio em pacotes separados dos dados dos diversos sensores na eficácia dos algoritmos.
- Reunir na mesma versão do protocolo BLE os serviços IMU e DMP desenvolvidos em separado no decorrer deste trabalho.
- Averiguar se um sistema de câmaras de vídeo, que faz a monitorização de movimentos mediante técnicas de processamento de imagem, pode ser utilizado para validar os algoritmos desenvolvidos. O sistema indicado já existe e é uma possibilidade a realização destes testes.

Finalmente, a utilização do DMP implica um procedimento de engenharia inversa relativamente ao código fornecido pela Invensense. Este processo foi iniciado mas ainda não produziu os resultados desejados, pelo que a resolução deste processo será uma das primeiras tarefas a ser realizada futuramente. O objetivo é utilizar o método desenvolvido neste trabalho para validar a orientação que o DMP determina.

REFERÊNCIAS BIBLIOGRÁFICAS

- [1] F. Abyarjoo, A. Barreto, J. Cofino e F. R. Ortega, "Implementing a Sensor Fusion Algorithm for 3D Orientation Detection with Inertial/Magnetic Sensors," *Innovations and Advances in Computing, Informatics, Systems Sciences, Networking and Engineering Lecture Notes in Electrical Engineering*, vol. Volume 313, 2015.
- [2] S. O. H. Madgwick, A. J. L. Harrison e R. Vaidyanathan, "Estimation of IMU and MARG Orientation using a Gradient Descent Algorithm," *International Conference on Rehabilitation Robotics*, 2011.
- [3] Z.-Q. Zhang, L.-Y. Ji, Z.-P. Huang e J.-K. Wu, "Adaptive Information Fusion for Human Upper Limb Movement Estimation," *IEEE Transactions IEEE Transactions on Systems - Part A: Systems and Humans*, Vol. 42, No. 5, 2012.
- [4] Z. Zhang, Z. Huang e J. Wu, "Hierarchical Information Fusion for Human Upper Limb Motion Capture," *12th International Conference on Information Fusion*, 2009.
- [5] G. Panahandeh, I. Skog e M. Jansson, "Calibration of the Accelerometer Triad of an Inertial Measurement Unit, Maximum Likelihood Estimation and Cramér-Rao Bound," *International Conference on Indoor Positioning and Indoor Navigation*, 2010.
- [6] R. Mahony, T. Hamel e J.-M. Pflimlin, "Nonlinear Complementary Filters on the Special Orthogonal Group," *Transactions on Automatic Control*, 2008.
- [7] Invensense, *MPU-9150 Product Specification*, 2012.
- [8] C.-Y. Lee e J.-J. Lee, "Estimation of Walking Behaviour Using Accelerometers in Gait Rehabilitation," *International Journal of Human-friendly Welfare Robotic Systems*, 2002.
- [9] P.-C. Chung, Y.-L. Hsu, C.-Y. Wang, C.-W. Lin e J.-S. W. M.-C. Pai, "Gait Analysis for Patients with Alzheimer's Disease Using A Triaxial Accelerometer," *IEEE*, 2012.
- [10] A. Salarian, H. Russmann, J. G. Vingerhoets, C. Dehollain, Y. Blanc, P. R. Burkhard e K. Aminian, "Gait Assessment in Parkinson's Disease: Toward an Ambulatory System for Long-Term Monitoring," *IEEE Transactions on Biomedical Engineering*, 2004.
- [11] J. Stamatakis, J. Crémers, D. Maquet, B. Macq e G. Garraux, "Gait feature extraction in Parkinson's disease using low-cost accelerometers," *33rd Annual International Conference of the IEEE EMBS*, 2011.

- [12] S. Sabatelli, M. Galgani, L. Fanucci e A. Rocchi, "A Double-Stage Kalman Filter for Orientation Tracking With an Integrated Processor in 9-D IMU," *Instrumentation and Measurement, IEEE Transactions*, 2013.
- [13] S. Sabatelli, F. Sechi, L. Fanucci e A. Rocchi, "A sensor fusion algorithm for an integrated angular position estimation with inertial measurement unit," *Design, Automation & Test in Europe Conference & Exhibition*, 2011.
- [14] "AN-1005 Understanding Euler Angles," 2012.
- [15] J. B. Kuipers, *Quaternions and Rotation Sequences: A Primer with Applications to Orbits, Aerospace and Virtual Reality*, Princeton University Press, 1999.
- [16] J. R. Beaty, "Introduction to the Theory and Application of Quaternions Part 1 of 2 ("Introduction to Quaternions")," 2012.
- [17] M. Pedley, "AN3461 Tilt Sensing Using a Three-Axis Accelerometer," 2013.
- [18] R. Burchfield e S. Venkatesan, "A Framework for Golf Training Using Low-Cost Inertial Sensors," *International Conference on Body Sensor Networks*, 2010.
- [19] P.-G. Jung, G. Lim e K. Kong, "Human Posture Measurement in a Three-Dimensional Space Based on Inertial Sensors," *12th International Conference on Control, Automation and Systems*, 2012.
- [20] C. Robotics, "AN-1008 Sensors for Orientation Estimation," 2012.
- [21] K. Seifert e O. Camacho, "AN3397 Implementing Positioning Algorithms Using Accelerometers," 2007.
- [22] C. Robotics, "AN-1007 Estimating Velocity and Position Using Accelerometers," 2012.
- [23] G. To e M. R. Mahfouz, "Design of Wireless Inertial Trackers for Human Joint Motion Analysis," *Biomedical Wireless Technologies, Networks, and Sensing Systems*, 2012.
- [24] J. R. Beaty, "Introduction to the Theory and Application of Quaternions Part 2 of 2 ("Working with Quaternions")," 2012.
- [25] T. Ozyagcilar, "AN4246 Calibrating an eCompass in the Presence of Hard and Soft-Iron Interference," 2013.
- [26] T. Ozyagcilar, "AN4248 Implementing a Tilt-Compensated eCompass using Accelerometer and Magnetometer Sensors," 2012.

-
- [27] Wikipedia, https://en.wikipedia.org/wiki/Gimbal_lock, 2015.
- [28] C. Robotics, "AN-1006 Understanding Quaternions," 2012.
- [29] M. Euston, P. Coote, R. Mahony, J. Kim e T. Hamel, "A Complementary Filter for Attitude Estimation of a Fixed-Wing UAV," *International Conference on Intelligent Robots and Systems*, 2008.
- [30] D. Comoti, "Orientation Estimation Based on Gauss-Newton Method and Implementation of a Quaternion Complementary Filter," 2011.
- [31] G. Welch e G. Bishop, "An Introduction to the Kalman Filter," *Department of Computer Science, University of North Carolina, Chapel Hill*, 2001.
- [32] R. E. Kalman, "A New Approach to Linear Filtering and Prediction Problems," *Transactionsof the ASMAE - Journal of Basic Engineering*, 1960.
- [33] M. Wooley, *BlackBerry-10-Bluetooth-LE-primer-for-developers*, [Online], Accessed 30 July 2015.
- [34] T. I. Wiki, http://processors.wiki.ti.com/index.php/CC2541_SensorTag, 2015.
- [35] T. Instruments, <http://www.ti.com/tool/cc2540emk-usb>, 2015.
- [36] T. Instruments, "Texas Instruments CC2540/41 Bluetooth® Low Energy Software Developer's Guide v1.4," 2013.
- [37] T. I. Wiki, http://processors.wiki.ti.com/index.php/BLE_Device_Monitor_User_Guide, 2015.
- [38] Evothings, <http://evothings.com/doc/examples/ble-ti-sensortag.html>, 2015.
- [39] Richards-Tech, <https://github.com/richards-tech/MPU9150Lib/blob/master/README.md>, 2015.
- [40] OpenTK, <http://www.opentk.com/>, 2015.
- [41] T. I. Wiki, <http://processors.wiki.ti.com/index.php/Category:HealthDemo>, 2012.
- [42] S. Electronics, <https://www.sparkfun.com/products/11486>, 2015.