



INSTITUTO SUPERIOR DE ENGENHARIA DE LISBOA

ÁREA DEPARTAMENTAL DE ENGENHARIA DE ELECTRÓNICA E TELECOMUNICAÇÕES E DE COMPUTADORES

TRABALHO FINAL DE MESTRADO PARA OBTENÇÃO DO GRAU DE MESTRE EM
ENGENHARIA DE ELECTRÓNICA E TELECOMUNICAÇÕES

Magnetómetro 3D em sistema de mapeamento: Caraterização, Integração e Testes

Valter Rodrigues

Orientador:

Prof. Doutor Manfred Niehus

Júri:

Presidente:

Prof. Doutor Mário Véstias

Vogais:

Prof. Doutor Paulo Marques

Prof. Doutor Mário Moreira

Fevereiro 2018

Agradecimentos

Quero agradecer muito ao Professor Doutor Manfred Niehus por todo trabalho e dedicação no acompanhamento e orientação durante este trabalho, foi sem dúvida fundamental.

Agradecer também Professor Doutor Mário Moreira pela forma sempre pronta a que respondeu aos meus pedidos de auxílio aquando das medidas com o drone e do tratamento de dados.

Agradecer a todos os professores e colegas do ODI que se disponibilizaram sempre para ajudar e me forneceram algum material necessário para o projecto.

Um agradecimento especial à minha namorada Flávia Frazão que sempre me ajudou, deu força, carinho e motivação para alcançar este objectivo.

E claro aos meus pais que me possibilitaram chegar até aqui, dando-me sempre as melhores condições possíveis, carinho e também motivação.

Finalmente um agradecimento a todos os meus colegas de curso, professores e amigos que me deram sempre ânimo para alcançar esta meta.

Resumo

Neste projecto é caracterizado, integrado e testado um magnetómetro 3D em sistema de mapeamento, com a finalidade de ser utilizado em mapeamentos de áreas terrestres. O sistema desenvolvido é independente, podendo ser utilizado em mapeamentos manuais ou acoplado a dispositivos específicos de medida. É utilizado *hardware* comercial de baixa e média gamas obtendo desta forma um produto final versátil, preciso, modular e de custo reduzido. Todo o *software* desenvolvido tem por base ser *open source*, evitando assim a necessidade de licenças ou outros compromissos na execução do trabalho.

É feito o estudo comparativo entre diversos processos de calibração, que garantem a inexistência de distorções e erros de instrumentação nos valores obtidos através do magnetómetro. Para além disso, de forma a garantir que todos os desenvolvimentos e processos utilizados são bem implementados procede-se ao teste e comparação entre os valores medidos pelo sistema desenvolvido e os valores medidos por um magnetómetro de céσιο. Obtém-se uma boa correlação, ligeiramente limitada pelo facto de a precisão do sistema desenvolvido não ser tão elevada quanto a do magnetómetro de céσιο utilizado.

Posteriormente, são feitos mapeamentos de áreas terrestres onde é introduzida uma anomalia, neste caso concreto um carro, de forma a se poder aferir a capacidade do sistema para caracterizar a mesma. São obtidos resultados onde se detecta claramente a anomalia e são medidos os valores máximos dessa alteração do campo magnético que são próximos de 1,5 microTesla em mapeamentos manuais a uma altitude de um metro e de aproximadamente 0,7 microTesla em mapeamentos aéreos com drone a uma altitude de seis metros, sendo que o campo regional para o local de medida tem um valor médio de aproximadamente 43,8 microTesla. Antes de se utilizar o drone é verificada a possibilidade de este introduzir interferências nos valores medidos e também são planeadas e simuladas a missões a realizar por este.

Palavras-chave: Sistema, Campo magnético, 3D, Calibração, Compensação, Drone, Caracterização, Integração, Testes, Arduino, Interferência, Mission Planner e Voo.

Abstract

In this project a 3D magnetometer in a mapping system is characterized, integrated and tested, with the purpose of being used in mapping of terrestrial areas. The developed system is independent, ie can be used in manual mapping mode or coupled to specific measurement devices. Commercial hardware of low and medium range is used, thus obtaining a versatile, precise, modular and low cost final product. All software developed is based on open source code, thus avoiding licenses or other commitments in the execution of the work.

A comparative study is made between several calibration processes, which guarantee the absence of distortions and instrumentation errors in the values obtained through the magnetometer. In addition, in order to guarantee that all development processes are well implemented, the developed system is tested and compared with a commercial cesium magnetometer. A good correlation is observed, however the precision of the developed system is below that of the cesium magnetometer.

Then, mappings are made in terrestrial areas where a magnetic anomaly is introduced. For example, a vehicle is used to analyse the system capability to detect and quantify the anomaly. Results are obtained where the anomaly is clearly detected and the maximum values of this magnetic field change are measured, which are close to 1.5 microTesla in manual mappings at an altitude of one meter and approximately 0.7 microTesla in aerial mapping with UAV at an altitude of six meters. The regional field for the measurement site has an average value of approximately 43.8 microTesla.

Finally, the impact of motor-induced magnetic interferences is investigated, and survey mission is planned and simulated to missions to perform by him.

Keywords: Magnetic Field, 3D, Calibration, Compensation, Drone, Characterization, Integration, Testing, Arduino, Interference, Mission Planner and Flight.

Índice Geral

Agradecimentos.....	iii
Resumo	v
Abstract	vii
Índice de Figuras	xi
Índice de Tabelas	xv
Índice de Equações	xvii
Lista de Acrónimos	xix
1. Introdução	1
1.1. Objectivos a alcançar.....	1
1.2. Motivação	3
1.3. Enquadramento	4
1.4. Estado da Arte.....	9
1.5. Estrutura do Relatório.....	14
2. Definição do sistema e planeamento	15
2.1. Módulos e componentes	15
2.2. Metodologias de testes, de medições manuais e dos voos	23
3. Desenvolvimento e Implementação	27
3.1. Hardware	28
3.2. Software	36
4. Calibração	45
4.1. Modelo de calibração	45
4.2. Processos de calibração	49
5. Testes e Validação do Sistema	61
5.1. Em Laboratório.....	61
5.2. No Campo.....	64

6. Mapeamentos.....	69
6.1. Método de Kriging	69
6.2. Mapeamentos Manual.....	75
7. Mapeamentos com Drone	79
7.1. Drone	79
7.2. Simulação com Ardupilot e Mission Planner	81
7.3. Estudo de Interferências e Mapeamentos.....	85
8. Discussão e Conclusões	93
Referências Bibliográficas	99
Anexo A1	103
Anexo A2.....	104

Índice de Figuras

Figura 1 – Veículo aéreo de controlo remoto e voo automático programável, Magdrone [8].	4
Figura 2 - Exemplos de sistemas existentes para medida de campos magnéticos. [9] [10]	5
Figura 3 - Campo magnético em redor de um dipolo magnético e de linhas de media e alta tensão. [11]	6
Figura 4 - Representação de referência do campo magnético. [35]	7
Figura 5 – Representação da variação da intensidade do campo magnético terrestre. [38]	8
Figura 6 - Mapeamento mundial das anomalias mais significativas. [12]	9
Figura 7 - Anomalias mais significativas na península ibérica. [13]	9
Figura 8 - Protótipo GeoSurv II e pássaro magnético.	11
Figura 9 - UAS Venture e UAS SkyLance. [16]	12
Figura 10 - Anomalias magnéticas identificadas com recurso ao UAS SkyLance.	12
Figura 11 – MagDrone R3 da Sensys. [17]	13
Figura 12 –DroneMag da Gemsys. [18]	13
Figura 13 - Diagrama de blocos de alto nível do sistema.	15
Figura 14 - Ângulo aproximado do campo magnético para a região de Lisboa. [36]	24
Figura 15 – Diagrama de blocos do sistema.	27
Figura 16 - Ligações do magnetómetro Pololu LIS3MDL de 3 eixos.	28
Figura 17 - Placa microcontroladora e <i>schematic</i> do Arduino 101 Intel® Curie.	29
Figura 18 - Ligações do sistema de gravação Adafruit MicroSD card.	30
Figura 19 - Ligações do dispositivo de localização Ublox GY-NEO6MV2.	31
Figura 20 – Baterias TopFuel LiPo 25C-ECO-X 2S e 3S de alimentação do sistema.	32
Figura 21 - Ligações dos botões e LED.	33
Figura 22 - Acoplamento de todo o sistema à barra de alumínio de um metro.	34
Figura 23 - Descrição da localização de cada dispositivo no sistema.	35
Figura 24 - Fluxograma do <i>setup()</i> e <i>loop()</i> do sistema.	38
Figura 25 - Fluxograma do modo STANDBY.	39

Figura 26 - Fluxograma do modo COMPENSACAO.	40
Figura 27 - Fluxograma do modo TESTES.....	41
Figura 28 - Fluxograma do modo GRAVAR.	42
Figura 29 - Esquematização dos erros de instrumentação. [35].....	46
Figura 30 – Esquematização das interferências magnéticas. [35].....	47
Figura 31 - Comparação entre a Pré e Pós calibração para o método ST Microelectronics.....	52
Figura 32 - Comparação entre a Pré e Pós calibração para o método TESLABS (Qingde Li e J.G. Griffiths).....	53
Figura 33 - Execução do <i>software</i> Magneto (Linguagem C).....	55
Figura 34 – Exemplo da execução do <i>software</i> Magneto.....	56
Figura 35 - Exemplo de utilização do MagMater para calibrar o magnetómetro.....	57
Figura 36 - Sistema de calibração com esquematização das várias posições necessárias (Extraído de <i>software</i> MagMaster).	58
Figura 37 - Valores do magnetómetro antes da calibração (Imagem MagViewer)...	59
Figura 38 - Valores do magnetómetro depois da calibração (Imagem MagViewer). 60	
Figura 39 - Bobines de Helmholtz. [43]	61
Figura 40 - Correlação entre o sinal gerado e o sinal medido e entre as partes positiva e negativa deste.	62
Figura 41 - Medida de longa duração para aferir a estabilidade do sinal.	63
Figura 42 - Magnetómetro de Césio Geometrics utilizado nas medidas. [45]	64
Figura 43 - Conversão para coordenadas UTM e translação para obter distância em metros.....	66
Figura 44 - Rotação dos valores obtidos e remoção dos valores absurdos.	66
Figura 45 - Correlação dos valores obtidos pelo sistema desenvolvido e os obtidos pelo magnetómetro de césio.	67
Figura 46 – Esquematização de um percurso realizado para obtenção das medidas de campo magnético.	69
Figura 47 - Representação dos vários pontos medidos pelos dois sistemas.	72
Figura 48 - Interpolação e obtenção da variância com o método de Kriging para os valores obtidos com o magnetómetro de césio.	72
Figura 49 - Interpolação e obtenção da variância com o método de Kriging para os valores obtidos com o sistema desenvolvido.....	73

Figura 50 – Representação em gráfico 3D dos valores obtidos valores obtidos com o magnetómetro de céσιο, com a subtração do plano regional efectuada.....	73
Figura 51- Representação em gráfico 3D dos valores obtidos valores obtidos com o sistema desenvolvido, com a subtração do plano regional efectuada.....	74
Figura 52 - Campo magnético residual da área estudada, com representação da anomalia provocada pelo carro.	75
Figura 53 - Representação das linhas de iso dos valores do campo magnético e do campo regional.....	76
Figura 54 - Representação 3D da área estudada e da anomalia provocada pelo carro.	77
Figura 55 - Drone quadrirotor (Magdrone).....	79
Figura 56 - Software Ardupilot a executar na máquina virtual Linux.....	81
Figura 57 - Exemplo de um mapeamento de terreno Nordeste para Sudoeste -> Sudoeste para Nordeste.....	83
Figura 58 -Análise de interferência/ruído com motores a armar (Passar por cima)..	85
Figura 59 - Análise de interferência/ruído com motores a armar (Parar em cima). ..	86
Figura 60 - Análise de interferência/ruído com motores a acelerar bruscamente.....	86
Figura 61 - Gráfico que ilustra a inexistência de interferências provocadas pelo funcionamento dos motores do drone. [17]	87
Figura 62 - Varrimentos lineares de nordeste para sudoeste e vice-versa.....	88
Figura 63 - Resultado da média dos dois perfis e evolução do campo regional.....	89
Figura 64 - Resultado da subtração do campo regional aos dois perfis.....	89
Figura 65 - Representação dos vários pontos medidos com recurso ao drone.....	90
Figura 66 - Interpolação e obtenção da variância com o método de Kriging para os valores obtidos com o sistema desenvolvido com recurso ao drone.....	91
Figura 67 - Representação em gráfico 3D dos valores obtidos valores obtidos com o sistema desenvolvido com recurso ao drone, com a subtração do plano regional efectuada.....	92

Índice de Tabelas

Tabela 1 - Listagem de magnetómetros a escolher. [19] [20] [21].....	17
Tabela 2 - Listagem de placas microcontroladoras a escolher. [22] [23] [24] [25]....	18
Tabela 3 - Listagem de dispositivos de gravação de dados a escolher. [26] [27]....	20
Tabela 4 - Listagem de dispositivos de localização via GPS a escolher. [28] [29] ...	21
Tabela 5 - Listagem de baterias para alimentação do sistema. [30] [31].....	22
Tabela 6 – Resumo dos processos de calibração.	51
Tabela 7 – Passos necessários para se efectuar a análise dos valores obtidos.	65
Tabela 8 - Passos necessários para se efectuar a análise com o método de Kriging dos valores obtidos.....	70
Tabela 9 – Datasheet do Sistema Desenvolvido.	96

Índice de Equações

Equação 1 – Representação do campo magnético terrestre.....	7
Equação 2 – Equação geral que descreve um elipsoide.....	45
Equação 3 – Representação semi-matricial que descreve um elipsoide.....	45
Equação 4 – Representação matricial que descreve um elipsoide.	46
Equação 5 - Matriz diagonal que representa o erro de factor de escala.	46
Equação 6 - Matriz que representa o erro de não-ortogonalidade dos eixos do sensor.....	47
Equação 7 – Matriz que representa o erro devido ao deslocamento do sensor.	47
Equação 8 – Matriz que representa a distorção provocada por <i>hard iron</i>	48
Equação 9 – Matriz que representa a distorção provocada por <i>soft iron</i>	48
Equação 10 – Campo magnético medido combinado com todas as fontes de distorção.....	48
Equação 11 – Campo magnético ideal livre de perturbações magnéticas.	48
Equação 12 – Quadrado da intensidade do campo magnético.	49
Equação 13 - Reescrita da Equação 11.	49
Equação 14 – Equação resultante da combinação da Equação 13 com a Equação 12.	50
Equação 15 – Equação que descreve o comportamento do campo magnético entre as bobinas de Helmholtz. [21]	61

Lista de Acrónimos

ARC – Argonaut RISC Core

BLE – Bluetooth Low Energy

CLK – Clock

CS – Chip Select

FAT16 – 16 Bits File Allocation Table

FAT32 – 32 Bits File Allocation Table

GPS – Global Positioning System

GPS-RTK – Global Positioning System with Real Time Kinematics

I2C – Inter-Integrated Circuit

ICSP – In Circuit Serial Programming

IDE – Integrated Development Environment

LIDAR – Light Detection And Ranging

PFC – Projecto Final de Curso

PWM – Pulse-Width Modulation

RGB – Red, Green and Blue

RMS – Root Mean Square

RTOS – Real Time Operating System

SNR – Signal-to-Noise Ratio

SPI – Serial Peripheral Interface

UAS – Unmanned Aircraft Systems

UTM – Universal Transverse Mercator

1. Introdução

1.1. Objectivos a alcançar

Ao iniciar o projecto traçam-se de imediato alguns objectivos, não só para ajudar a definir os requisitos do mesmo como para começar a estruturar os processos de desenvolvimento. Tendo por base que o sistema desenvolvido é utilizado em mapeamentos de áreas terrestres, podendo ser utilizado de forma manual ou acoplado a dispositivos específicos de medida. Assim sendo, definem-se três objectivos principais:

- Construir um sistema de medida de campo magnético que seja versátil, fiável e de baixo custo;
- Ser um sistema independente;
- Que possibilite a caracterização clara de campos magnéticos 2D em qualquer área terrestre.

A partir destes objectivos base, surgem outros, que permitem alcançar os primeiros e estruturar o sistema:

- Integrar um magnetómetro 3D que possibilite a caracterização do campo magnético terrestre.
- Introduzir um dispositivo de localização, visto que este é um sistema independente, um dispositivo de localização é indispensável para localizar as medidas.
- Criar uma *interface* com o utilizador que permite uma fácil utilização do sistema na área de medidas.

- Permitir a gravação dos dados obtidos.

Para testar e se ter confiança no sistema são definidas também algumas metas a atingir:

- Obter uma boa calibração dos valores medidos.
- Realizar medidas com o sistema em ambientes controlados de campo magnético, onde se consegue saber exactamente a evolução deste, o que permite aferir a fiabilidade e precisão do mesmo.
- Realizar medidas de longa duração para aferir a estabilidade dos valores medidos ao longo do tempo.
- Medir e comparar os valores de campo magnético obtidos manualmente com o sistema, com os obtidos com um magnetómetro de céσιο altamente fiável e preciso.
- Aferir a repetibilidade dos valores medidos, para garantir o funcionamento correcto do sistema, introduzindo uma anomalia no percurso a realizar, medir num determinado sentido e repetir a medida no sentido oposto a este.
- Posteriormente, utilizar um drone para efectuar o mapeamento de áreas terrestres mas em primeiro lugar despistar as possíveis interferências causadas pelos motores deste nos valores medidos.
- Mapear áreas terrestres, quer de forma manual quer com o auxílio do drone de forma a obter uma representação magnética 2D das mesmas, comparando à posteriori as duas formas de obtenção de medidas.
- Finalmente, retirar conclusões dos resultados obtidos, discutir sobre o desenvolvimento de todo o projecto e propor abordagens futuras.

1.2. Motivação

Hoje em dia existem muitos sistemas de medição e caracterização de campos magnéticos usados nos mais diversos tipos de trabalhos de estudos electromagnéticos e geofísicos, tais como, perceber quais as alterações provocadas por linhas de alta tensão nos sistemas electrónicos e de telecomunicações [1], cartografar terrenos, analisar a evolução do magnetismo terrestre [2] [3] [4], estruturas geológicas que apresentem elevados valores de magnetização ou de contrastes de magnetização, como é o caso de rochas magmáticas, ambientes vulcânicos ou intrusões ricas em minerais metálicos em ambientes sedimentares ou metamórficos [5] [6]. Mas todos ou praticamente todos esses sistemas de medição apesar de muito precisos são também muito caros. Para além disso, normalmente, são sistemas pouco versáteis pois são desenvolvidos, à medida, tendo em conta o tipo de utilização que se pretende dar-lhes e também são sistemas com uma dimensão, em geral, grande. A principal motivação para o desenvolvimento deste projecto foi precisamente tentar construir um sistema que conseguisse solucionar grande parte destas problemáticas, ou seja, ser um sistema pequeno, de custo reduzido, versátil e com a maior precisão possível. Este sistema é concebido com o intuito de poder ser usado quer em medições manuais, quer acoplado a dispositivos específicos de medida, tais como aviões, drones, carrinhos ou quaisquer sistemas autónomos de caracterização de campos magnéticos.

Apesar de neste trabalho a maior incidência ser sobre o estudo de campos magnéticos com medições feitas de forma manual e também com recurso a um drone, o sistema pode ser integrado em qualquer outro dispositivo e usado de diversas formas, tendo a característica de ser modular e permitir ser reformulado facilmente para se acoplar a qualquer outro dispositivo.

O objectivo é então caracterizar, integrar e testar um magnetómetro 3D para que com ele seja possível caracterizar campos magnéticos numa grande diversidade de áreas terrestres, tornar esse processo mais simples, mais rápido e mais eficaz quer ele seja feito manualmente ou de forma autónoma recorrendo aos referidos dispositivos específicos de medida.

1.3. Enquadramento

No seguimento do trabalho desenvolvido anteriormente no projecto final de curso da licenciatura [7] onde foi analisada a alteração do campo magnético perto de objectos em movimento e também tendo em vista a necessidade de integração de um magnetómetro 3D no projecto Magdrone actualmente em desenvolvimento no ISEL, surge então a ideia de criar um sistema que possa não só dar resposta às problemáticas encontradas no desenvolvimento do PFC como também criar algo que possa ser usado de forma independente ou acoplado a um dispositivo específico de medida, como é o caso do Magdrone [8].

O projecto Magdrone [8] consiste na construção e desenvolvimento de um veículo aéreo de controlo remoto e voo automático programável, ou seja, um drone.

O Magdrone é composto por três módulos principais que são integrados no drone: um sistema de medida construído com base num sensor magnético 3D, de custo reduzido, um LIDAR e um GPS-RTK com precisões centimétricas que asseguram o posicionamento 3D das medidas efectuadas. O drone é utilizado, para a realização de perfis cartográficos através de um sensor LIDAR e aeromagnéticos através do sistema que aqui é apresentado. A utilização do GPS-RTK e do LIDAR permite a realização de voos com controlo automático da distância ao solo e cartografar facilmente áreas terrestres (Figura 1) [50].



Figura 1 – Veículo aéreo de controlo remoto e voo automático programável, Magdrone [8].

O sistema aqui desenvolvido é construído de forma a poder ser acoplado, por exemplo, ao drone ou a funcionar autonomamente para mapeamentos magnéticos de áreas terrestres. É transversal a várias áreas de estudo tais como telecomunicações, eletromagnetismo, electrónica e geofísica mas enquadra-se principalmente na aquisição, processamento, análise de sinais e integração do sistema de forma a permitir efectuar o estudo de campos magnéticos. A ideia deste projecto é precisamente desmistificar os processos utilizados actualmente para realização desses estudos, torná-los mais eficazes e versáteis, permitindo assim uma maior facilidade e qualidade de trabalho. Possibilitando também maior rapidez na execução destes e maior capacidade de análise dos resultados obtidos.

Desta forma é possível transformar o trabalho de campo num processo simples e automatizado, visto que para além de permitir mais abordagens ao trabalho e maior variedade de esquematização dos mapeamentos permite também enquadrar este tipo de trabalhos num patamar de eficiência e rapidez de execução que anteriormente era difícil de conseguir. Nomeadamente, com a versatilidade de opções de realização de medidas, quer manualmente quer com recurso a sistemas autónomos. Isto porque os sistemas actualmente usados para além de grandes e volumosos requerem processos de medição morosos e que são feitos de uma forma que oferecem pouca dinâmica, pois requerem tipos de medições muito específicas (Figura 2) [9] [10].



Figura 2 - Exemplos de sistemas existentes para medida de campos magnéticos. [9] [10]

Ao nível das telecomunicações e eletromagnetismo este sistema é uma mais-valia para aferir o impacto provocado, por exemplo, de linhas de média e alta tensão (Figura 3) em dispositivos electrónicos da rede de telecomunicações [1], permite estudar melhor dispositivos que se comportam, por exemplo, como dipolos magnéticos (Figura 3) e bobines [11], ao nível da geofísica agiliza os processos de mapeamento e de estudo magnético de terrenos [2] [3] [4], de estruturas geologicamente activas que apresentem elevados valores de magnetização ou de contrastes de magnetização, como por exemplo, rochas magmáticas, ambientes vulcânicos ou intrusões ricas em minerais metálicos em ambientes sedimentares ou metamórficos [5] [6] e permite estudar anomalias electromagnéticas [12] significativas que caracterizam estruturas desconhecidas, alterações da crosta terrestre ou a composição de terrenos.

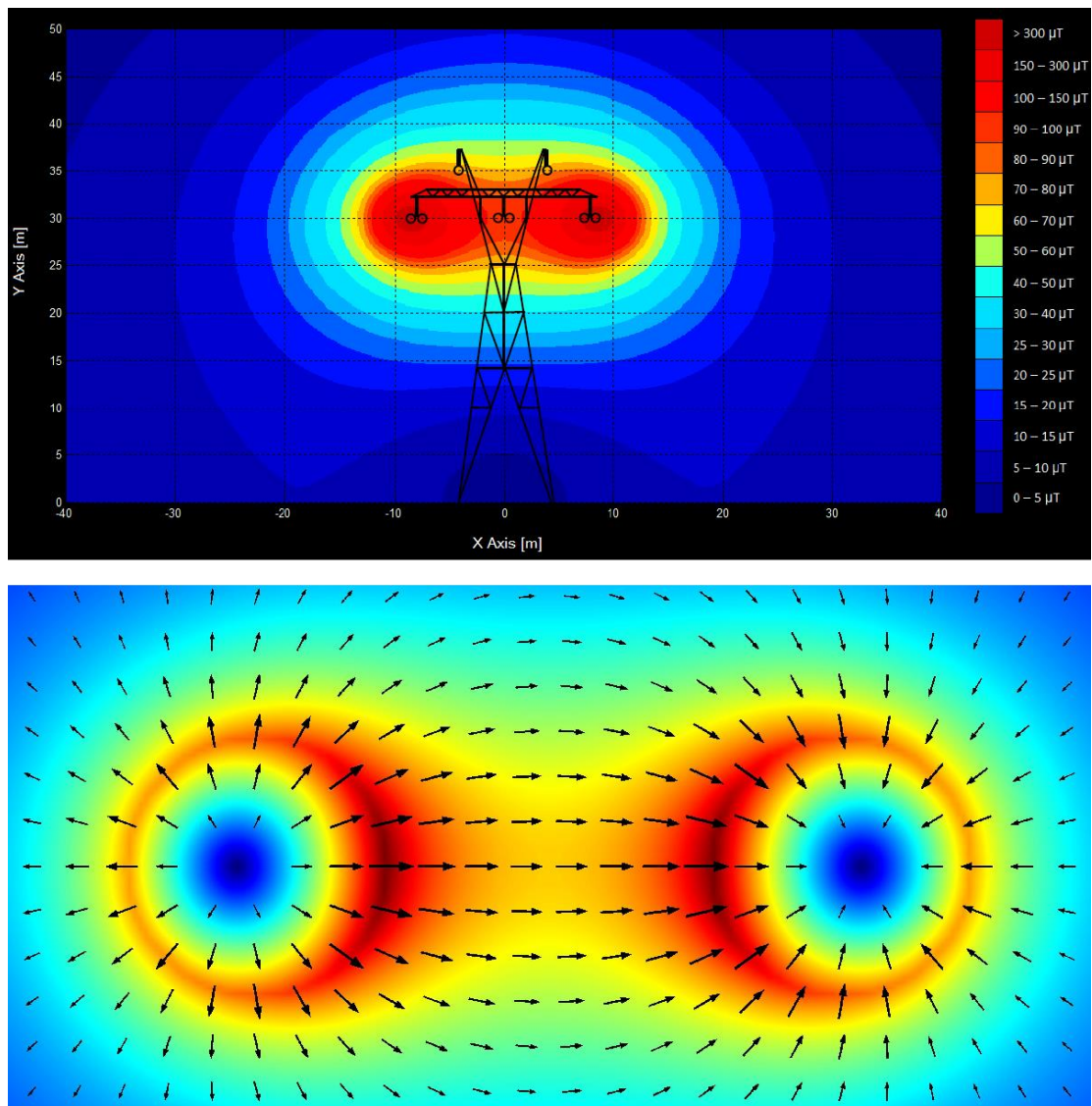


Figura 3 - Campo magnético em redor de um dipolo magnético e de linhas de media e alta tensão. [11]

- **Campo Magnético**

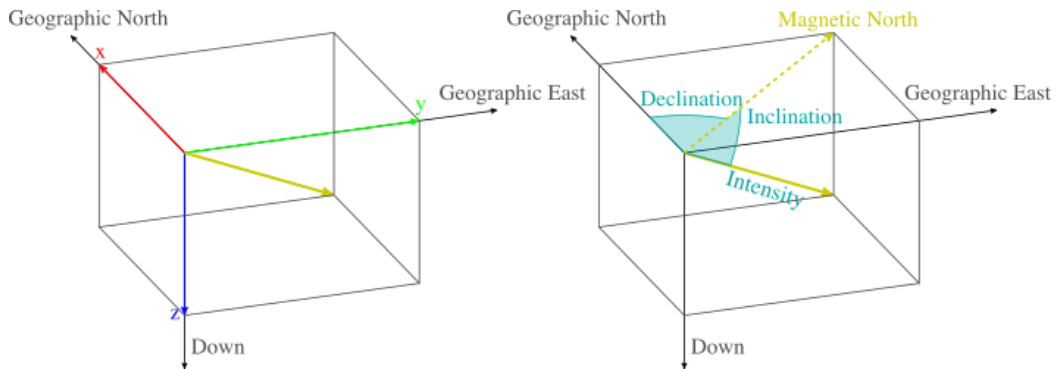


Figura 4 - Representação de referência do campo magnético. [35]

Em qualquer local na Terra, o campo magnético pode ser representado localmente como um vector tridimensional constante (\vec{h}_0) e o seu vector é caracterizado por três propriedades. A primeira, a intensidade ou magnitude (F), normalmente medida em nanoTesla (nT) com uma faixa aproximada de 25000 nT a 65000 nT, a segunda, a inclinação (I), com valores negativos (até -90°) se apontar para cima e positivos (até 90°) se apontar para baixo e a terceira, a declinação (D) que mede o desvio do campo em relação ao norte geográfico e é positivo para leste (Figura 4) [35]. O campo magnético terrestre h_0 pode ser descrito como:

$$\vec{h}_0 = F \begin{bmatrix} \cos(I) \cos(D) \\ \cos(I) \text{sen}(D) \\ \text{sen}(I) \end{bmatrix}$$

Equação 1 – Representação do campo magnético terrestre.

Dado um determinado ponto geográfico é possível obter os valores de F , I e D e para isso pode ser usado o Modelo Magnético Mundial WMM [38].

US/UK World Magnetic Model - Epoch 2015.0
Main Field Total Intensity (F)

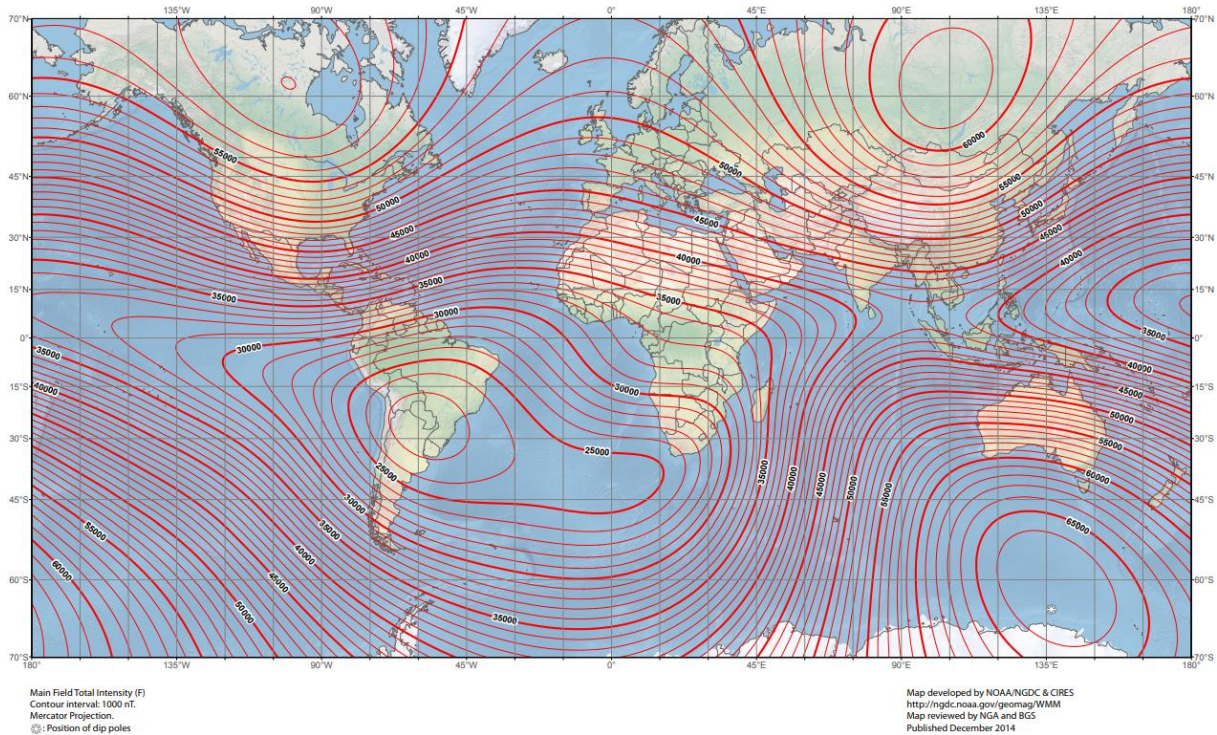


Figura 5 – Representação da variação da intensidade do campo magnético terrestre. [38]

Este modelo é um modelo padrão utilizado pelo Departamento de Defesa dos EUA, o Ministério da Defesa do Reino Unido, a Organização do Tratado do Atlântico Norte (OTAN) e a Organização Hidrográfica Internacional (OHI), para sistemas de referência de navegação, altitude e *heading* usando o campo geomagnético. Também é amplamente utilizado em sistemas de navegação e de navegação civis. O modelo é produzido em intervalos de 5 anos e distribuído pelos Centros Nacionais de Informação Ambiental (NCEI) em nome do Agência Nacional de Inteligência Geoespacial (NGA) (Figura 5) [38].

- **Anomalia**

Uma anomalia magnética é a alteração na magnitude do campo magnético terrestre em relação ao valor médio esperado para um determinado local, onde contrastes de magnetização alteram a intensidade deste (Figura 6 e Figura 7) [12] [13]. Uma anomalia é caracterizada pela subtração do valor esperado para o local em questão ao valor medido, e é medida em nanoTesla (nT). Na geofísica, uma anomalia magnética é uma variação local no campo magnético terrestre, que resulta de

estruturas geológicas que apresentem elevados valores de magnetização ou de contrastes de magnetização, como por exemplo, rochas magmáticas, ambientes vulcânicos ou intrusões ricas em minerais metálicos em ambientes sedimentares ou metamórficos. As anomalias magnéticas são uma pequena fração do campo magnético, o campo total, como referido anteriormente, que varia de 25000 a 65000 nanoTesla (nT). Para se medirem anomalias, os magnetómetros precisam de uma sensibilidade de pelo menos 10 nT.

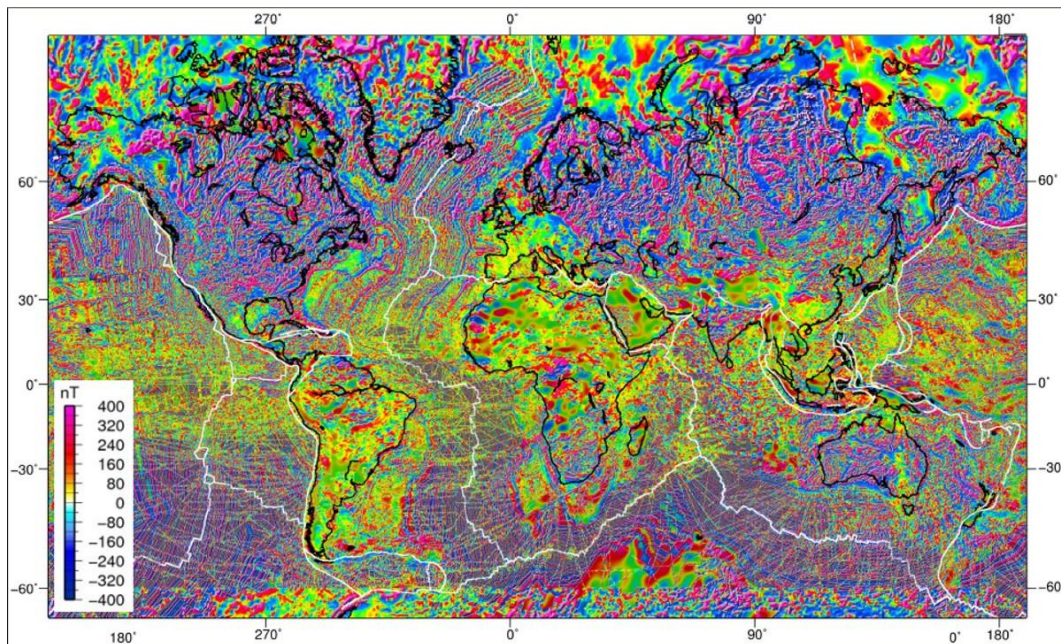


Figura 6 - Mapeamento mundial das anomalias mais significativas. [12]

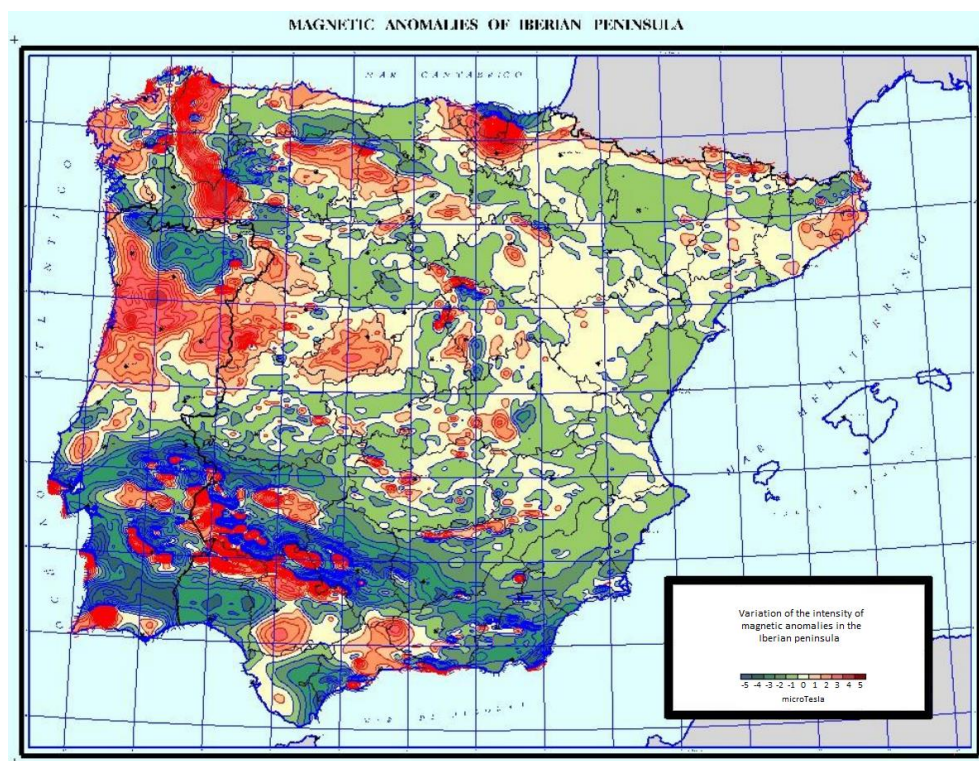


Figura 7 - Anomalias mais significativas na península ibérica. [13]

1.4. Estado da Arte

Desde a década de 50, que é possível medir as variações da intensidade do campo magnético terrestre, inicialmente este tipo de mapeamentos magnéticos eram exclusivamente feitos manualmente, depois devido à necessidade de se cobrirem grandes áreas passaram a ser utilizados aviões e helicópteros tripulados nos quais eram acoplados ou pendurados magnetómetros e posteriormente realizados os mapeamentos. Mais recentemente estes estudos passaram a ser feitos com recurso a pequenas aeronaves não tripuladas, conhecidas com *Unmanned Aircraft Systems* (UAS), o seu *design* é bastante reduzido em relação às anteriores aeronaves, mais convencionais. Neste momento já existem variados protótipos de tamanho reduzido quer de aviões quer de drones, que para áreas de dimensão gigantesca ainda não são a melhor solução, devido à sua baixa autonomia de voo, mas a curto/médio prazo serão sem dúvida os melhores sistemas disponíveis [2] [3] [4] [5] [6].

Existem inúmeras aplicações e sistemas desenvolvidos actualmente para o estudo magnético de grandes áreas com recurso na grande maioria a pequenos aviões ou helicópteros. Apesar de pequenos, estes ainda são de uma dimensão considerável, têm um custo elevado e são de configuração complexa. Relativamente a dispositivos como o Magdrone, que está a ser desenvolvido actualmente pelo ISEL, existem alguns sistemas já desenvolvidos mas não com as mesmas características nem com a mesma tecnologia aplicada. Os dispositivos concebidos até então utilizam magnetómetros de Césio, de elevada qualidade mas também de elevado custo, no caso do Magdrone o magnetómetro utilizado é um sensor comercial muito mais simples mas que tem boa sensibilidade e precisão. Existem também várias empresas que desenvolvem dispositivos deste género com qualidade mas de elevado custo também. Ao longo dos tempos têm surgido muitos sistemas desenvolvidos e alguns protótipos deste género, que são citados abaixo:

Em 2008 o National Institute of Polar Research no Japão como parte integrante do projeto Ant-Plane para pesquisa científica e logística de verão na região costeira da Antártida, desenvolveu seis tipos de UAS's, autónomos pequenos (veículos aéreos não tripulados, semelhantes a drones, denominados Ant-Plane) com base em quatro tipos de fuselagem. Em voos de teste, o Ant-Plane 2 voou com uma precisão de 20

metros ao longo de um curso directo durante o tempo calmo no vulcão Sakurajima, Kyushu, Japão. Um magnetómetro magneto-resistente de 3 eixos integrado, com 400g registou variações no campo magnético para uma precisão de 10 nanoTesla durante períodos de vento calmo, mas observou-se forte ruído magnético durante os ventos fortes, especialmente com ventos de cabeça. O Ant-Plane 4-1 alcançou um voo contínuo de 500 km, com uma altitude máxima de 5690 metros. O Ant-Plane pode ser usado para vários tipos de pesquisa antártica como uma plataforma básica para mapeamentos magnéticos. [14]

Em 2010 a Universidade Carleton, em colaboração com a Sander Geophysics Limited, desenvolveu um *design* original para um sistema de aeronaves não tripuladas para levantamento aeromagnético, o GeoSurv II (Figura 8). Um primeiro protótipo do GeoSurv II da aeronave modular foi construído e tinha uma extensão de asa de 4,88m e um comprimento de 4,27m e uma massa máxima de descolagem de 90 kg.

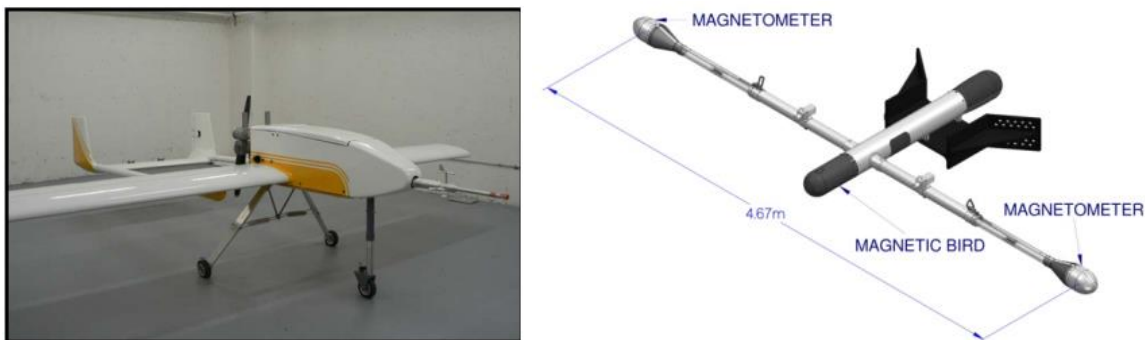


Figura 8 - Protótipo GeoSurv II e pássaro magnético.

Este protótipo foi, em primeiro lugar, usado para testes de voo e depois equipado com dois magnetómetros de Césio nas pontas das asas e um magnetómetro de fluxgate na fuselagem. Durante a construção do GeoSurv II foi também simulado um UAS, este UAS é um pássaro magnético modificado e rebocado com recurso a um helicóptero. Possui dois magnetómetros com a mesma distância de separação que o *design* do GeoSurv II através do qual foi feita uma pesquisa de teste, usando o UAS simulado e os dados adquiridos foram comparados com dados aeromagnéticos regionais convencionais obtidos usando um avião de asa fixa. Este dispositivo foi desenvolvido para estudo de grandes áreas, de no máximo 1000 metros quadrados. [15]

Em 2016, Michael Cunningham da Carleton University, Ottawa, Ontario, Canadá construiu um UAS de asa rotativa, ou seja, um drone chamado SkyLance que é muito parecido com o Magdrone, é composto por 8 rotores alimentados por 4 baterias, inclui sistema LIDAR e também um GPS-RTK, a maior diferença para além da dimensão está precisamente no magnetómetro, que é de Césio, um Geometrics G-823A. Ele voou sobre um depósito de zinco com três anomalias magnéticas e produziu dados repetíveis que foram comparados com medições manuais e com voos realizados com outro dispositivo, nomeadamente um UAS de asa fixa, um pequeno avião, construído também no mesmo projecto, chamado Venturer (Figura 9).



Figura 9 - UAS Venture e UAS SkyLance. [16]

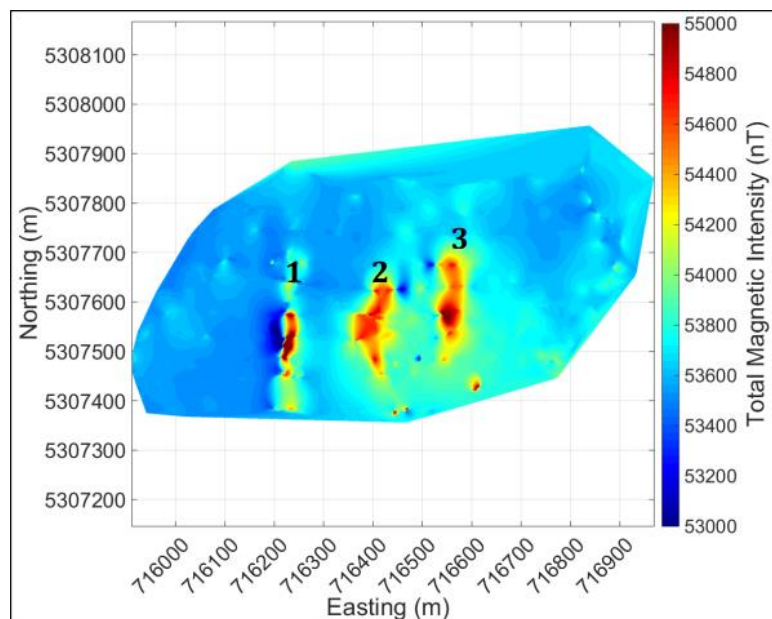


Figura 10 - Anomalias magnéticas identificadas com recurso ao UAS SkyLance. [16]

Com o UAS SkyLance conseguiu mapear a área de terreno definida de forma clara e verificar variações de intensidade do campo magnético entre 53 microTesla (μT) e 55 μT , sendo que o valor máximo das anomalias encontrado foi de aproximadamente 1,66 μT , visto que o campo magnético médio para a região avaliada era de aproximadamente 53,340 μT (Figura 10). [16]

Relativamente a sistemas idênticos ao que é desenvolvido aqui, existem algumas empresas que se dedicam à construção de dispositivos deste género e que podem ser adquiridos como um produto final, ou seja, prontos a usar. Estes sistemas de medida podem ser acoplados a vários tipos de dispositivos de medida, como é o caso dos UAS. Um exemplo disso é a Sensys, uma empresa Alemã, que se dedica à construção de magnetómetros portáteis de elevada precisão e sensibilidade, que podem ser acoplados a vários tipos de dispositivos, neste caso concreto, para acoplar a UAS têm o MagDrone R3 que é um *kit* de levantamento portátil que pode ser utilizado por qualquer UAS com uma carga útil mínima de 1kg, é constituído por um tubo ultraleve com dois magnetómetros de 3 eixos embutidos, cartão SD interno, baterias recarregáveis e um GPS integrado (Figura 11). [17] Para além da Sensys, há também a Gemsys, uma empresa Canadiana, que tem um dispositivo semelhante, o DroneMag que pode ser acoplado a sistemas de asa fixa, helicópteros e drones, inclui um magnetómetro de potássio projectado para alta resolução de dados e sem ruído (Figura 12). [18]



Figura 11 – MagDrone R3 da Sensys. [17]



Figura 12 –DroneMag da Gemsys. [18]

1.5. Estrutura do Relatório

O relatório está estruturado em várias partes, a primeira delas aborda a definição do sistema, esquematizando todos os componentes necessários para o mesmo e o planeamento dos trabalhos onde são abordadas todas as metodologias de testes, de medição manual e dos voos. A parte seguinte descreve todo o desenvolvimento e implementação do sistema desenvolvido, ou seja, todas as características de *hardware* e *software*. Posteriormente, é desenvolvido todo o modelo de calibração, apresentados vários tipos de processos de calibração e efectuada a comparação entre métodos. Após a descrição da calibração surgem os testes e validações do sistema, feitas em laboratório e no campo.

A fase seguinte do relatório apresenta o método de análise dos dados obtidos e descreve os resultados obtidos em mapeamentos manuais com o sistema desenvolvido. De seguida são apresentados o drone utilizado nos trabalhos, a simulação e planeamento de missões com este, feito o estudo de interferências provocado pelos motores em funcionamento e são feitos os respectivos mapeamentos recorrendo ao quadrirotor.

Finalmente são feitas as discussões sobre o trabalho desenvolvido e retiradas as conclusões devidas. No Anexo A1 está esquematizada a calendarização do projecto e no Anexo A2 está todo o código desenvolvido.

2. Definição do sistema e planeamento

2.1. Módulos e componentes

- **Estruturação**

Antes de se proceder à escolha dos componentes que compõem o sistema é necessário definir o que efectivamente se pretende construir, tendo por base algumas das características definidas para o Magdrone mas tendo em conta também que este é uma sistema que se quer independente e versátil. Então define-se que é necessário ter um microcontrolador que é responsável pelo funcionamento de todo o sistema, uma unidade de interface com o utilizador que permite controlar o sistema de forma simples, ter três modos de funcionamento, um para calibração, outro para testes dos sistema e outro para gravação de dados, onde a interface gráfica é o monitor de série do *software* do microcontrolador, para além da unidade de medição onde está inserido o magnetómetro é necessária uma unidade de gravação de dados que permite

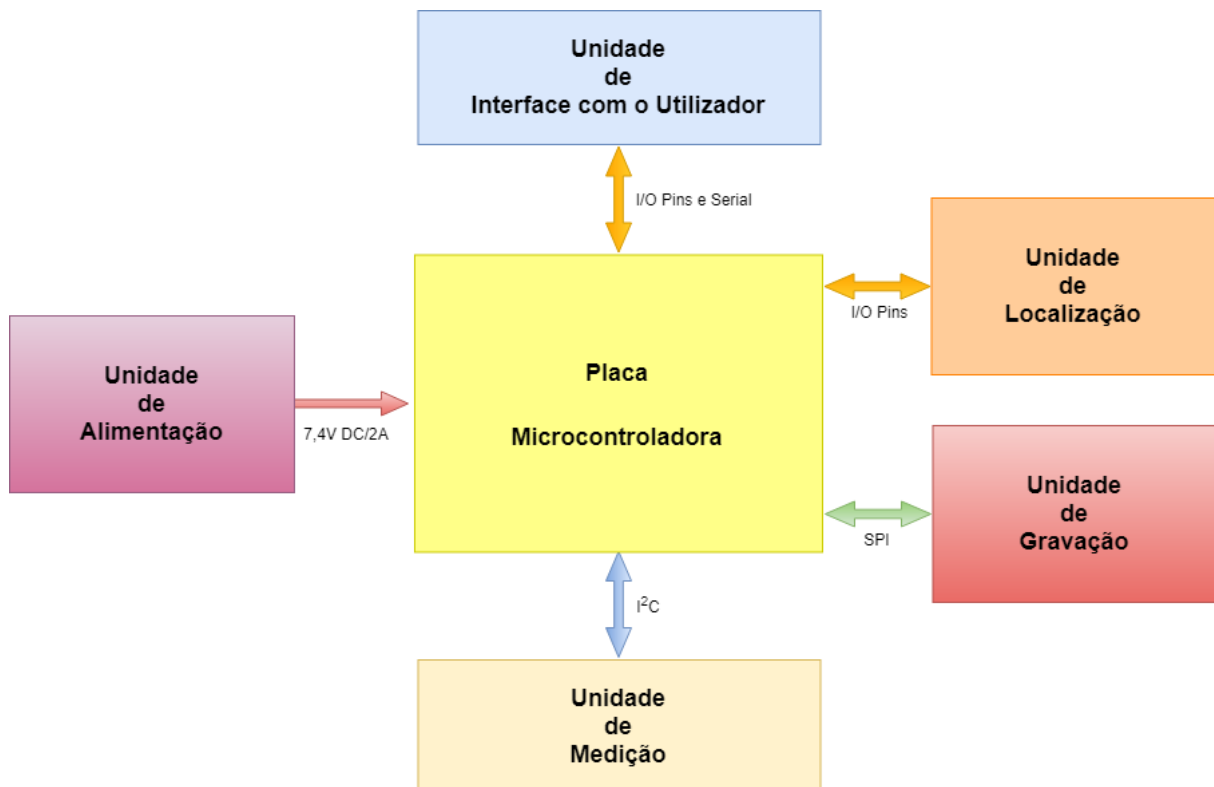


Figura 13 - Diagrama de blocos de alto nível do sistema.

guardar todos os valores medidos e uma unidade de localização que dá a localização exacta de cada valor medido, finalmente a unidade de alimentação que suporta o funcionamento de todas as outras unidades (Figura 15).

Nesta fase inicial do projecto trata-se então de fazer um levantamento criterioso de todo o material necessário e estrutura-se *step-by-step* todas as fases de implementação do mesmo. Assim sendo, é feito um estudo sobre toda a tecnologia disponível, quais os custos desta e posteriormente feita a escolha, tendo por base três premissas importantes, o custo, a fiabilidade e versatilidade dos dispositivos. Para além disso, é feita também a esquematização de todo o trabalho e objectivos a alcançar, quer através da calendarização das etapas quer através da estruturação de todos os processos a seguir.

Numa primeira fase trata-se de escolher todo o *hardware* necessário à implementação do sistema de medida, onde se analisam criteriosamente as vantagens e desvantagens de cada dispositivo e se chega a uma escolha inicial. A ideia é tentar reduzir ao máximo o tamanho do produto final, mas ao longo do desenvolvimento do sistema são feitas algumas adaptações e esse requisito deixa de ter uma importância tão relevante, tendo em conta as limitações encontradas quer ao nível de código quer ao nível da montagem.

Posteriormente pensa-se também em todos os testes, esquematizam-se os mesmos e definem-se processos de comparação e análise com outros sistemas e dispositivos conhecidos para se ter fiabilidade no produto final desenvolvido. Para além disso recorre-se a simuladores que dão um óptimo contributo quer ao nível de plano de missões quer ao nível de comparação e assertividade dos valores obtidos.




Mas de forma a garantir que todos os valores medidos representam efectivamente a realidade, a calibração representa uma parte fundamental neste projecto visto que garante a seriação dos valores obtidos, promovendo a compensação de possíveis interferências ou distorções encontradas no ambiente medido. Esta é uma etapa algo complexa mas que tem a maior importância no desenvolvimento do sistema, requerendo uma grande atenção ao nível de planeamento.

- **Módulos**

Para construir o sistema é necessário ter vários módulos bem estruturados e devidamente integrados, tal como foi referido anteriormente, o principal é o módulo de medida onde está integrado um magnetómetro, depois ter um módulo de controlo e processamento onde está inserida a placa microcontroladora, outro módulo, o de localização, onde está o sistema de GPS, para possibilitar a gravação dos dados obtidos é necessário adicionar um outro módulo, o de gravação, onde está acoplado o dispositivo responsável pela gravação dos dados em ficheiros no cartão SD, finalmente e não menos importante, o módulo de alimentação onde está inserida a bateria responsável pelo funcionamento de todo o sistema.

- **Magnetómetro**

Tabela 1 - Listagem de magnetómetros a escolher. [19] [20] [21]

	Pololu LIS3MDL 	SparkFun MicroMag 	SparkFun Mag3110 
Escala de medição	±4; ±8; ±12; ±16 Gauss ^(#)	±11 Gauss	±10 Gauss
Sensibilidade	0,15; 0,29; 0,44; 0,58 mGauss	0,15 mGauss	1 mGauss
Ruído RMS	3,2 a 4,1 mGauss	-----	25 mGauss
Faixa de temperatura operacional	-40 a +85 °C	-20 a +70 °C	-40 a +85°C
Protocolo de comunicação	I ² C/SPI	SPI	I ² C
Frequência Sensor	80 Hz	-----	80 Hz
Compatibilidade Arduino	Micro, Uno, Mega, 101	Micro, Uno, Mega, 101	Micro, Uno, Mega, 101
Alimentação	2.5V a 5V	3,3V	1.95V to 3.6V
Número de Eixos	3 (X, Y, Z)	3 (X, Y, Z)	3 (X, Y, Z)
Preço	Baixo ^(*)	Alto ⁽⁼⁾	Baixo





(*) < 10€; (=) > 50€; (#)1 Gauss [G] = 100 Microtesla [μ T]

Uma das primeiras análises a fazer é sobre qual o magnetómetro mais adequado para o projecto, primeiro porque ele é a peça principal do mesmo e depois porque só após esta escolha é possível adequar os restantes componentes para o projecto. Como uma das metas a atingir é conseguir encontrar um sensor de valor reduzido mas com a maior sensibilidade possível, chega-se a uma lista com três hipóteses (Tabela 1), da qual, a escolha recai sobre o Pololu LIS3MDL de três eixos,

com regulador de tensão incluído [19], é sem dúvida a melhor hipótese em relação qualidade/preço. Este magnetómetro quando comparado com os restantes tem as melhores características, visto que possibilita uma gama de medição superior, de +-4 a +-16 Gauss e porque trabalha numa gama de temperaturas superior, a segunda hipótese seria o Sparkfun MicroMag, igualmente de três eixos [20]. Para além do preço a principal vantagem é trabalhar com dois protocolos de comunicação I²C/SPI enquanto o MicroMag apenas trabalha com o protocolo de comunicação SPI. Outro inconveniente do magnetómetro MicroMag é o facto de recentemente ter sido interrompida a sua produção por parte da Sparkfun, apesar de ainda ser possível a compra, o Pololu é uma opção mais fiável. Nesta altura a possibilidade de serem introduzidas alterações e de escolher outros componentes também é grande e ter a versatilidade dada pelo Pololu dá também mais margem de escolha para os restantes componentes. O Sparkfun Mag3110 [21] para além de não ser o magnetómetro de mais baixo custo destas três opções é o que oferece menos qualidade e também menos sensibilidade visto que apenas mede intervalos de 1 mGauss enquanto os outros dois medem intervalos de 0,15 mGauss.

- **Placa Microcontroladora**

Tabela 2 - Listagem de placas microcontroladoras a escolher. [22] [23] [24] [25]

	Arduino Micro	Arduino Uno	Arduino Mega	Arduino 101
				
Microcontrolador	ATmega32U4	ATmega328	ATmega2560	Intel® Curie
Faixa de temperatura operacional	-40 a +85 °C	-40 a +85 °C	-40 a +85 °C	-40 a +85 °C
Portas Digitais	20	14	54	14
Portas PWM	7	6	15	4
Portas Analógicas	12	6	16	6
Memória SRAM	2,5kB	2kB	8kB	24kB
Memória Flash	32kB	32kB	256kB	196kB
Clock	16 MHz	16 MHz	16MHz	32MHz
Tensão de Operação	3,3V ou 5V	5V	3,3V ou 5V	3,3V
Sensores	-----	-----	-----	Acelerómetro e Giroscópio ambos com 6 eixos
Comunicação Externa	-----	-----	-----	BLE



Alimentação	7V a 12V	7V a 12V	7V a 12V	7V a 12V
Preço	Baixo ^(*)	Baixo	Alto ⁽⁼⁾	Médio ⁽⁺⁾

(*) < 25€; 25€ < (+) < 45€; (=) > 50€;

Após a escolha do magnetómetro a escolha seguinte recaiu sobre a placa microcontroladora de todo o sistema, peça central e muito importante. Para além de ter de possibilitar uma versátil integração dos vários dispositivos a utilizar, é também necessário que seja um dispositivo com o qual é fácil trabalhar, pois isso poupa tempo de execução ao projecto e torna toda a construção, em si, também mais fácil. Não obstante, os requisitos principais são ter uma boa capacidade de processamento e uma boa capacidade de memória *flash*. Isto porque já é previsível a utilização de bibliotecas e código pesados, quer para construção de todas as funções básicas do sistema quer para o complexo processo de calibração e compensação. Assim sendo, apesar de inicialmente se pensar na utilização do Arduino Micro [23], visto que este oferece os requisitos pretendidos e pelo facto de ser pequeno, acabou por se optar pelo Arduino 101 [22]. Este possui um microcontrolador Intel[®] Curie que para além de oferecer mais capacidade de processamento do ATmega32U4 do Arduino Micro oferece também uma memória *flash* cerca de seis vezes maior. Para além disso, este dispositivo inclui um acelerómetro e um giroscópio, ambos de seis eixos, que possibilitam uma possível utilização como suporte ao sistema de localização a introduzir também. Não menos importante, este possui um sistema de comunicação *Bluetooth Low Energy* que torna possível uma implementação de comunicação a curta distância e em tempo real com outros dispositivos externos. Qualquer um deles disponibiliza uma quantidade suficiente de portas quer analógicas, quer digitais, quer PWM para suporte a todos os dispositivos a integrar no sistema (Tabela 2).

- **Gravação de dados**

Tabela 3 - Listagem de dispositivos de gravação de dados a escolher. [26] [27]



	Adafruit microSD Card	Itead Studio SD Shield 3.0
		
Durabilidade	10000 inserções/remoções	-----
Cartão	Classe 6	-----
Tipo de Cartão	microSD	microSD/SD
Faixa de temperatura operacional	-40 a +85 °C	-40 a +85 °C
Protocolo de comunicação	SPI	UART, IIC, SPI
Compatibilidade Arduino	Micro, Uno, Mega, 101	Uno, Mega, 101
Alimentação	3,3V a 5V	3,3V a 5V
Preço	Baixo ^(*)	Baixo

^(*) < 7€;

Como módulo de gravação de dados adota-se o dispositivo Adafruit microSD Card [26], visto ser um dispositivo fiável e estável que funciona através do protocolo de comunicação SPI. Apesar de a outra hipótese, o Itead Studio SD Shield 3.0 [27], encaixar milimetricamente no Arduino 101 este leva a alguma limitação ao nível das portas disponíveis para a integração dos restantes dispositivos, tendo em conta que este *shield* adiciona algumas saídas específicas pouco interessantes na óptica deste sistema em concreto. Para além disso é um sistema com alguma instabilidade, pois gera algumas problemáticas na utilização de cartões mais recentes, ou seja, de classes superiores (Tabela 3). Não esquecendo também que a escolha inicial é feita com base no Arduino Micro e o *shield* não é compatível com este.

- **Localização via GPS**

Tabela 4 - Listagem de dispositivos de localização via GPS a escolher. [28] [29]

	ITEAD Studio RoyalTek REB-4216 GPS Shield	Ublox NEO-6M-V2 GPS Module
		
Precisão	Até 1,5m	Até 2m
Número de Satélites	20	20
Arranque a frio	35s	27s
Arranque a quente	1s	1s
Baud Rate	9600	9600
Sensibilidade	-159dBm	-161dBm
Número de Canais	48	50
Faixa de temperatura operacional	-25 a +85 °C	-40 a +85 °C
Protocolo de comunicação	UART	UART
Compatibilidade Arduino	Uno, Mega, 101	Micro, Uno, Mega, 101
Alimentação	3,3V a 5V	2,7V a 5V
Preço	Médio ^(*)	Baixo ^(*)

(*) < 8€; 25€ < (+) < 45€;

Após a análise das características dos dois dispositivos chega-se à conclusão que é melhor perder alguma precisão na localização em detrimento de outras características não menos importantes para o sistema. O dispositivo ITEAD Studio RoyalTek REB-4216 GPS Shield [28] tem maior precisão do que o Ublox NEO-6M-V2 GPS Module [29] mas tem por outro lado um arranque a frio mais lento, uma sensibilidade menor, menos canais e uma gama de temperatura operacional menor também (Tabela 4). Ou seja, como estas últimas características são muito mais importantes, como todos os dispositivos escolhidos até ao momento estão dentro de uma gama de temperatura operacional de -40 a +85 °C opta-se por manter esta característica e ganhar também uma maior estabilidade do conjunto propriamente dito. Apesar de se perder cerca de 0,5m de precisão na localização, que ao fim ao cabo é praticamente impercetível, na grande maioria dos casos não são obtidas as condições perfeitas de sinal, quer pelas características dos terrenos, quer pela envolvente dos locais, quer por possíveis tempestades solares que alteram o sinal recebidos dos satélites, entre outras problemáticas existentes.

- **Alimentação do Sistema**

Tabela 5 - Listagem de baterias para alimentação do sistema. [30] [31]

	TopFuel LiPo 25C-ECO-X 1300mAh 2S	TopFuel LiPo 25C-ECO-X 1800mAh 3S
		
Número de Células	2	3
Capacidade	1300mAh	1800mAh
Voltagem Nominal	7,4V	11,1V
Duração da Carga	25C	25C
Conector de Balanço	System XH	System XH
Conector	EC3	EC3
Preço	Baixo ^(*)	Baixo ^(*)

^(*) < 15€

Quanto à alimentação do sistema são usadas duas baterias, uma com maior capacidade que a outra e também com uma tensão nominal superior. Isto porque, para medições mais longas e que requerem bastante tempo de utilização, ter uma capacidade elevada a nível de alimentação é uma mais-valia, a bateria do tipo 3S, ou seja, de três células. Em alternativa, com uma finalidade diferente, por ser suficiente e ter também um tamanho reduzido a bateria do tipo 2S, de duas células, serve para medições mais curtas e que requerem menos tempo de utilização. A nível de capacidades temos a 2S que pesa cerca de 100g com 7,4V/1300 mAh [30] e uma duração de aproximadamente 45 horas e a 3s que pesa cerca de 150g com 11,1V/1800mAh [31] e uma duração de aproximadamente 65 horas, tanto uma como a outra dispõe de um conector EC3, tipicamente utilizado em drones, o que facilita muito a sua utilização e carregamento, visto que dispomos de uma carregador para este tipo de baterias, utilizado para fazer o carregamento do drone do ISEL, o Magdrone (Tabela 5).

2.2. Metodologias de testes, de medições manuais e dos voos

Inicialmente, nesta fase do planeamento é feita toda a esquematização dos testes do sistema, antes deste poder ser efectivamente usado. Para além de testes básicos ao nível do processo de desenvolvimento e do processo de calibração, esquematizam-se também testes mais complexos, no terreno de forma manual.

Os primeiros testes a serem feitos são ao nível dos dispositivos seleccionados, criando rotinas de código individuais e percebendo quais os valores obtidos, testando exaustivamente as funcionalidades das bibliotecas envolvidas, não só para obter um maior *know how* sobre as mesmas como também para avaliar se todas as funções executavam bem o seu papel.

Depois é feita a integração dos vários dispositivos e feitos os respectivos testes a estes, ou seja, perceber se existem conflitos, falhas de funcionamento ou volatilidade dos valores obtido. Como o magnetómetro é o elemento principal de todo o sistema, é necessário ter segurança nos valores obtidos através dele, para isso é necessário testar bem a calibração e compensação dos valores medidos, não só recorrendo a *software* já existente como também recorrendo a pequenas rotinas de código que ajudam a efectuar a calibração do magnetómetro. Também são feitos testes mais complexos, nomeadamente com o recurso às bobines de *Helmholtz* do departamento de física existente no ISEL, onde através destas é possível criar um ambiente estável de campo magnético, onde são feitas várias medições para aferir a boa leitura dos valores por parte do sensor.

Numa fase posterior, antes dos voos, seguem-se os testes manuais onde se tenta perceber qual a capacidade do sensor para medir campos magnéticos num determinado local, tentando perceber qual a variação dos mesmos numa determinada área. Onde se efectua a comparação/correlação entre os valores medidos com o magnetómetro implementado e os valores medidos com um magnetómetro de Césio de alta qualidade, permitindo desta forma ter a certeza absoluta de que o sistema implementado funciona bem.

Finalmente, o sistema é integrado no drone e são feitos os primeiros voos para aferir a estabilidade da montagem e para o operador se adaptar às novas características de voo. Passando depois para a fase de execução de missões e de recolha de dados para mapeamento da área a analisar.

Finalizada a primeira fase do planeamento dos testes do sistema, a fase seguinte é a esquematização das medidas manuais, ou seja, estruturam-se todas as medições e definem-se as melhores formas de caracterizar a área escolhida.

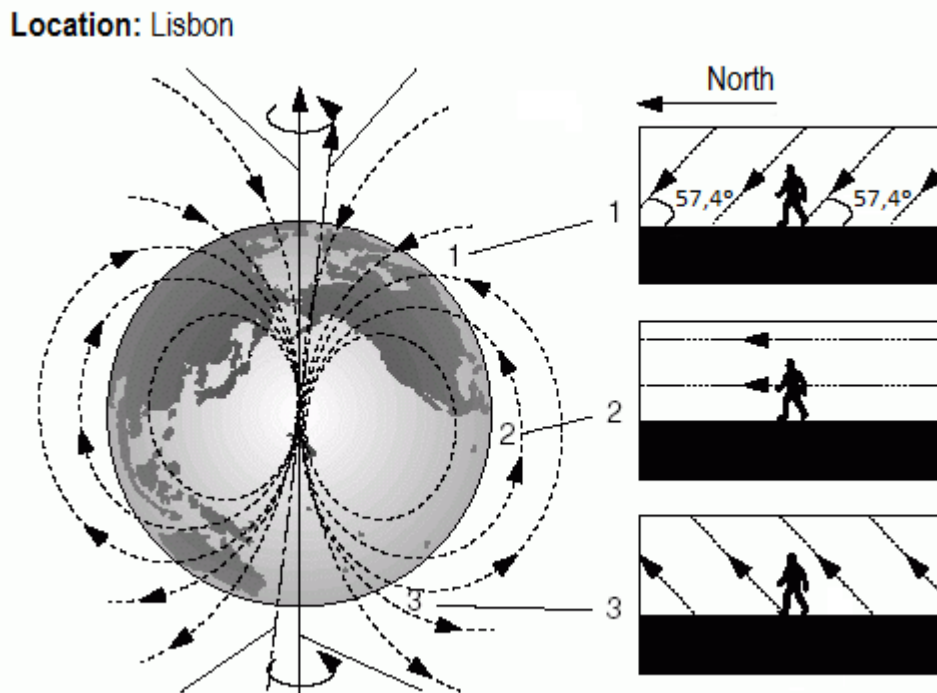


Figura 14 - Ângulo aproximado do campo magnético para a região de Lisboa. [36]

As linhas de campo magnético terrestre fluem de Sul para Norte e na localização escolhida para a realização das medidas o campo magnético flui para um interior da Terra com um inclinação de aproximadamente $57,4^\circ$ (Figura 14). Ou seja, de forma a se maximizar e se obter a melhor qualidade possível dos valores medidos, são idealmente realizadas as medidas no sentido Norte para Sul -> Sul para Norte sem alterar a posição do sensor. Claro que muitas vezes por causa das características do terreno este sentido não é 100% cumprível.

Todas as medidas são feitas a uma frequência de 10 Hz, sendo que a frequência máxima admitida pelo sistema são 80 Hz:

- No sentido Norte para Sul -> Sul para Norte dispendo o sensor no sentido Norte para Sul. As primeiras medidas são feitas em linha recta com a introdução de uma anomalia, mais concretamente um carro, para

ajudar a criar uma fonte de alteração do campo magnético na área medida e para ajudar posteriormente na análise dos dados recolhidos.

- Repetem-se as medidas, recorrendo a um magnetómetro de campo de alta precisão, um magnetómetro de Césio, que possibilita obter medidas de campo magnético para se ter uma base de comparação e para aferir se efectivamente estamos a medir correctamente com o novo sistema.
- Posteriormente, para se alcançar um dos principais objectivos do projecto, nomeadamente, a caracterização de um determinado local/terreno/espço, são feitos mapeamentos 2D da área a caracterizar igualmente no sentido Norte para Sul -> Sul para Norte mantendo a anterior posição do sensor e sem alteração desta. Com estes mapeamentos obtém-se uma caracterização manual da área mas também se estabelece uma base de comparação para a fase seguinte do projecto.
- Os voos, são então uma fase crucial do trabalho, e tal como para os varrimentos manuais, são estabelecidos vários varrimentos de forma a aferir a repetibilidade dos valores medidos no sentido Norte para Sul -> Sul para Norte mas desta vez recorrendo ao drone.
- Finalmente, todos os mapeamentos feitas manualmente são planeados em missões para posteriormente carregar no drone, ou seja, realizam-se os mesmos mapeamentos de Norte para Sul -> Sul para Norte, bem como mapeamentos 2D exactamente nas mesmas condições referidas anteriormente. O último passo é analisar todas as medições, comparar os resultados e verificar as diferenças entre os resultados obtidos, procedendo posteriormente aos ajustes necessários para obter resultados desejáveis.

3. Desenvolvimento e Implementação

Nesta etapa do projecto estrutura-se toda a montagem do sistema, quer através da esquematização de todas as ligações quer com a disposição dos diferentes dispositivos no sistema. Para além disso pensa-se também numa forma eficaz de afastar o magnetómetro o mais possível dos restantes dispositivos, isto para evitar a interferência causada pelos componentes ferromagnéticos e eléctricos destes. Esse afastamento é também importante para a aplicação deste no drone, visto já se saber de antemão que este é um dispositivo a utilizar na execução do projecto e se suspeitar da possível problemática que são os potentes motores deste mas é também importante para a aplicação em qualquer outro dispositivo de específico de medida. A ideia é ter o magnetómetro o mais afastado possível de todas as possíveis fontes de interferências ou distorções.

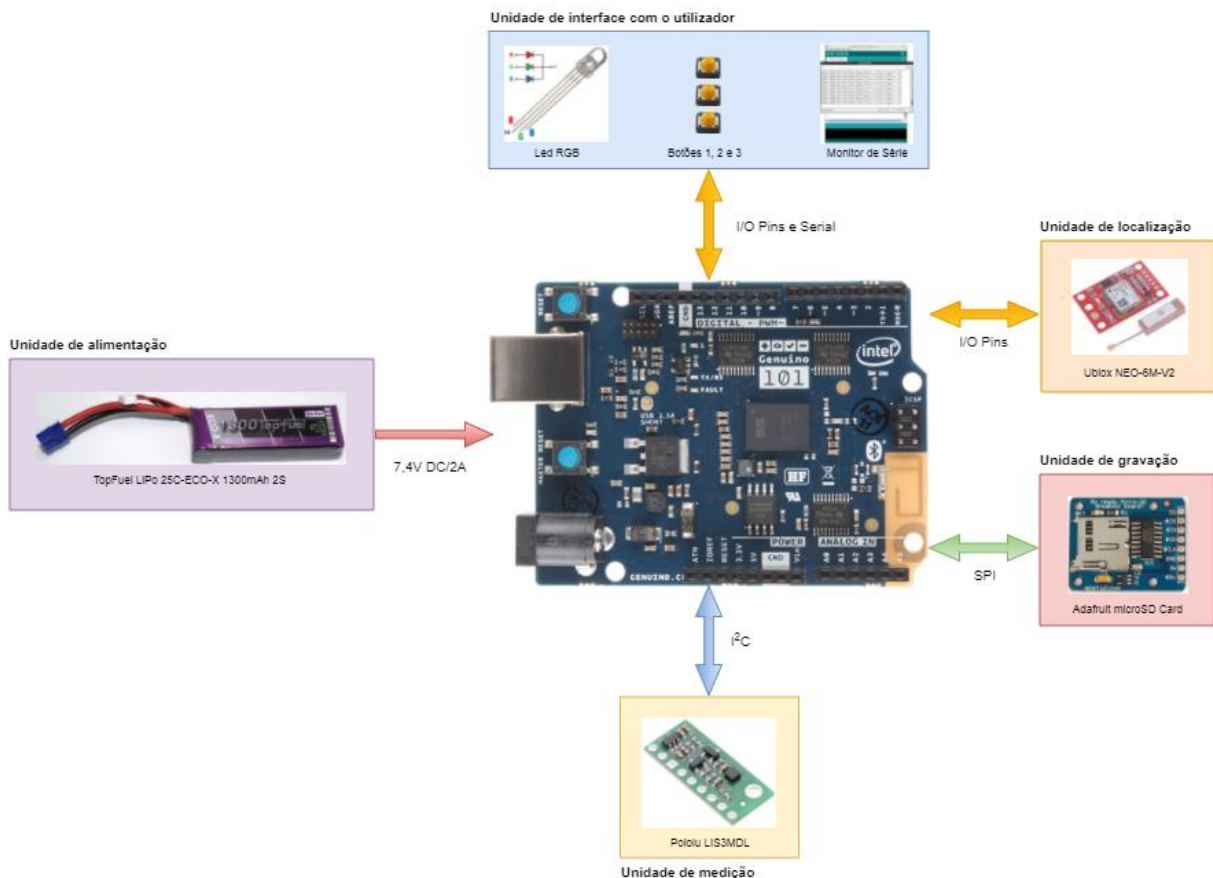


Figura 15 – Diagrama de blocos do sistema.

3.1. Hardware

- Magnetómetro

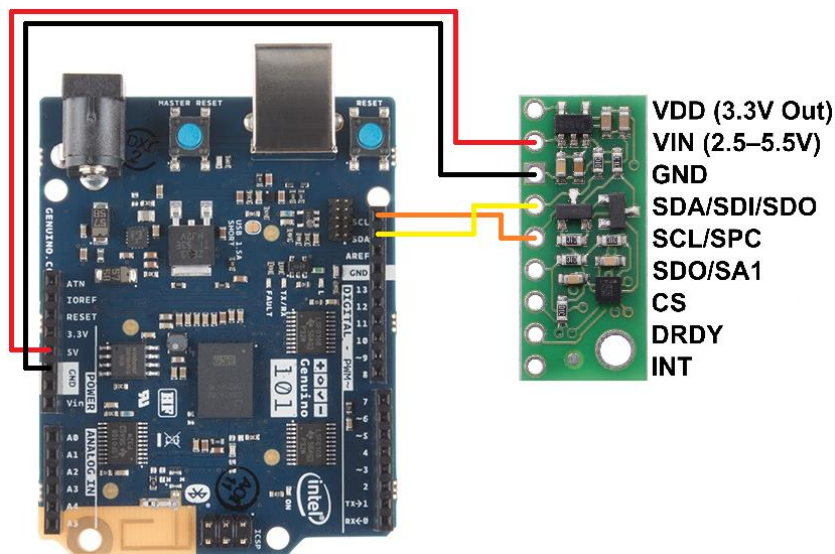


Figura 16 - Ligações do magnetómetro Pololu LIS3MDL de 3 eixos.

Para integrar o magnetómetro no sistema opta-se por utilizar o protocolo de comunicação I²C, e utiliza-se o protocolo SPI para o sistema de gravação de dados. O LIS3MDL possui muitas opções configuráveis, incluindo quatro configurações de sensibilidade (ganho) selecionáveis, de +- 4 Gauss a +- 16 Gauss [19], uma grande variedade de taxas de dados de saída e um sinal de interrupção externo programável. A leitura das três componentes do campo magnético está disponível através de uma das interfaces digitais, neste caso, opta-se pelo I²C. A placa de suporte inclui um regulador de tensão *low-dropout* que fornece os 3.3V exigido pelo LIS3MDL, o que permite que o sensor seja alimentado a partir de uma fonte de 2,5-5,5V. A placa de deslocamento também inclui um circuito que desloca o relógio I²C e as linhas de dados para o mesmo nível de tensão lógica de VIN, tornando simples a interface da placa com sistemas de 5V e o espaçamento de pinos de 0,1cm da placa facilitam a sua utilização com *pinboards* sem solda padrão, *perfboards* e *breadboards*. O LIS3MDL é alimentado com 5V e ligados os seus pinos SDA e SCL aos pinos SDA e SCL do Arduino 101, requisito essencial do protocolo I²C (Figura 16).

- Placa Microcontroladora

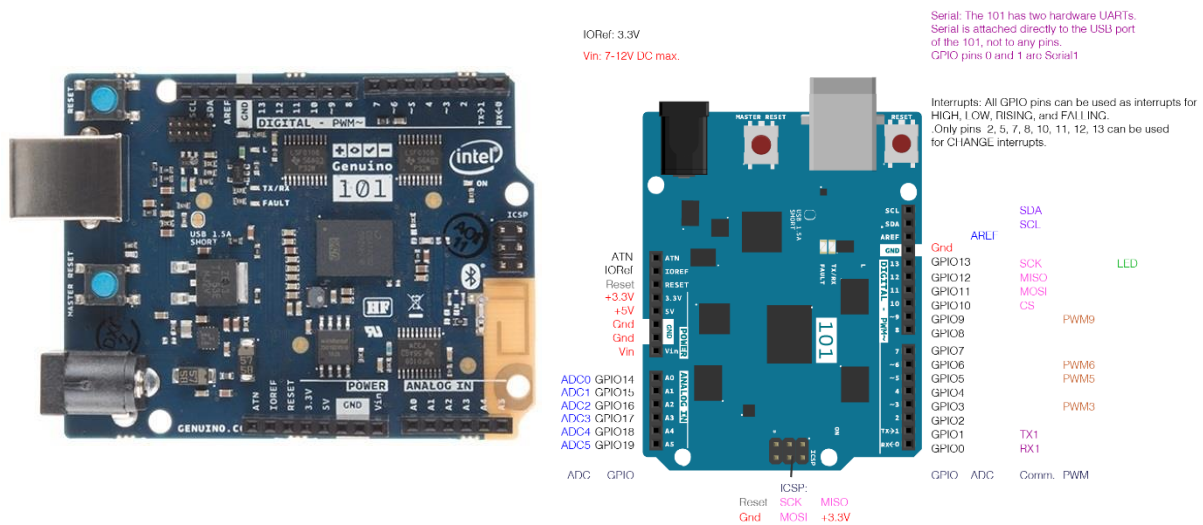


Figura 17 - Placa microcontroladora e *schematic* do Arduino 101 Intel® Curie.

A placa de desenvolvimento Arduino 101 [22] oferece o desempenho e o baixo consumo de energia do módulo Intel® Curie com a simplicidade, robustez e lista de periféricos do Arduino UNO mas com a adição de recursos integrados de *Bluetooth Low Energy*, um acelerômetro e giroscópio de seis eixos. É uma peça fundamental no desenvolvimento deste projecto, porque possibilita realizar as operações mais complexas com a maior simplicidade e com um consumo energético muito reduzido. O módulo contém dois núcleos pequenos, um *x86 Quark* e um núcleo de arquitetura ARC de 32 bits, ambos com *clock* de 32MHz (Figura 17). O Intel Toolchain compila de forma excelente o código Arduino em ambos os núcleos para realizar as tarefas mais exigentes. O Sistema Operacional em Tempo Real (RTOS) e o *framework* desenvolvido pela Intel são de origem aberta o que torna este dispositivo extremamente versátil. O núcleo Arduino comunica com o RTOS através de *mailboxes* estáticas para realizar uma lista predefinida de tarefas (como a interface com o PC usando o USB, programar o código em *flash*, expor a funcionalidade Bluetooth LE, executar o PWM, etc...). O Arduino 101 tem 14 pinos de entrada/saída digitais (dos quais quatro podem ser utilizados como saídas PWM), seis entradas analógicas, um conector USB para comunicação serial e *upload* do código, uma tomada de energia, um cabeçalho ICSP com sinais SPI e pinos dedicados I²C. A tensão de operação da placa e I/O é de 3.3V, mas todos os pinos estão protegidos contra sobretensão de 5V. Todas estas características do Arduino 101 são fundamentais para o sistema, pois providenciam excelente estabilidade ao mesmo, versatilidade e grande qualidade a baixo custo. São

usados o cabeçalho ICSP com sinais SPI e um pino de entrada/saída digital para o *chip select* para acoplar o dispositivo de gravação de dados, os pinos dedicados I²C para o magnetômetro, dois pinos de entrada/saída digitais para o dispositivo de localização, três pinos de entrada/saída digitais para os três botões, três pinos de entrada/saída digitais para o led RGB e finalmente os pinos de *output voltage* de 5V e *ground*.

- **Gravação de dados**

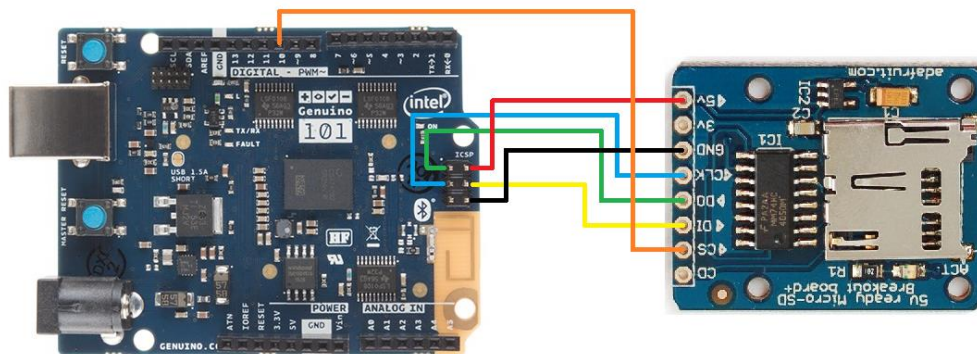


Figura 18 - Ligações do sistema de gravação Adafruit MicroSD card.

Devido à necessidade de gravação dos dados medidos pelo sistema é fundamental ter uma forma simples de guardá-los e a maneira mais simples e eficaz de o fazer é através de um cartão SD, neste caso concreto, microSD. O dispositivo Adafruit MircoSD card [26] permite utilizar duas opções de interface com cartões microSD, modo SPI e modo SDIO. O modo SDIO é mais rápido, mas é mais complexo e requer a assinatura de documentos de compromisso de não divulgação, por essa complexidade inerente, opta-se pelo modo SPI. Cada cartão microSD possui um modo SPI de velocidade ligeiramente inferior que é fácil de usar por qualquer microcontrolador mas deve ser formatado em formato FAT16 ou FAT32 visto que as bibliotecas SD apenas funcionam para estes sistemas de arquivo. Para o modo SPI é usado o cabeçalho ISCP e o pino 10 (CS) de entrada/saída digital para *chip select* (CS) e alimentação é feita com 5V (Figura 18). De forma mais específica e tendo como referência o *schematic* do Arduino 101 (Figura 17), são ligados o pino CLK do dispositivo ao pino CLK do Arduino, o pino DO do dispositivo ao pino MISO do Arduino, o pino DI do dispositivo ao pino MOSI do Arduino e o pino CS do dispositivo ao pino 10 (CS) do Arduino.

- **Localização via GPS**

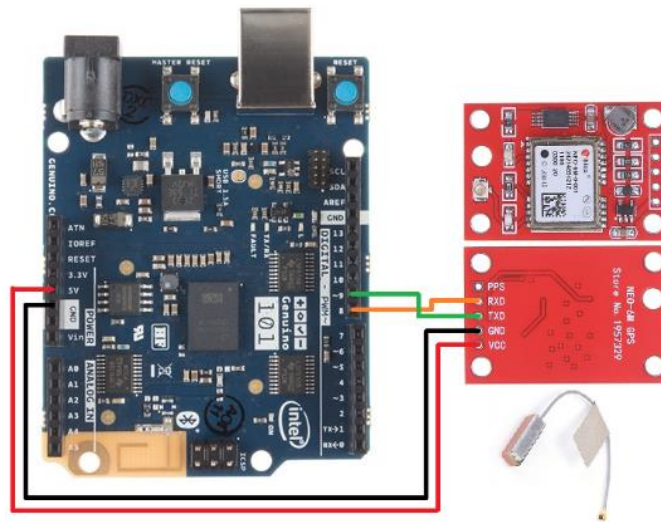


Figura 19 - Ligações do dispositivo de localização Ublox GY-NEO6MV2.

O módulo de GPS utilizado é o Ublox GY-NEO6MV2 da Ublox [28], este dispositivo de tamanho reduzido está inserido numa série de módulos NEO-6 de alto desempenho, com base em informação recebida de seis posições (via satélite). A Ublox 6 é projetado para baixo consumo de energia e baixo custo, esta gestão inteligente de energia é um benefício para aplicações de baixa potência como a que é desenvolvida. Estes receptores combinam um alto nível de capacidade de integração e conectividade flexível. Isto torna-os perfeitamente adequados e como são produzidos em massa e com tamanho reduzido o seu preço é igualmente pequeno mas ainda assim têm uma excelente precisão e são muito fiáveis. O módulo NEO-6M GPS é compatível com APM2 e APM2.5, possuindo uma EEPROM onde são guardadas as configurações necessárias, utiliza uma interface RS232 TTL para comunicação com o Arduino e um *baudrate* padrão é de 9600bps. A alimentação do dispositivo é feita com 5V e ligado o pino RX do dispositivo está o pino 8 do Arduino e o pino TX do dispositivo é ligado ao pino 9 do Arduino (Figura 19).

- **Alimentação do Sistema**



Figura 20 – Baterias TopFuel LiPo 25C-ECO-X 2S e 3S de alimentação do sistema.

A alimentação do sistema está ao cargo da bateria TopFuel LiPo 25C-ECO-X 1300mAh 2S [30] que é uma bateria de alta qualidade e de baixo custo. Pode ser carregada facilmente através do carregador de baterias balanceadas do drone pois dispõem da comumente usada ficha EC3 (Figura 20) e de entrada de balanceamento. Tal como a bateria 2S, a bateria TopFuel LiPo 25C-ECO-X 1800mAh 3S [31] é igualmente uma bateria de muita qualidade mas com uma autonomia consideravelmente superior, o que facilita na execução de medidas de longa duração, onde a ficha de ligação é a mesma, um ficha EC3 com entrada de balanceamento também (Figura 20).

Para além disso, são compactas e como o sistema em si tem um consumo muito reduzido, é possível ter autonomia durante muito tempo entre carregamentos, para além disso têm também um peso reduzido, cerca de 150g e 180g, o que é fundamental para se obter um sistema leve.

- Botões e Led

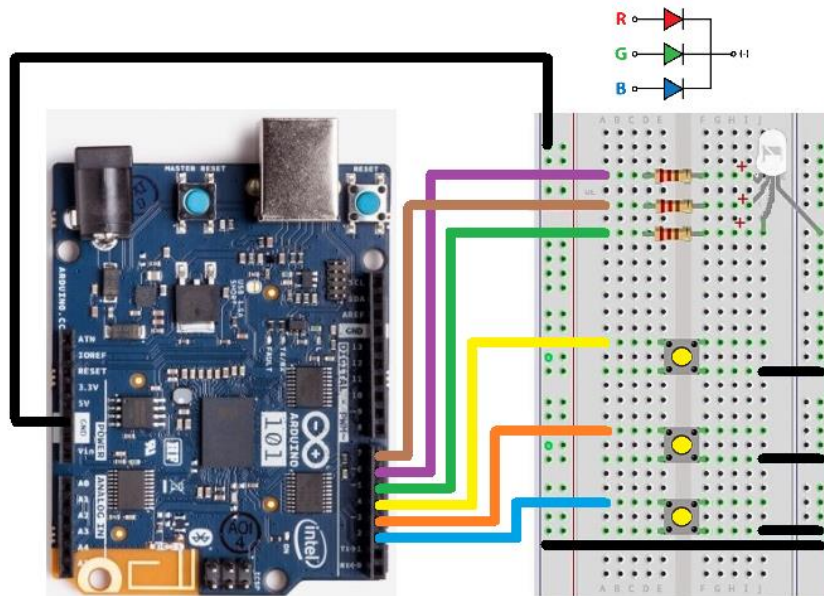


Figura 21 - Ligações dos botões e LED.

Para facilitar a utilização do sistema são adicionados três botões que servem para alternar entre modos de utilização e um led RGB que altera mediante o modo selecionado. Há também o aumento do valor das três resistências necessárias para acoplar o led RGB com o intuito de reduzir a intensidade da cor deste, visto que o mesmo provoca alguma dificuldade de visualização ao utilizador quando necessita de olhar na direção da montagem para operar o sistema. Os botões são do tipo *Momentary Tact*, ou seja, apenas quando pressionados desempenham a sua função, ao largar eles desligam-se (Figura 21).

- **Montagem final**

Depois de definidas todas as ligações necessárias inicia-se o processo de montagem e integração de todos os dispositivos que tem por base o diagrama anterior (Figura 22), para isso, acopla-se ao Arduino a bateria e integra-se a unidade de localização, a unidade de gravação de dados e a unidade de interface com o utilizador. Neste diagrama é possível observar de forma resumida todas as ligações, interfaces e esquematização dos módulos (Figura 22 e Figura 23).

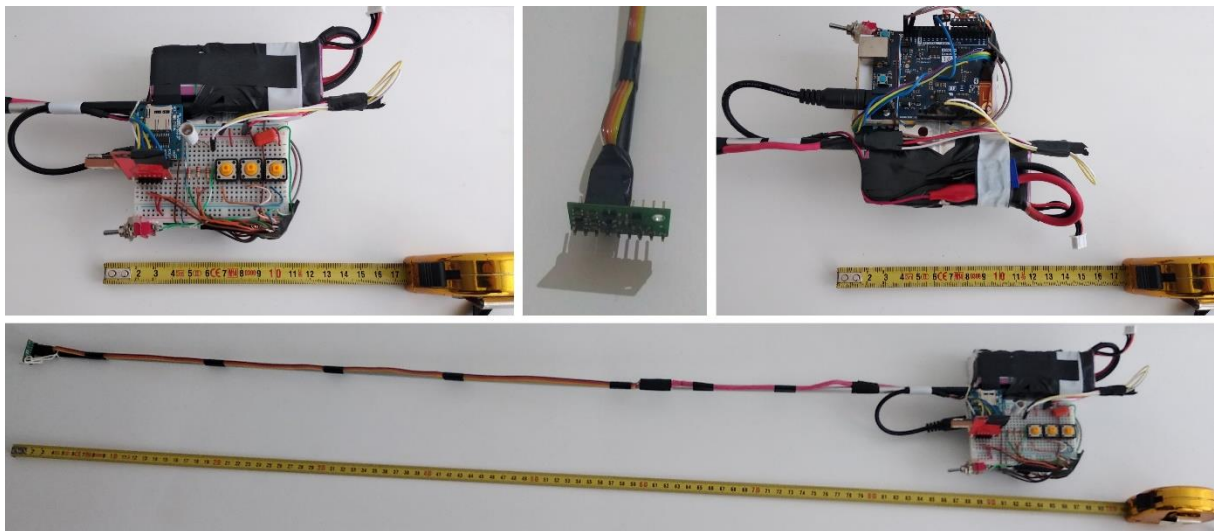


Figura 22 - Acoplamento de todo o sistema à barra de alumínio de um metro.

Para facilitar a utilização do sistema e cumprir o objectivo de afastar o máximo possível o magnetómetro dos restantes componentes para evitar possíveis interferências destes utiliza-se uma vara de alumínio de um metro de comprimento. Numa das pontas foi adicionado o magnetómetro e na outra os restantes componentes do sistema já devidamente acoplados. A barra de alumínio é escolhida porque o alumínio é um metal que não tem propriedades ferromagnéticas, ou seja, não provoca qualquer alteração ao campo em redor do magnetómetro. Para além disso é suficientemente resistente para suportar o sistema e para ser acoplado ao drone de forma simples e robusta. Outra vantagem de montar o sistema desta maneira prende-se com o facto de se poder centrar a parte mais pesada do sistema, numa das pontas da vara de alumínio, facilitando o acoplamento do sistema aos dispositivos

específicos de medida, neste caso concreto ao drone, mantendo desta maneira a estabilidade do mesmo e para além disso mantêm o sensor a aproximadamente 60 centímetros de distância dos dois motores mais próximos do drone.

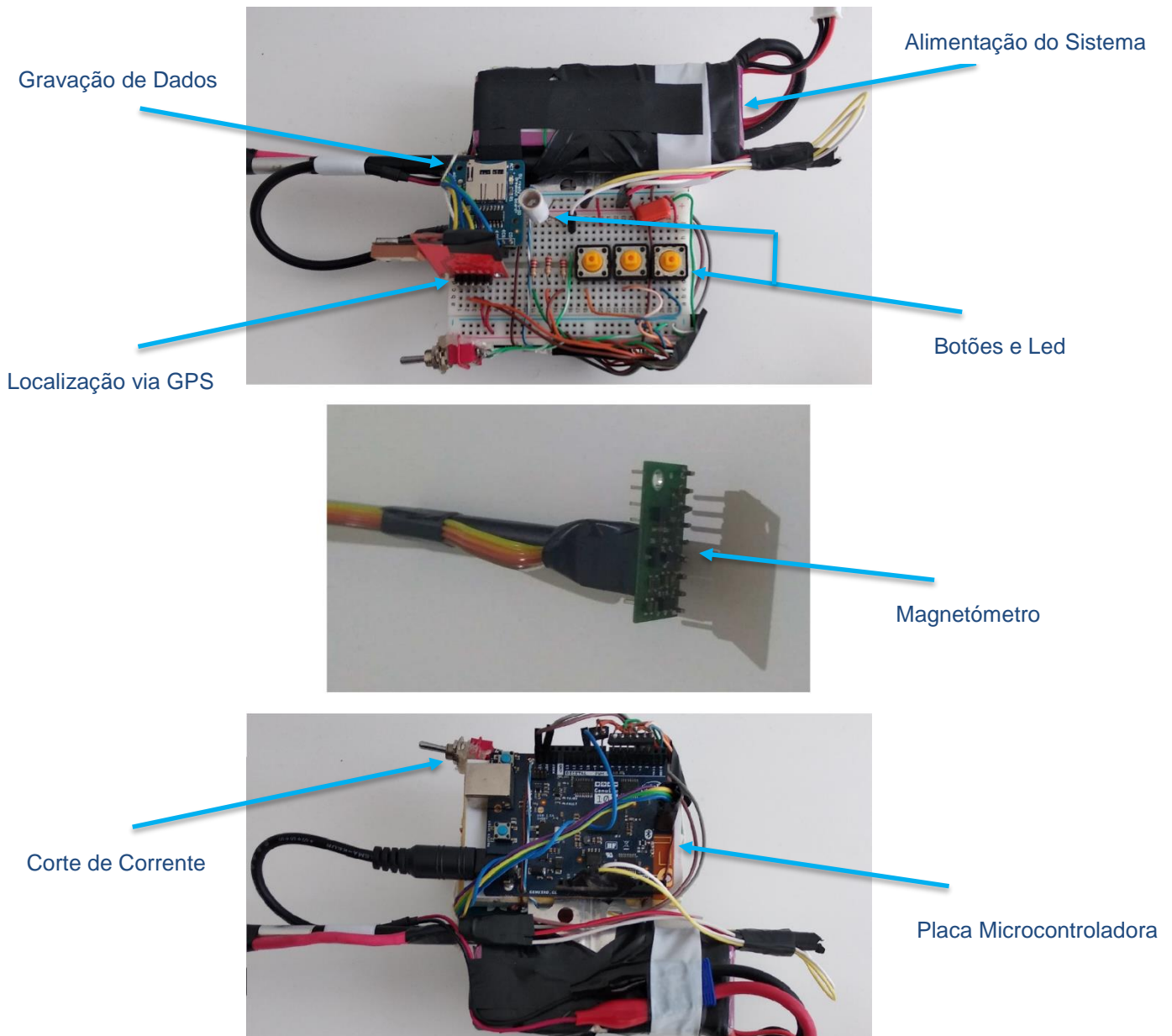


Figura 23 - Descrição da localização de cada dispositivo no sistema.

Como uma das principais funcionalidades do sistema é este poder ser usado de forma independente, para ser utilizado manualmente esta disposição também é muito importante, pois confere uma boa estabilidade ao pegar no sistema e permite que o utilizador tenha junto às mãos os botões de controlo deste. O que permite uma usabilidade muito boa e também um fácil manuseamento (Figura 23).

3.2. Software

Ao construir-se este sistema pretende-se que o mesmo tenha algumas funcionalidades que permitam ao seu utilizador obter valores de campo magnético correctos e que estes representem a realidade. Para isso é necessário que seja possível calibrar o sensor no ambiente de medição e para isso é disponibilizado um modo de compensação que permite obter valores de campo magnético “crus” que são usados posteriormente no processo de calibração escolhido, bastando então que o utilizador accione o botão 1 do sistema.

Para além disso é necessário que o utilizador possa testar a operacionalidade do sistema e verificar se a etapa de compensação foi bem-sucedida, verificando se efectivamente se obtêm valores de campo magnético e se o módulo de localização transmite de forma estável a localização actual das medidas. Para visualizar estes dados é utilizado o monitor de série do IDE do próprio microcontrolador, neste caso o Arduino, e accionando o botão 2, de imediato se tem acesso aos dados medidos e recebidos no ecrã do computador.

Finalmente, accionado o botão 3 do sistema, garantindo que anteriormente já existe estabilidade dos dados, é possível proceder à gravação dos mesmos no cartão microSD, passado assim ao processo de medição e caracterização da área escolhida.

- **Estruturação do código**

Tendo por base uma boa estruturação e leitura do código desenvolvido a estratégia passa por criar o fluxo do mesmo com base num autómato de quatro estados, ou seja, partir o código em vários blocos e métodos criando um software modular. Desta forma permite que qualquer correção, alteração ou adição de funcionalidades seja rápida e intuitiva, para além disso, permite também que qualquer adição ou eliminação de *hardware* seja feita facilmente. Através dos modos STANDBY, COMPENSACAO, TESTES e GRAVAR é possível separar muito bem cada método, isto é, em modos diferentes podem ser chamados os mesmos ou outros métodos sem haver repetição de código. Tendo por base esta estruturação em desenvolvimento de software é possível evitar código repetido e evitar a desorganização deste quando a dimensão desenvolvida se torna muito grande. Esta é uma premissa importante, principalmente quando se trabalha com código Arduino,

onde a memória *flash* deste é limitada e quanto menos ocupação desta for feita melhor, principalmente em sistemas que requerem utilização de muitas bibliotecas e onde algumas delas são muito “pesadas”. Com a construção do autómato é possível realizar várias tarefas ao mesmo tempo, o chamado *multithreading*, paralelizando os vários processos visto que o Arduino 101 apesar de possuir dois núcleos não tem forma de por si só controlar várias tarefas ao mesmo tempo mas em contrapartida permite um bom controlo sobre a temporização, nomeadamente no processo de aquisição dos dados. Permitindo assim que se definam frequências de aquisição muito bem definidas.

Todas as bibliotecas são importadas no início do cabeçalho do *script*, depois são criadas e iniciadas todas as variáveis globais, inicializados os objectos, definidos os pinos de entrada e saída a utilizar e criadas as comunicações série quer com o monitor de série do IDE quer com o dispositivo de localização via GPS. Dentro do *setput()* são inicializados todos os dispositivos e no *loop()* invocado o autómato (Anexo A2).

- Fluxogramas do sistema

Inicialização do sistema, *Setup()* e *Loop()*

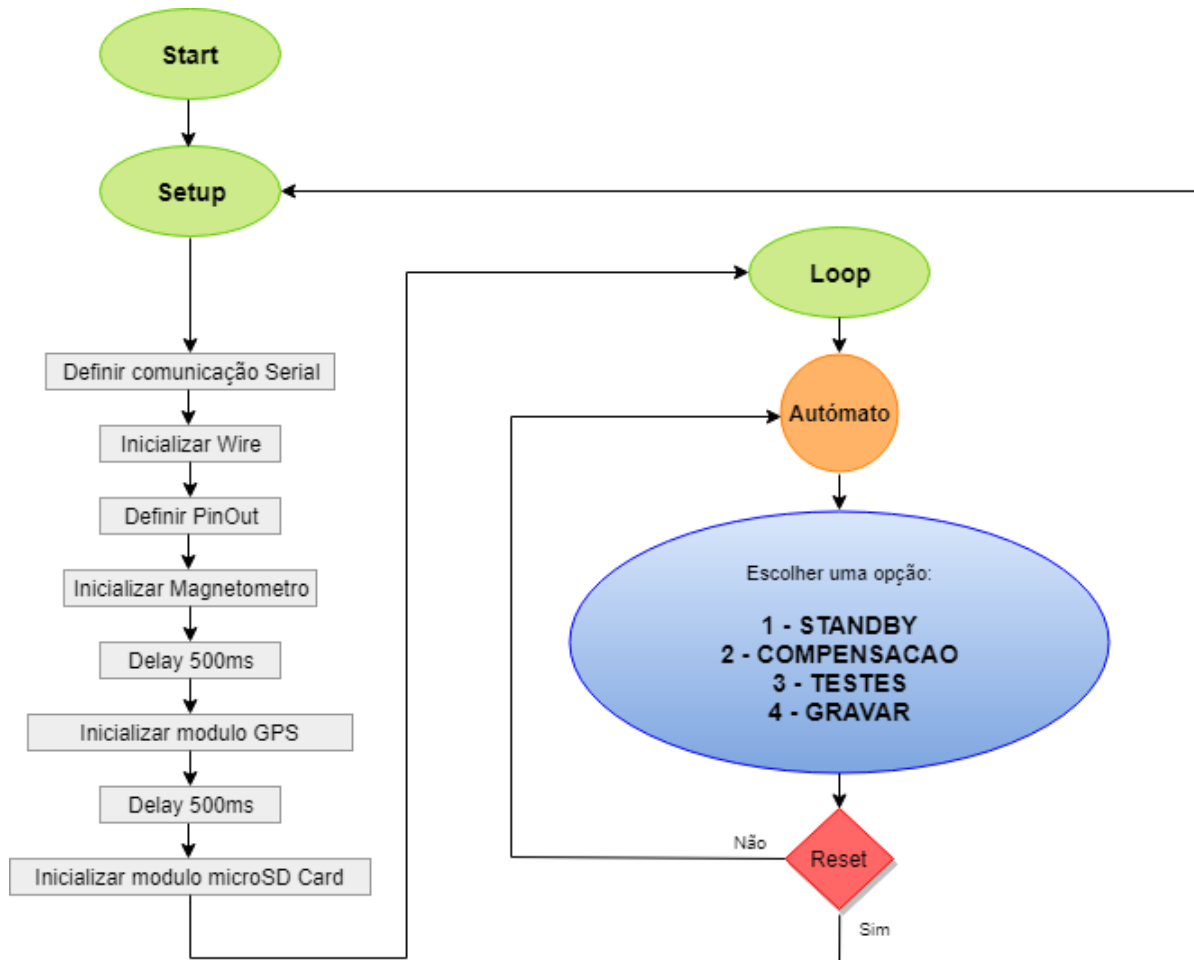


Figura 24 - Fluxograma do *setup()* e *loop()* do sistema.

Inicialmente é feita a importação de todas as bibliotecas necessárias, nomeadamente, a *Wire.h* que possibilita a utilização do protocolo de comunicação I²C e a *LIS3MDL.h* que inclui todas as funcionalidades do magnetómetro Pololu LIS3MDL e que possibilitam o seu funcionamento, a *SPI.h* que possibilita a utilização do protocolo de comunicação SPI e a *SD.h* que inclui todas as funcionalidades de escrita e leitura em cartões microSD por parte do dispositivo Adafruit microSD card e finalmente a *TinyGPS++.h* que inclui todas as funcionalidades do dispositivo de localização o Ublox GY-NEO6MV2 GPS. Posteriormente é feita a inicialização de todas as variáveis globais necessárias, a criação dos objectos específicos, LIS3MDL mag do magnetómetro e TinyGPSPlus tinyGPS do sistema de localização GPS. São

iniciadas as comunicações série, definem-se os *baudrate* destas e é criado o ficheiro de gravação de dados File logFile.

Quanto ao *setup()* é criada a comunicação série *SerialMonitor.begin(SERIAL_BAUD)*, inicializa-se o protocolo I²C *Wire.begin()*, define-se o *PINOUT* dos botões e do led, inicializa-se o magnetómetro *mag.init()* e *mag.enableDefault()*, depois inicializa-se o sistema de localização *ssGPS.begin(GPS_BAUD)*, a seguir inicializa-se o sistema de gravação de dados *SD.begin(ARDUINO_SD_CS)*. Finalmente, depois de todos os dispositivos estarem operacionais inicia-se o *loop()* que lança o autómato *automato()* e inicia o modo *STANDBY* (Figura 24).

Modo STANDBY

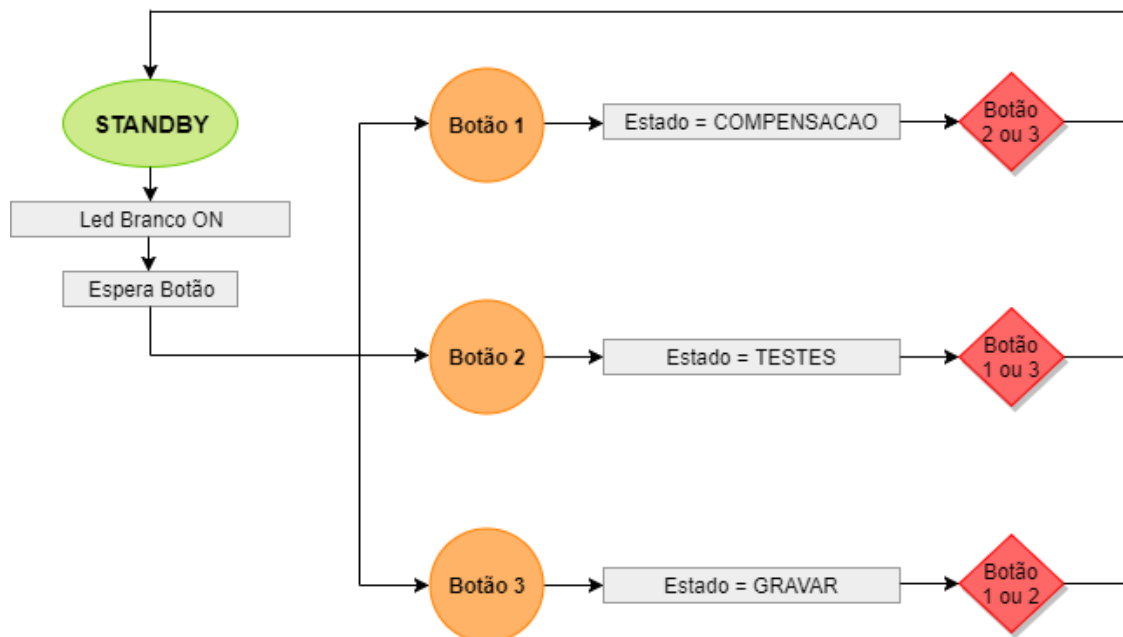


Figura 25 - Fluxograma do modo STANDBY.

Este modo é o mais simples, nele apenas se coloca em funcionamento o led com luz branca, que representa precisamente o modo STANDBY e espera-se a ação de algum dos botões para se iniciar o processo respectivo, ou seja, botão 1 inicia o modo de CALIBRACAO, botão 2 inicia o modo de TESTES e o botão 3 inicia o modo GRAVAR. Esta é também o modo ao qual se regressa sempre que se termina a execução de qualquer um dos restantes modos, iniciando-se novamente processo de espera por qualquer um dos botões (Figura 25).

Modo COMPENSACAO

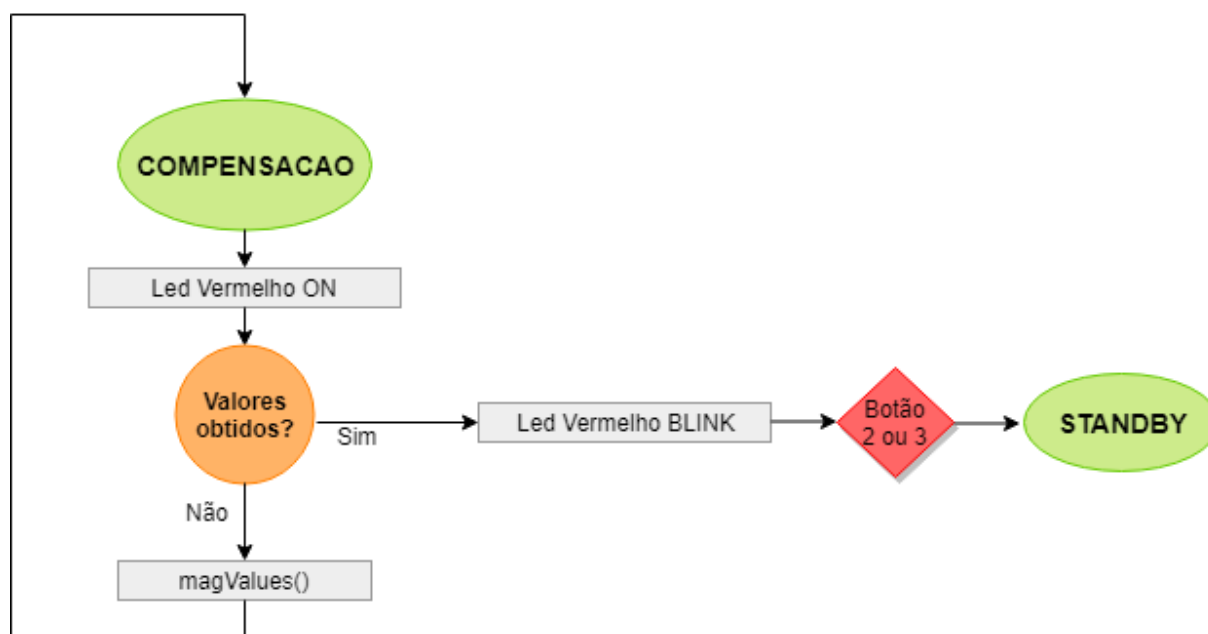


Figura 26 - Fluxograma do modo COMPENSACAO.

No modo COMPENSACAO assim que se inicia o processo o led passa a vermelho e inicia-se também a obtenção de valores, através do magnetómetro Pololu LIS3MDL e da função *magCalibration()*, até se atingirem os valores suficientes, para a posterior obtenção da matriz de calibração e do *bias*. Assim que os pontos necessários são obtidos o led vermelho começa a piscar e podem então carregar-se os valores na rotina de calibração e efectuar-se o cálculo da matriz de compensação e do *bias*, sendo estes posteriormente carregados na aplicação. Assim que o led vermelho passa para o modo intermitente é nesse momento que o utilizador deve voltar ao modo STANDBY, clicando num dos botões 2 ou 3, podendo depois de efectuar a calibração, “chamar” qualquer um dos modos, novamente a COMPENSACAO se houver necessidade de recalibração ou então os modos TESTES ou GRAVAR (Figura 26).

Inicialmente neste modo a ideia era através dele proceder-se à calibração do magnetómetro sem se recorrer a dispositivos externos, neste caso a um computador, mas esta abordagem acabou por não se consumir visto que as bibliotecas existentes para o cálculo matricial são bastante pesadas e ocupam muita memória, limitando desta forma a construção do autómato e a implementação das funcionalidades necessárias para o sistema. Optou-se então por obter apenas os valores não

calibrados e transferir esta funcionalidade para o computador, reduzindo assim a complexidade do desenvolvimento no Arduino.

Modo TESTES

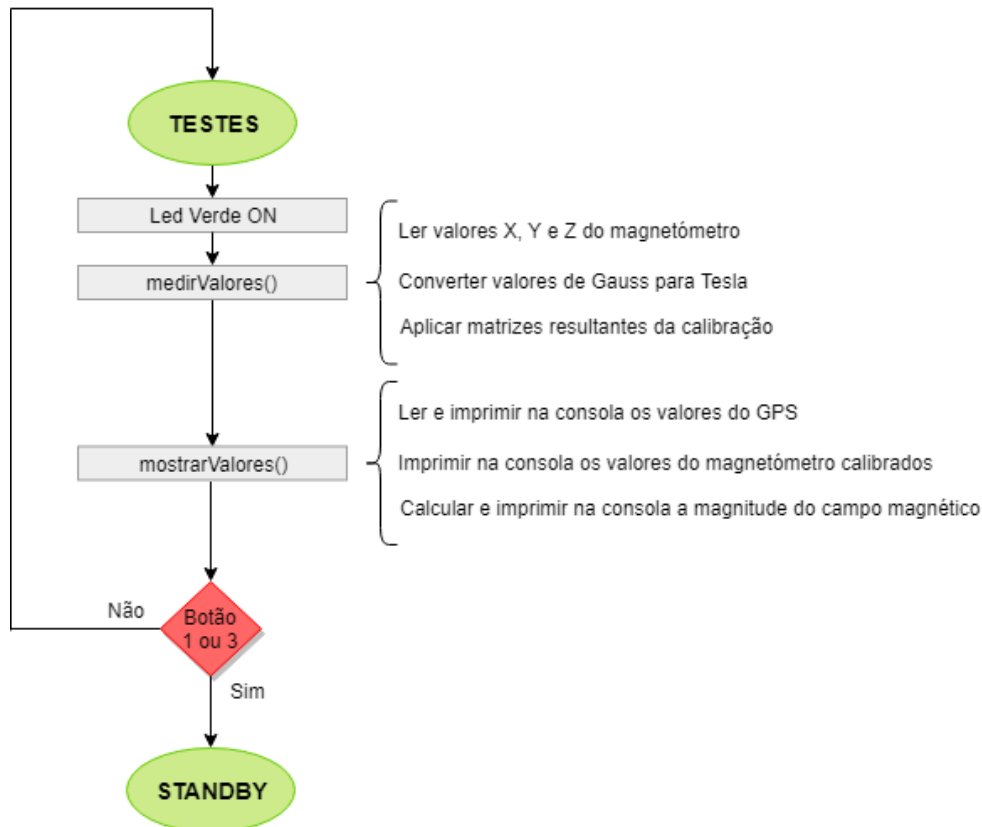


Figura 27 - Fluxograma do modo TESTES.

No caso do modo TESTES assim que o processo se inicia o led é colocado a verde e inicia-se de imediato a medição de valores através do método *medirValores()*, que recorrendo ao método *mag.read()* obtêm os valores das três componentes de campo magnético X, Y e Z. A seguir é feita a conversão desses valores para a unidade do sistema internacional que representa o campo magnético, ou seja, para Tesla, converte-se os valores de LSB/Gauss para Tesla. Depois aplica-se a estes valores a matriz de compensação e *bias* calculados anteriormente, corrigindo desta forma os valores lidos de acordo com as características do local a caracterizar. De seguida é chamado o método *mostrarValores()* que trata de imprimir na consola os valores correspondentes à localização actual através do dispositivo Ublox GY-NEO6MV2 GPS

e da maioria das funções que compõe a biblioteca TinyGPS++, conjugando as mesma com os valores de campo magnético lidos anteriormente. Neste modo aguarda-se sempre por um dos botões 1 ou 3 para se voltar ao modo STANDBY e aguardar novamente o início de qualquer um dos modos (Figura 27).

Modo GRAVAR

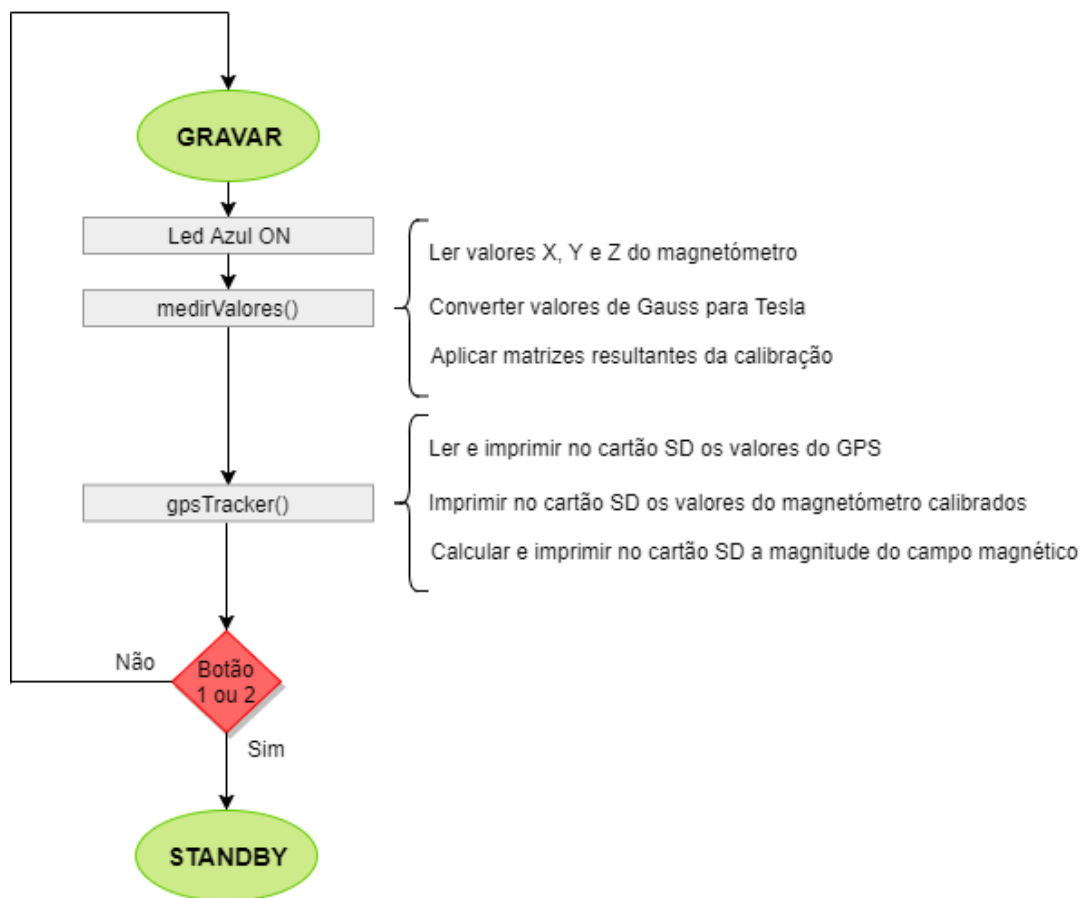


Figura 28 - Fluxograma do modo GRAVAR.

Finalmente, no modo GRAVAR, o led, assim que se inicia o processo passa a azul e tal como no modo TESTES o método `medirValores()` é chamado para obter os valores lidos pelo magnetómetro, para efectuar a conversão dos valores para Tesla e para aplicar o algoritmo de calibração. Depois disso é chamada a função `gpsTracker()` que para além de obter todos os valores da localização novamente através da biblioteca TinyGPS++ efectua também a gravação de todos os valores no cartão microSD através do dispositivo Adafruit microSD card e da criação do ficheiro através

da função `logFile = SD.open(logFileName, FILE_WRITE)`, da escrita do `header` e escrevendo no seu interior os referidos valores com a função `logFile.print(...)`. Esta gravação só se processa quando os valores de localização são estáveis e se verifica a validade dos mesmos. Neste modo é extremamente importante que todos os valores estejam estáveis e que sejam fidedignos, visto que este modo será o mais importante aquando da realização de trabalhos de campo e das respectivas medidas de campo magnético. Tal como nos modos anteriormente referidos espera-se que haja o acionamento de um botão, neste caso o botão 1 ou 2 para que se volte ao modo STANDBY e possibilitar o reinício de qualquer um dos processos (Figura 28).

4. Calibração

4.1. Modelo de calibração

A calibração do magnetómetro é sem dúvida o processo mais importante para que os valores obtidos sejam correctos, isto porque magnetómetros de custo reduzido como o Pololu LIS3MDL não recebem qualquer tipo de calibração de fábrica, são produzidos em massa e pressupõem que o utilizador final irá tratar do processo de calibração do mesmo.

A representação das medições obtidas através de um sensor magnético como o Pololu LIS3MDL não calibrado são caracterizadas por terem a forma de um elipsoide e por não estarem centradas na origem do referencial cartesiano. Um elipsoide é descrito por um polinómio de segundo grau em x , y e z , representado pela equação implícita geral de uma superfície quadrática S (Equação 2) [35].

$$S: ax^2 + by^2 + cz^2 + 2fxy + 2gxz + 2hyz + 2px + 2qy + 2rz + d = 0$$

Equação 2 – Equação geral que descreve um elipsoide.

Este elipsoide também pode ser descrito de forma semi-matricial, o que é especialmente útil para o problema em questão (Equação 3).

$$S: x^T M x + x^T n + d = 0$$

Equação 3 – Representação semi-matricial que descreve um elipsoide.

Onde x é:

$$x = [x \ y \ z]^T$$

e M , n são, respectivamente:

$$M = \begin{bmatrix} a & h & g \\ h & b & f \\ g & f & c \end{bmatrix} \text{ e } n = \begin{bmatrix} p \\ q \\ r \end{bmatrix}$$

Após o processo de calibração é espectável que a representação das medições obtidas seja caracterizada por uma esfera de raio F centrada na origem que também é representada matricialmente como (como exemplo assume-se $F = 1$):

$$M = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, n = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \text{ e } d = -1$$

Mas o elipsoide, esfera depois de calibrado, também pode ser descrito numa forma matricial pura.

$$S: x^T Q x = 0$$

Equação 4 – Representação matricial que descreve um elipsoide.

Onde x e Q são, respectivamente:

$$x = [x \ y \ z \ 1]^T \text{ e } Q = \begin{bmatrix} M & n \\ n^T & d \end{bmatrix}$$

A representação da superfície quadrática é determinada por algumas propriedades das matrizes M e Q [35].

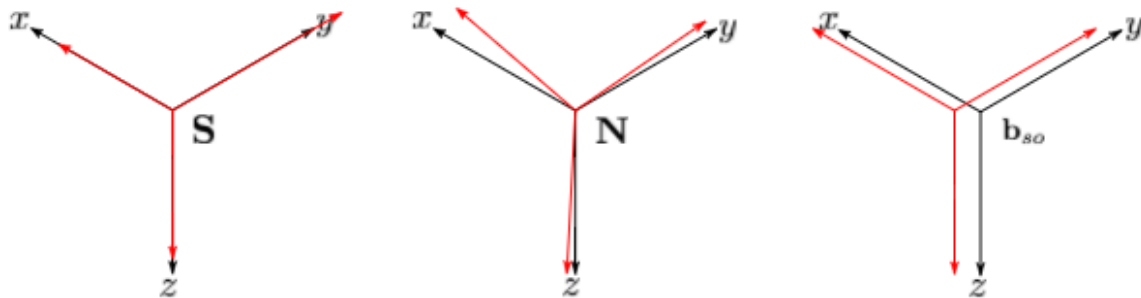


Figura 29 - Esquematização dos erros de instrumentação. [35]

Depois de detalhada a representação das medições obtidas são especificados os erros de instrumentação que são únicos e constantes para cada dispositivo, eles podem ser modelados como resultado de três componentes [35]. O primeiro é o factor de escala, que pode ser modelado como uma matriz diagonal S (Equação 5).

$$S = \begin{bmatrix} S_x & 0 & 0 \\ 0 & S_y & 0 \\ 0 & 0 & S_z \end{bmatrix}$$

Equação 5 - Matriz diagonal que representa o erro de factor de escala.

O segundo é a não-ortogonalidade dos eixos do sensor, que pode ser modelada como N (Equação 6).

$$N = [n_x \ n_y \ n_z]$$

Equação 6 - Matriz que representa o erro de não-ortogonalidade dos eixos do sensor.

Onde cada coluna representa um vector de tamanho 3 correspondente a cada eixo do sensor em relação à estrutura do sensor. Finalmente, o terceiro é o deslocamento do sensor que pode ser simplesmente modelado como b_{so} (Equação 7).

$$b_{so} = [b_{so_x} \ b_{so_y} \ b_{so_z}]^T$$

Equação 7 – Matriz que representa o erro devido ao deslocamento do sensor.

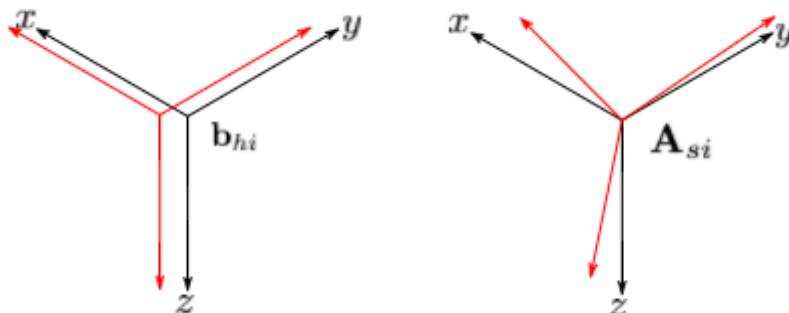


Figura 30 – Esquematização das interferências magnéticas. [35]

A medição do campo magnético será submetida a distorção e existem duas categorias para essas distorções: as distorções de magnetismo permanente conhecido como *hard iron* e as distorções de magnetismo induzido conhecido como *soft iron* [35]. Os erros de *hard iron* referem-se à presença de campos magnéticos em torno do sensor (ímãs permanentes e histerese magnética (indução magnética provocada de ferro magnetizado), fios de alimentação, componentes, etc...) e estão relacionados com erros de compensação de medição que podem ser modelados por b_{hi} (Equação 8).

$$b_{hi} = [b_{hi_x} \ b_{hi_y} \ b_{hi_z}]^T$$

Equação 8 – Matriz que representa a distorção provocada por *hard iron*.

Os erros de *soft iron* referem-se à presença de materiais ferromagnéticos ao redor do sensor, que distorcem a densidade do campo magnético da Terra localmente e estão relacionados com erros de compensação de escala, causando uma mudança na intensidade e direção do campo detectado que pode ser modulada por A_{Si} (Equação 9).

$$A_{Si} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix}$$

Equação 9 – Matriz que representa a distorção provocada por *soft iron*.

Depois de detalhadas todas as fontes de distorção, pode escrever-se facilmente o modelo de medição. Todas as fontes de distorção descritas em Equação 5, Equação 6, Equação 7, Equação 8 e Equação 9 podem ser combinadas para expressar o campo magnético medido h_m (Equação 10) [40].

$$h_m = SN(A_{Si}h + b_{hi}) + b_{so}$$

Equação 10 – Campo magnético medido combinado com todas as fontes de distorção.

Onde h é o verdadeiro campo magnético, assumindo que estamos num ambiente livre de perturbações magnéticas e que temos um magnetómetro ideal de três eixos a leitura do magnetómetro tomada uma orientação arbitrária é dado por h (Equação 11) [40].

$$h = R_x(\phi)R_y(\theta)R_z(\varphi)h_0$$

Equação 11 – Campo magnético ideal livre de perturbações magnéticas.

Onde h_0 é o campo magnético terrestre local (Equação 4) e $R_x(\phi)$, $R_y(\theta)$ e $R_z(\varphi)$ são matrizes de rotação em torno dos eixos de referência x , y e z , respectivamente. As amostras obtidas formam então uma esfera de raio F , onde F^2 é dado por:

$$F^2 = h^T h$$

Equação 12 – Quadrado da intensidade do campo magnético.

É necessário ressaltar que este modelo de medição não inclui ruído estocástico e isso é de alguma forma pouco realista, no entanto, simplifica a abordagem da solução assumindo de antemão um ruído residual de amplitude. De qualquer das formas a Equação 11 pode ser reescrita como:

$$h_m = Ah + b$$

Equação 13 - Reescrita da Equação 11.

Onde:

$$A = SNA_{si}$$

$$b = SNb_{hi} + b_{so}$$

A matriz A combina os fatores de escala, desalinhamentos e efeitos de *soft iron*, enquanto b é o vector de polarização *bias*. Desta forma a transformação linear de h demonstra que as medidas obtidas h_m formam um elipsoide e assim sendo os processos de calibração resumem-se a um problema de ajuste de elipsoides [40].

4.2. Processos de calibração

Para se obterem os dados correctos do campo magnético, devem-se obter os dados e calibrá-los devidamente, obtendo o vector *bias* e a matriz de compensação A e aplicando-os ao vector do campo magnético não calibrado obtido através do magnetómetro (coordenadas X, Y, Z).

A partir da análise da Equação 13 verifica-se que é necessário encontrar uma estimativa de A e b , a que chamamos \hat{A} e \hat{b} , então qualquer nova medida pode ser calibrada como \hat{h}_n .

$$\hat{h}_n = \hat{A}^{-1}(h_n - \hat{b})$$

Expressando então h usando a Equação 13 como:

$$h = A^{-1}(h_m - b)$$

Que combinada com a Equação 12 permite obter (Equação 14):

$$F^2 = h_m^T A^{-T} A^{-1} h_m - 2h_m^T A^{-T} A^{-1} b + b^T A^{-T} A^{-1} b$$

$$\Leftrightarrow$$

$$h_m^T A^{-T} A^{-1} h_m - 2h_m^T A^{-T} A^{-1} b + b^T A^{-T} A^{-1} b - F^2 = 0$$

Equação 14 – Equação resultante da combinação da Equação 13 com a Equação 12.

A Equação 14 pode ser reescrita para uma forma quadrática conhecida.

$$h_m^T M h_m + h_m^T n + d = 0$$

Onde:

$$M = A^{-T} A^{-1}$$

$$n = 2A^{-T} A^{-1} b$$

$$d = b^T A^{-T} A^{-1} b - F^2$$

Como demonstrado anteriormente, através das medições sem calibração é obtida uma superfície quadrática com o formato de um elipsoide, assim sendo, é necessário usar um algoritmo que o ajuste e que forneça estimativas dos parâmetros da superfície quadrática. Múltiplos algoritmos foram desenvolvidos para ajustar um conjunto de pontos, como é o caso do método desenvolvido pelo TESLABS [35] com base no artigo de Qingde Li e J.G. Griffiths [41].

Usando a Equação 14, as reescritas da mesma e usando como valores estimados \hat{M} , \hat{n} e \hat{d} é possível encontrar \hat{A} ou \hat{A}^{-1} e \hat{b} . É necessário então encontrar a escala adequada para a solução, descrita em [42].

$$\hat{b} = -\hat{M}^{-1}\hat{n}$$

$$\hat{A}^{-1} = \frac{F}{\sqrt{\hat{n}^T \hat{M}^{-1} \hat{n} - \hat{d}}} \hat{M}^{\frac{1}{2}}$$

Para se proceder à calibração pode-se recorrer a vários métodos existentes, implementados de várias maneiras quer em Matlab, em C, C# ou Fortran. Mas todas elas se resumem ao ajuste de elipsoides, apenas variam os processos para se atingir esse objectivo, e tal como referido anteriormente, através da calibração é possível obter a matriz de compensação e o vector *bias* que permite efectuar a correção de todas as problemáticas que caracterizam cada local.

Inicialmente para qualquer dos métodos apresentados a seguir é necessário obter valores “crus” (não calibrados), para os três primeiros basta obter valores arbitrários tentando que estes represente todas as direções dos vários eixos XYZ, basta para isso que se gire o sensor nos vários sentidos de cada eixo, em troco de si próprio. Para o último método a obtenção dos valores tem de ser feita de maneira mais cuidadosa e estruturada mas esse procedimento é devidamente explicado mais à frente.

Tabela 6 – Resumo dos processos de calibração.

Processo de Calibração	Tecnologia	Obtenção de dados	Duração
ST Microelectronics	Matlab	Arbitrária	5 a 10 minutos
TESLABS	Matlab	Arbitrária	5 a 10 minutos
Magneto	Fortran e C	Arbitrária	5 a 10 minutos
MagMaster	C#	12 Pontos	10 a 15 minutos

- **Calibração no Matlab**

Depois de obtidos e de guardados num ficheiro, mag.txt ou mag.csv, valores não calibrados, através do sistema desenvolvido, o primeiro método de calibração utilizado é o método fornecido pelo fabricante do sensor magnético incorporado no Pololu LIS3MDL, a ST Microelectronics disponibiliza uma rotina em Matlab que consiste no ajuste de um elipsoide não girado a um conjunto de dados XYZ, calculando os valores e vectores próprios da matriz de dados obtidos para a calibração.

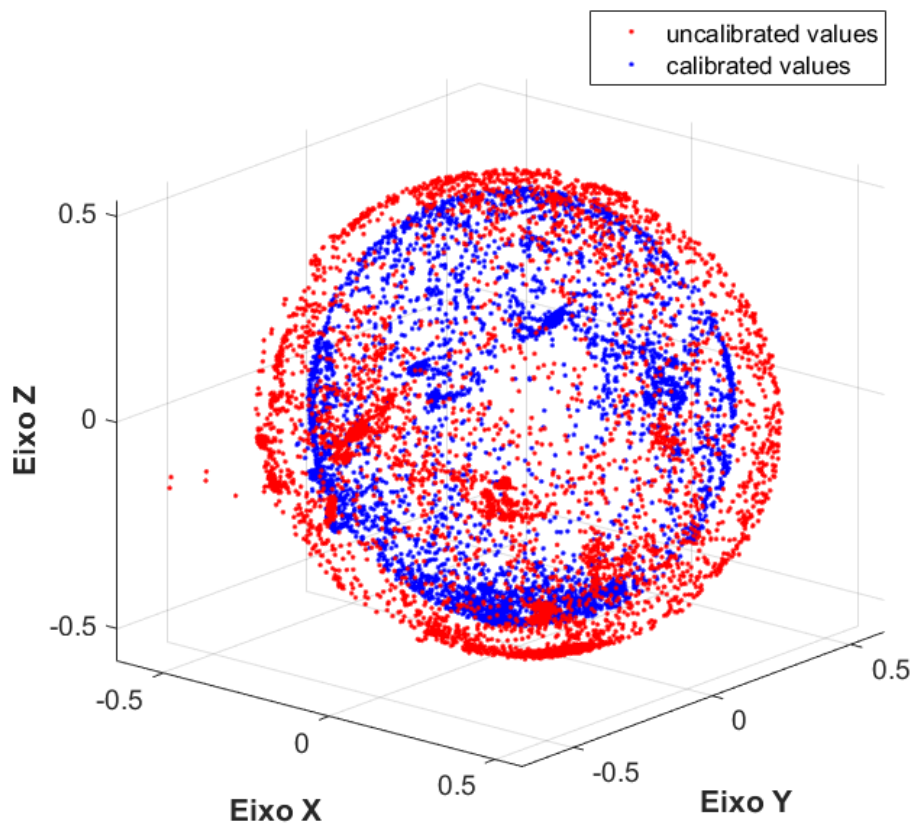


Figura 31 - Comparação entre a Pré e Pós calibração para o método ST Microelectronics.

Após a execução da rotina de calibração é possível observar a grande diferença, os pontos a vermelho caracterizam os dados lidos sem calibração e os pontos a azul correspondem ao ajuste após a aplicação da matriz de compensação e vector *bias* (Figura 31). As diferenças são muito significativas na medida em que para os pontos a vermelho é possível observar as distorções que estes sofreram devido às características do local e após a calibração essas distorções desaparecem, para além

disso existe uma centralização do elipsoide na origem do referencial cartesiano, tornando-o mais compacto e homogêneo.

Com este método de calibração são obtidos valores para a matriz de compensação e vector *bias* que compensarão todas as problemáticas do local em questão e são os seguintes:

$$\hat{A}^{-1} = \begin{bmatrix} 0.840620859102062 & 0.541617919327897 & -0.00256918380625702 \\ 0.540364006801855 & -0.838332479170017 & 0.0721484200914884 \\ -0.0369230469430041 & 0.062037761335923 & 0.997390597896768 \end{bmatrix}$$

$$\hat{b} = [0.61157250630141 \quad 0.583555100088783 \quad 0.553729688292507]$$

Para além do método sugerido pela ST Microelectronics existem outros igualmente eficientes e para se poder aferir essa eficiência executam-se todos eles e comparam-se os resultados obtidos, sendo que no final e mediante o tipo de medida a fazer seleciona-se aquele que se adequa mais às necessidades.

Posto isto, outro dos métodos que foi desenvolvido é precisamente o construído pelo TESLABS [35] com base no artigo de Qingde Li e J.G. Griffiths, que consiste na rotação e ajuste de um elipsoide, este foi criado usando a linguagem de programação

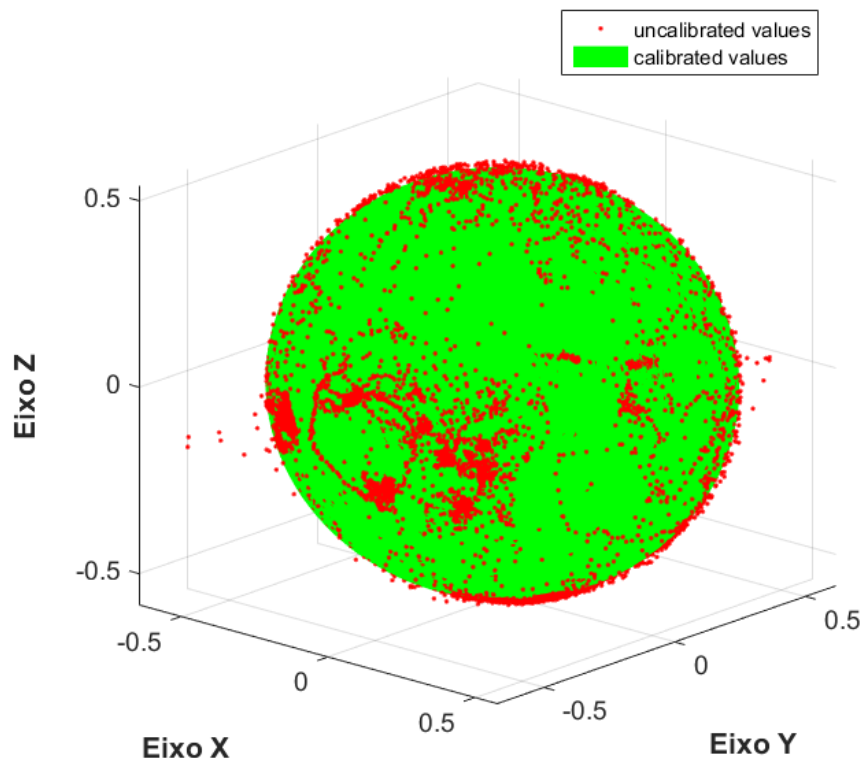


Figura 32 - Comparação entre a Pré e Pós calibração para o método TESLABS (Qingde Li e J.G. Griffiths).

Phyton e também Matlab e os resultados obtidos são também bons e muito parecidos com os obtidos através do método da ST Microelectronics.

Neste caso concreto os pontos a vermelho continuam a representar os valores lidos sem calibração e a superfície esférica verde representa o ajuste o elipsoide, sendo este também centrado na origem e ajustadas as distorções características do local de medida (Figura 32).

Deste método de calibração obtém-se os seguintes valores para a matriz de compensação e vector *bias*:

$$\hat{A}^{-1} = \begin{bmatrix} 0.77771 & 0.62392 & -0.076749 \\ 0.62388 & -0.78104 & -0.027363 \\ 0.077016 & 0.026601 & 0.99667 \end{bmatrix}$$

$$\hat{b} = [0.60434 \quad 0.58132 \quad 0.54852]$$

Apesar de este método devolver resultados com menos casas décimas em comparação com o primeiro, os valores obtidos são muito próximos, principalmente no vector *bias*, para a matriz de compensação as diferenças são maiores mas isto prende-se pelo facto de este ser um ajuste variável e que depende da rotação que é feita ao elipsoide, gerando desta forma diferenças mais significativas entre os vários métodos.

- **Calibração no Magneto**

O Magneto é um *software open source* específico para efectuar a calibração de magnetómetros como o Pololu LIS3MDL que não vêm calibrados de fábrica, desenvolvido em linguagem C e também em Fortran, que utiliza a mesma técnica implementada pelo TESLABS, conhecida como, "*Li's ellipsoid specific fitting algorithm*", que tal como referido realiza a rotação e ajuste de um elipsoide [51]. Neste caso, e tal como nos métodos anteriores, é utilizado um ficheiro com os valores não calibrados, previamente obtidos, para efectuar a calibração. No caso da versão em linguagem C basta executar através da consola CMD, usando o compilador gcc para compilar o código. Depois basta executar o ficheiro gerado e é mostrado no ecrã o resultado da calibração, nomeadamente a matriz de compensação e o vector *bias* (Figura 33).

```
C:\Windows\system32\cmd.exe
C:\Users\vasr88\OneDrive\Valter\mag\calib\merlin_magneto_c_source_code_and_app>gcc merlin.c -o merlin
C:\Users\vasr88\OneDrive\Valter\mag\calib\merlin_magneto_c_source_code_and_app>.\merlin
-0.021659
0.013250
-0.026167
0.956973 -0.017809 0.006564
-0.017809 0.964533 0.003304
0.006564 0.003304 1.036145
C:\Users\vasr88\OneDrive\Valter\mag\calib\merlin_magneto_c_source_code_and_app>.\merlin_
```

Figura 33 - Execução do *software* Magneto (Linguagem C).

Após a execução a matriz de compensação e o vector *bias* são:

$$\hat{A}^{-1} = \begin{bmatrix} 0.956973 & -0.017809 & 0.006564 \\ -0.017809 & 0.964533 & 0.003304 \\ 0.006564 & 0.003304 & 1.036145 \end{bmatrix}$$

$$\hat{b} = [-0.021659 \quad 0.013250 \quad -0.026167]$$

Estes valores obtidos são diferentes dos obtidos através dos métodos anteriores, mas mais uma vez tem a ver com facto de ser possível realizar a rotação do elipsoide de várias maneiras e isso leva a que os valores tenham algumas diferenças entre si mas o resultado final é o mesmo.

Posteriormente, com o objectivo de se comparar os vários métodos, executa-se o *software* Magneto desenvolvido na linguagem de programação Fortran, coloca-se neste o valor aproximado do campo magnético local, ou seja, o valor característico do local de medida e carrega-se também o ficheiro com os valores não calibrados, já utilizados e medidos anteriormente. Os valores obtidos são exactamente iguais aos que se obtém através da rotina desenvolvida em C, o que permite verificar que tanto um como o outro modo de execução do método são fidedignos e representam o resultado final de forma precisa e coerente (Figura 34).

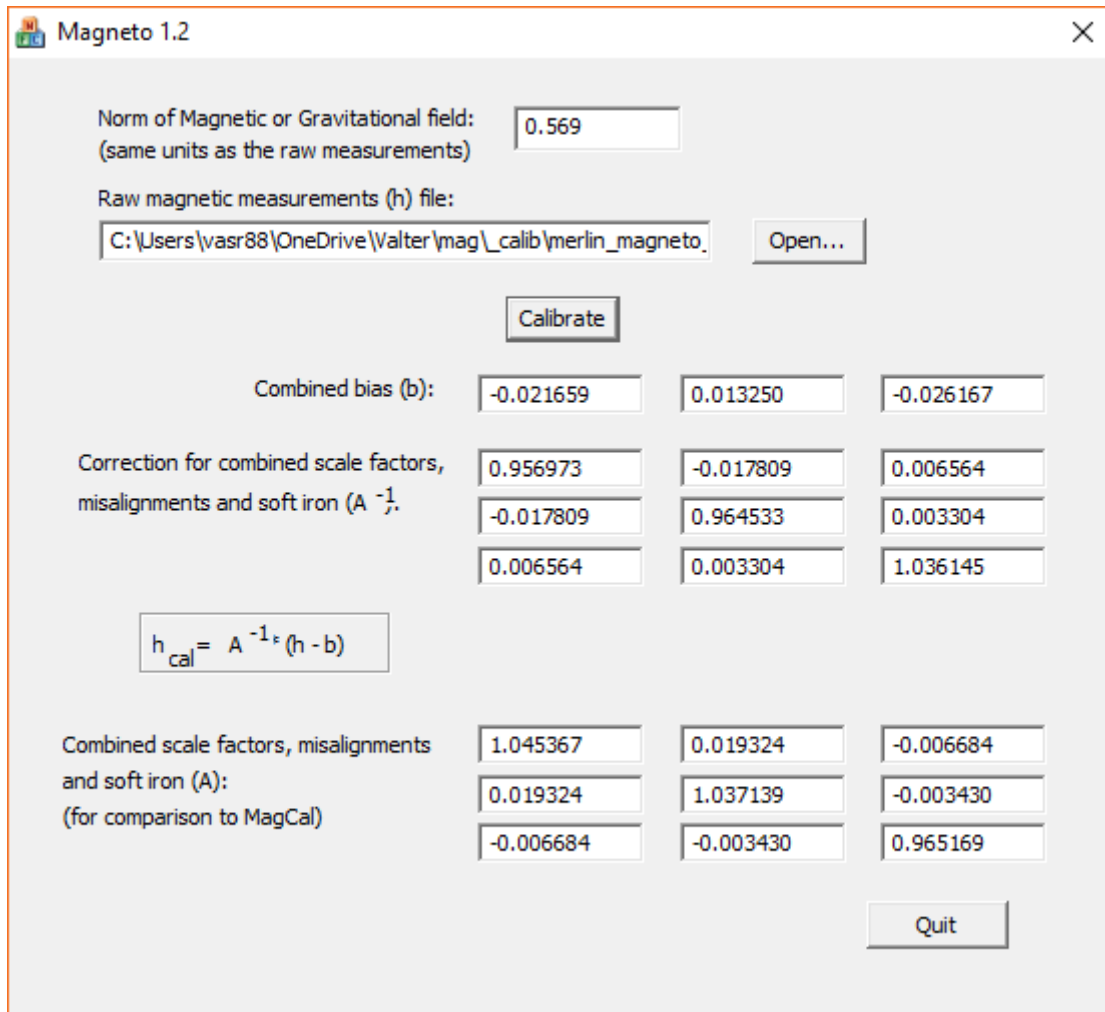


Figura 34 – Exemplo da execução do software Magneto.

Após a execução, como referido, os valores da matriz de compensação e o vector *bias* são exactamente iguais aos obtidos através da rotina desenvolvida em linguagem C:

$$\hat{A}^{-1} = \begin{bmatrix} 0.956973 & -0.017809 & 0.006564 \\ -0.017809 & 0.964533 & 0.003304 \\ 0.006564 & 0.003304 & 1.036145 \end{bmatrix}$$

$$\hat{b} = [-0.021659 \quad 0.013250 \quad -0.026167]$$

- **Calibração no MagMaster**

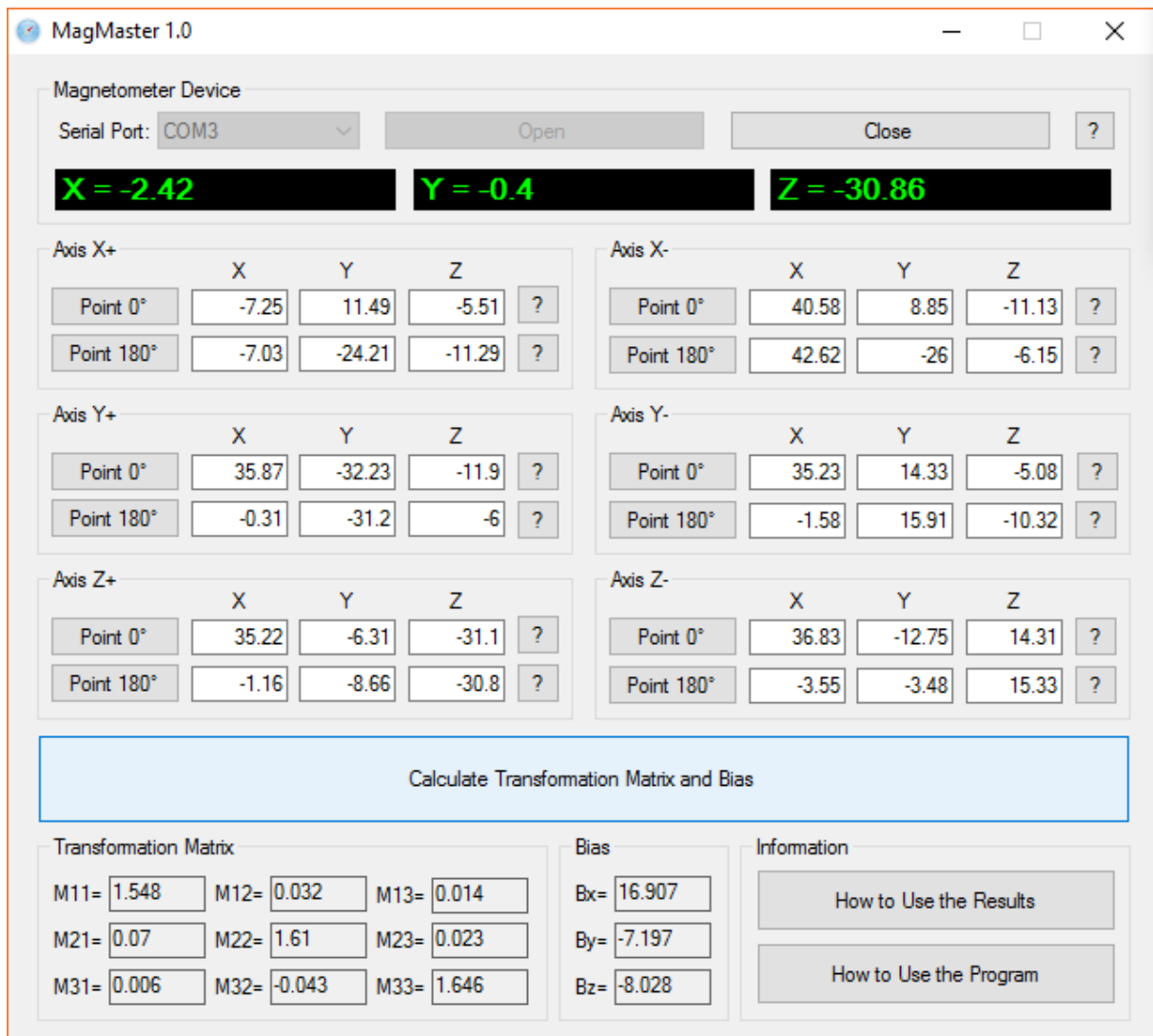


Figura 35 - Exemplo de utilização do MagMaster para calibrar o magnetómetro.

Tal como o Magneto o MagMaster é um *software open source* específico para efectuar a calibração de magnetómetros que não vêm calibrados de fábrica, este software é desenvolvido em C# e de fácil utilização mas neste caso o processo de obtenção da matriz de compensação e vector *bais* é bastante diferente quando comparado com os anteriores [36]. Enquanto nos métodos anteriores se obtêm valores arbitrários neste método obtêm-se valores específicos, que representam os valores máximos e mínimos para cada eixo do magnetómetro, dependendo da sua posição em relação ao Norte. Assim sendo é necessário encontrar a direcção do Norte, com a ajuda de uma bússola ou de um mapa por exemplo, a esta direcção corresponde o ângulo 0°, à direcção oposta a esta, ou seja, ao Sul corresponde o ângulo 180°.

Depois de se ter estas referências basta registrar o valor de cada componente do campo magnético para cada posição especificada pelo *software* MagMaster. Basta então obter as doze posições que correspondem a todas as posições máximas e mínimas possíveis dos eixos XYZ medidas pelo sensor (Figura 35).

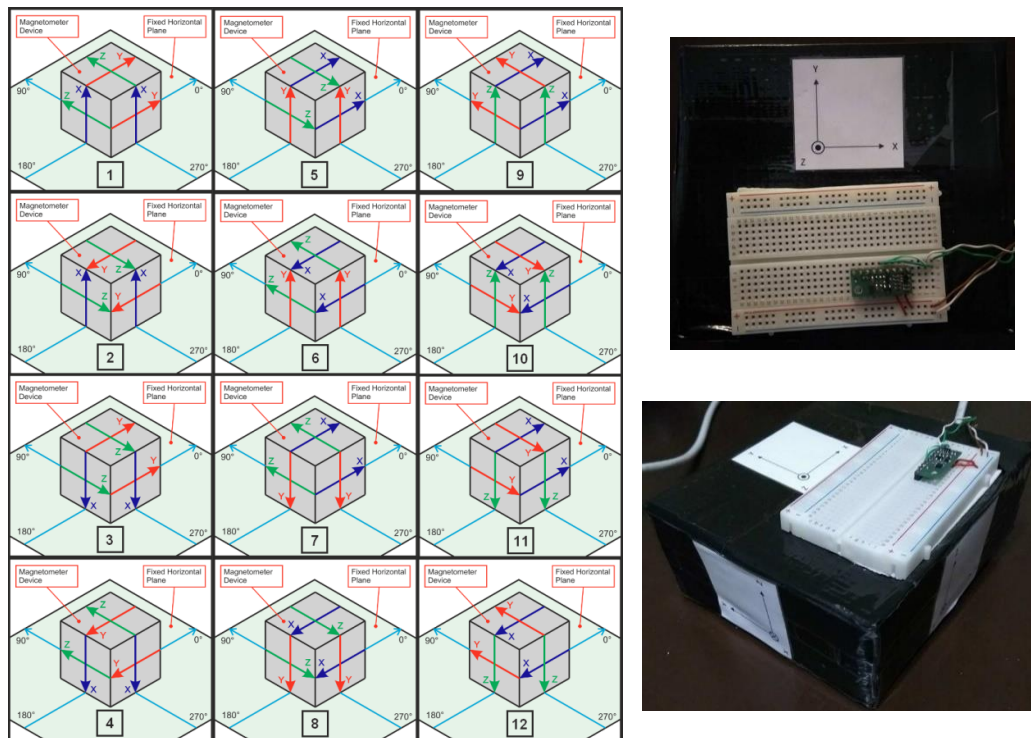


Figura 36 - Sistema de calibração com esquematização das várias posições necessárias (Extraído de software MagMaster).

Para que o MagMaster possa ter acesso aos valores medidos basta imprimir os valores para a porta série do Arduino com os valores de campo magnético escritos da forma X, Y, Z e conectá-lo a essa mesma porta, a partir daí o *software* inicia a leitura de valores. Para cada posição é necessário dispor o sensor de uma forma específica, para saber qual basta clicar no símbolo ? para se obter a posição correspondente ao eixo e ao ângulo (Figura 36). Depois de se obter todos os valores para todos os eixos e para todos os ângulos basta clicar em “*Calculate Tranformation Matrix and Bias*” e de imediato o MagMaster procede ao cálculo dos mesmos. Sendo que, a matriz de compensação e o vector *bais* obtidos são:

$$\hat{A}^{-1} = \begin{bmatrix} 1.548 & 0.032 & 0.0014 \\ 0.07 & 1.61 & 0.023 \\ 0.006 & -0.043 & 1.646 \end{bmatrix}$$

$$\hat{b} = [16.907 \quad -7.197 \quad -8.026]$$

Tal como para os métodos anteriores, estes valores são posteriormente colocados num ficheiro que pode ser lido pelo sistema ou directamente programados no método de calibração do código desenvolvido. Desta forma obtêm-se os valores calibrados do magnetómetro, sendo esta uma característica particular de cada local onde se está a realizar a caracterização magnética. Neste caso concreto, é um exemplo de uma calibração num dos locais de medida que são representados adiante, neste caso o método escolhido foi a calibração através do MagMaster mas qualquer um dos anteriores métodos podia ter sido utilizado.

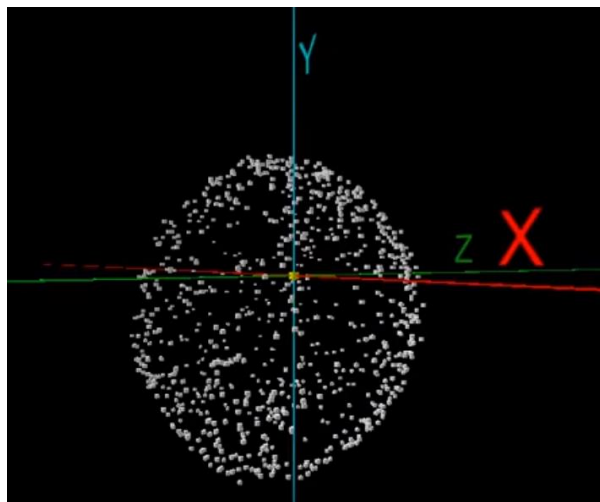


Figura 37 - Valores do magnetómetro antes da calibração (Imagem MagViewer).

Para ajudar na visualização dos resultados obtidos através dos *software's* Magneto e MagMaster utiliza-se um outro *software* igualmente *open source*, o MagViewer, que faz uma representação gráfica em tempo real dos valores obtidos através do sistema, ou seja, através da porta série do Arduino é possível injectar esses valores neste *software* que posteriormente os representa. Antes de se aplicar qualquer algoritmo de calibração os valores medidos sofrem precisamente os efeitos caracterizados anteriormente, provocados por erros de instrumentação, pelas distorções de magnetismo permanente (*hard iron*) e as pelas distorções de magnetismo induzido (*soft iron*). É observável que tal como esperado se forma um

elipsoide deformado, não centrado na origem do referencial cartesiano e encontra-se ligeiramente deslocado no sentido negativo dos três eixos X, Y e Z (Figura 37).

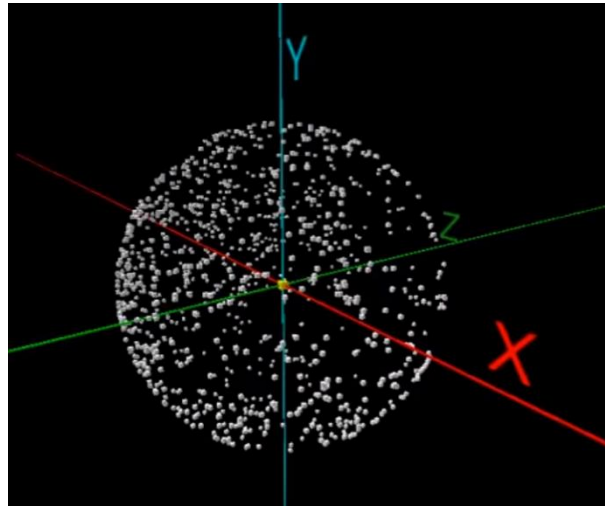


Figura 38 - Valores do magnetómetro depois da calibração (Imagem MagViewer).

Após a calibração todas as distorções são eliminadas, através da aplicação da matriz de compensação e do vector *bias*, o elipsoide transforma-se numa esfera perfeitamente bem definida e centrada na origem. Desta forma garante-se que o sistema está calibrado e ajustado às problemáticas do meio onde se estão a realizar as medidas, obtendo desta forma valores correctos do campo magnético envolvente. Esta é uma das questões fundamentais deste projecto, ou seja, é sempre necessário ter uma boa calibração do sistema, independentemente do método utilizado, neste caso é possível escolher qualquer um dos que foram apresentados anteriormente, isso garante versatilidade e ao mesmo tempo garante fiabilidade dos valores de campo magnético obtidos (Figura 38).

5. Testes e Validação do Sistema

5.1. Em Laboratório

- **Bobines de Helmholtz**

As bobinas de Helmholtz, formam um dispositivo que consiste em duas bobinas circulares com raio R igual, que são colocadas paralelamente uma à outra, e colocadas em oposição entre si a uma distância igual ao seu raio. Ao circular-se uma corrente elétrica nessas bobinas, é criado um campo magnético em seu redor, que tem a característica de ser uniforme no centro do dispositivo num volume menor do que as próprias bobinas (Figura 39) [43].

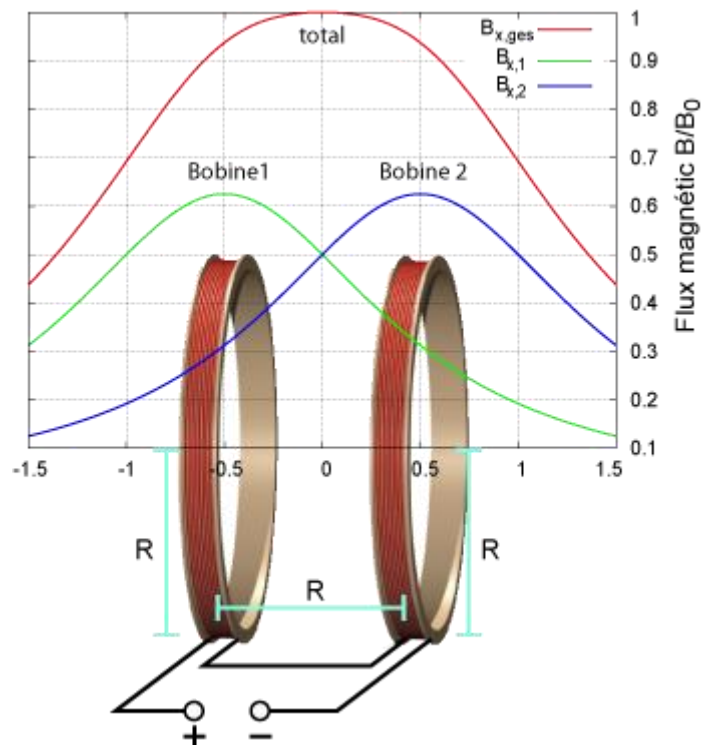


Figura 39 - Bobines de Helmholtz. [43]

$$B = \mu_0 \frac{8IN}{\sqrt{125}R}$$

Equação 15 – Equação que descreve o comportamento do campo magnético entre as bobines de Helmholtz. [43]

$$I - \text{Corrente}$$

$$\mu_0 - \text{Permitividade do vazio } (4\pi \times 10^{-7} \text{ H/m})$$

$$N - \text{N}^\circ \text{ de Espiras (124)}$$

$$R - \text{Raio das bobines e distância entre si (14,9c m)}$$

Através das bobines de Helmholtz é possível criar um campo magnético estável, é possível variar o mesmo ao longo do tempo e medi-lo utilizando o sistema desenvolvido. Isto é, sabendo que o campo magnético varia com a corrente basta aplicar um sinal triangular às bobines em que a corrente varia entre -55mA e +55mA, a tensão varia entre -1,1V e +1,1V o que faz com que o campo magnético gerando entre as bobines varie também entre aproximadamente -1 Gauss e +1 Gauss. Sendo que a frequência da onda é de 0,01 Hz o que perfaz um período de 100 segundos.

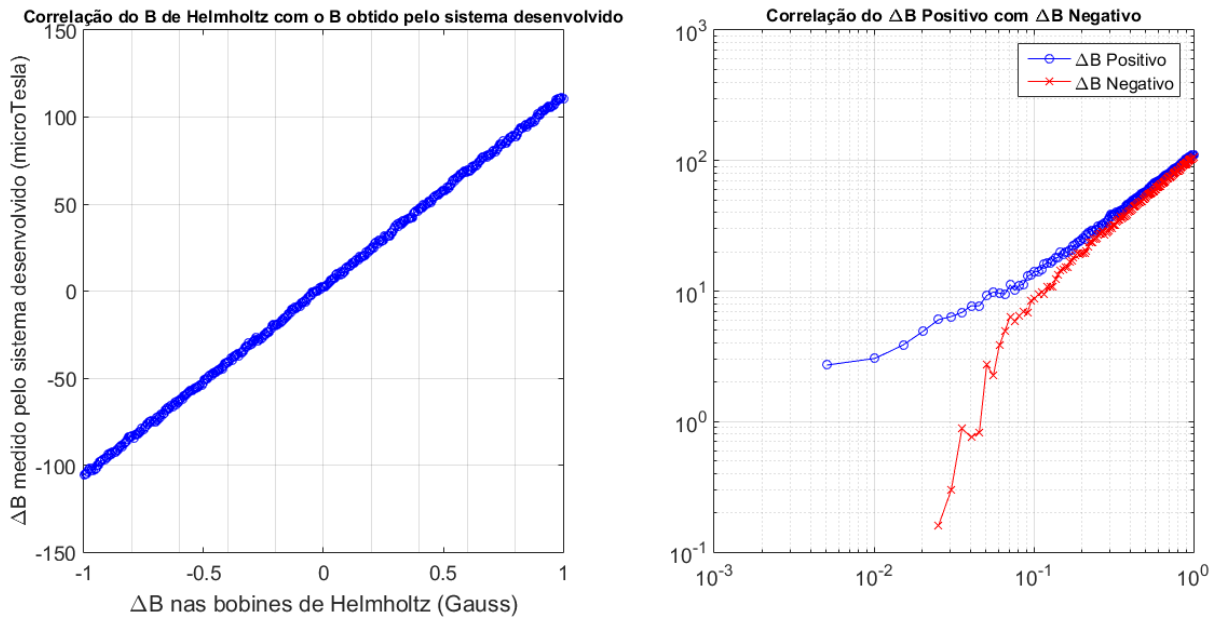


Figura 40 - Correlação entre o sinal gerado e o sinal medido e entre as partes positiva e negativa deste.

Após a correlação dos valores gerados pelas bobines de Helmholtz com os valores medidos pelo sistema desenvolvido verifica-se que existem correspondência dos valores, ou seja, 1 Gauss são 100 microTesla e -1 Gauss são -100 microTesla, o que possibilita concluir que a calibração do sistema desenvolvido está bem feita (Figura 40). Para além disso é feita também a correlação entre a parte positiva e a parte negativa da variação do campo magnético e é evidenciado um pequeno afastamento dos valores para quando o campo se aproxima do valor 0, este efeito verifica-se porque a precisão do sensor para valores muito pequenos é reduzida, ou seja, muito próximo de zero, o sensor não consegue medir os mesmos (Figura 40).

Este não é um problema grave porque as anomalias encontradas são sempre muito superiores as estes valores e nessa gama o sensor é muito preciso.

- **Medidas de longa duração**

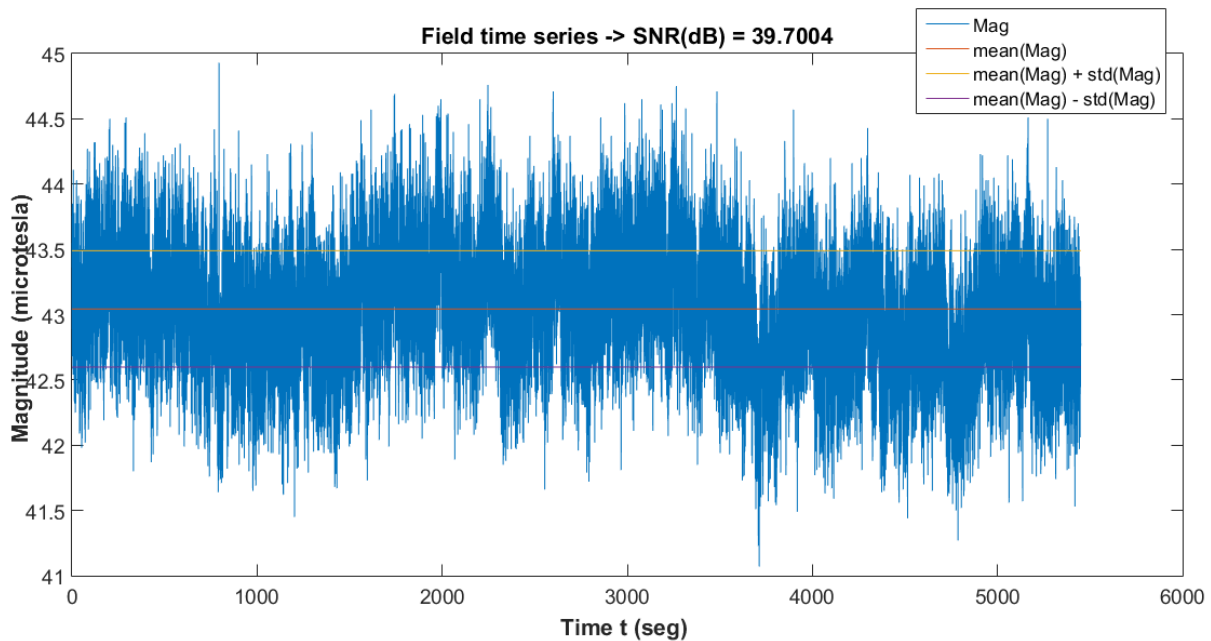


Figura 41 - Medida de longa duração para aferir a estabilidade do sinal.

Apesar de o teste de longa duração não ter sido feito em condições ideais, visto que foi feito em laboratório e existirem várias condicionantes que afectam o sinal, tais como a estrutura do edifício, o mobiliário existente e a própria rede elétrica do laboratório verificou-se que o sinal é bastante estável ao longo do tempo, verificando-se apenas uma flutuação mínima dos valores medidos. O valor médio obtido é de aproximadamente 43,1 microTesla, muito próximo do valor regional do campo magnético terrestre, cerca de 43,8 microTesla, e um desvio padrão de aproximadamente 0,44 microTesla, com uma relação sinal-ruído (SNR) de aproximadamente 39,7 dB (Figura 41).

5.2. No Campo

- **Correlação entre o Sistema implementado e um Magnetómetro de Césio**



Figura 42 - Magnetómetro de Césio Geometrics utilizado nas medidas. [45]

Como um dos objectivos do projecto é que o sistema seja fiável nos valores medidos, há a necessidade fazer uma correlação entre os valores que este obtém com os valores obtidos por um sistema no qual se tenha confiança na fiabilidade. É então aqui que surge o magnetómetro de Césio da Geometrics [45], um magnetómetro altamente preciso e estável, com baixo ruído, alta sensibilidade ($0.008\text{nT} / \sqrt{\text{Hz}}$) RMS, regista a magnitude e a localização GPS a uma taxa de 5 amostras por segundo e rejeita o ruído (50/60 Hz) da rede eléctrica, este magnetómetro viabiliza então a correlação entre sistemas, e vem ajudar a ter certeza de que o sistema desenvolvido funciona bem e mede com precisão o campo magnético envolvente.

Para se efectuar a correlação entre os dois sistemas é necessário realizar medidas em linha recta e no mesmo local para ambos. No exemplo seguinte percorreu-se uma distância em linha recta de aproximadamente 250 metros, um percurso bem definido de um ponto A para um ponto B. Posteriormente para se conseguir uma percepção melhor do percurso realizado e para se obter uma boa correlação dos sistemas são necessários alguns ajustes dos valores obtidos e estes são especificados a seguir:

Tabela 7 – Passos necessários para se efectuar a análise dos valores obtidos.

1º	É necessário converter as coordenadas geográficas (GPS) obtidas para coordenadas UTM ^(*) , para ambos os sistemas. Esta conversão é necessária porque no equador um minuto de arco em longitude tem a dimensão aproximada de 1852m (1 milha náutica) e à latitude de 38º, um minuto de arco em longitude tem a dimensão de aproximadamente 1460m. (Figura 43 - Gráfico da Esquerda).
2º	Faz-se a translação dos valores obtidos para se obter a distância percorrida em metros (Figura 43 - Gráfico da Direita).
3º	Faz-se a rotação também, para se obter um gráfico onde o ponto A fica posicionado mais à esquerda, e representa a distância mínima em metros e o ponto B fica posicionado mais à direita, e representa a distância máxima em metros (Figura 44).
4º	Devido à limitação de precisão dos dispositivos de GPS, com um erro variável dependendo da visibilidade para os satélites, faz-se a remoção dos valores absurdos, eliminam-se os valores superiores a 5 metros quer para a esquerda quer para a direita dos percursos realizados (Figura 44).
5º	Finalmente, basta fazer <i>plot</i> dos valores medidos pelo magnetómetro de césio em função dos medidos com o sistema desenvolvido (Figura 45).

(*) UTM é um sistema de coordenadas baseado no plano cartesiano (eixo x,y) e usa o metro (m) como unidade para medir distâncias e determinar a posição de um objeto. Ao contrário das coordenadas geográficas (GPS), o sistema UTM, não acompanha a curvatura da Terra e por isso os seus pares de coordenadas são comumente conhecidas como coordenadas planas. [44]

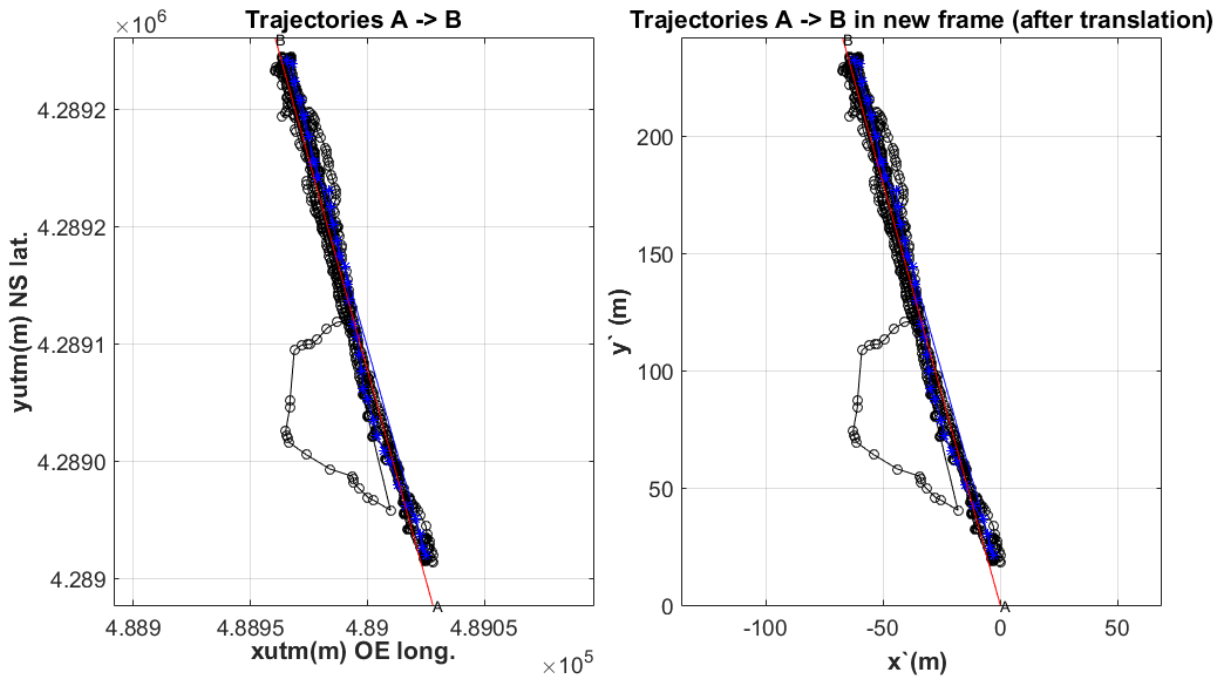


Figura 43 - Conversão para coordenadas UTM e translação para obter distância em metros.

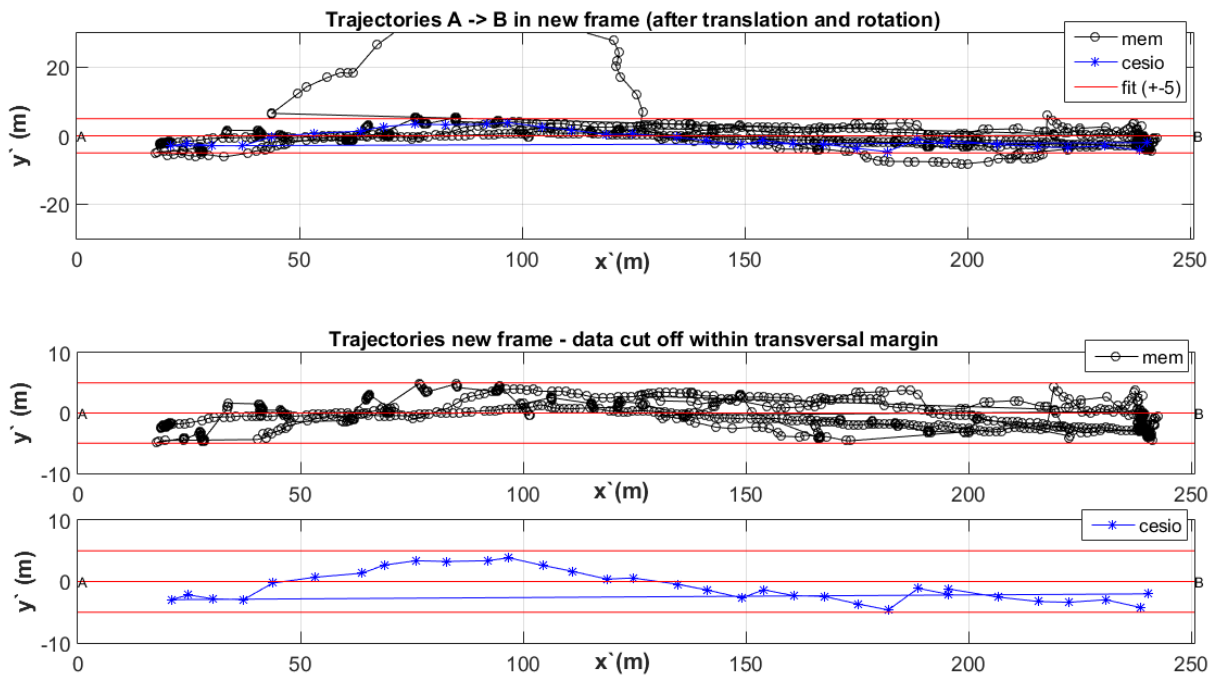


Figura 44 - Rotação dos valores obtidos e remoção dos valores absurdos.

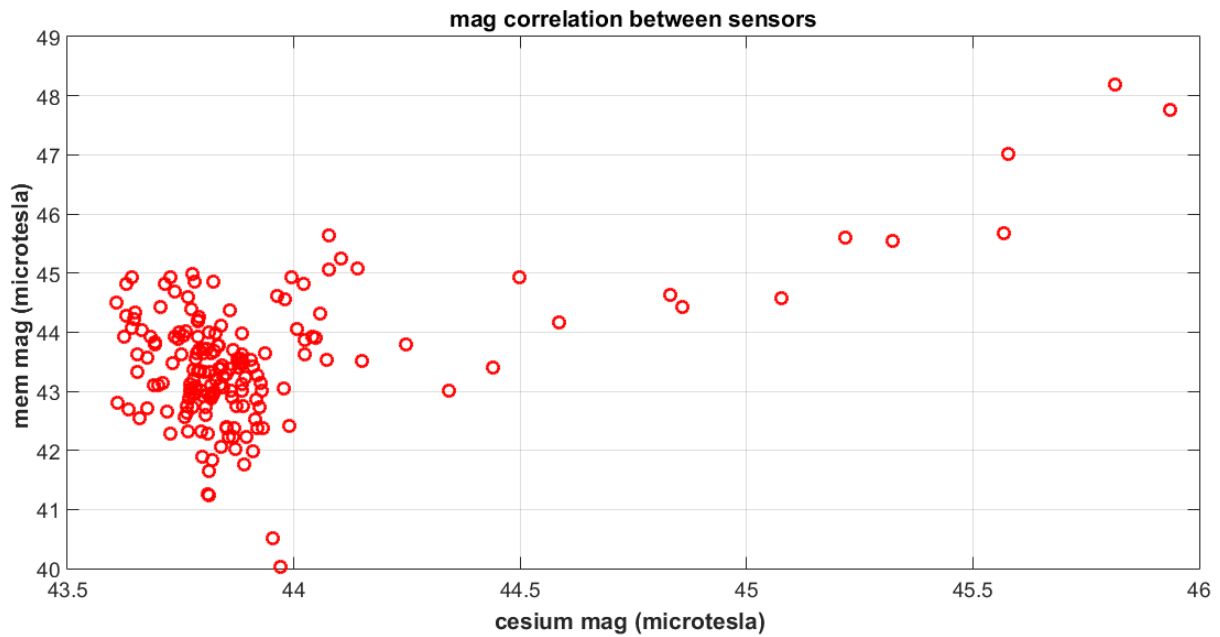


Figura 45 - Correlação dos valores obtidos pelo sistema desenvolvido e os obtidos pelo magnetómetro de céσιο.

Após a aplicação dos passos referidos anteriormente (Tabela 7) obtêm-se uma correlação boa dos valores obtidos, a mesma é ligeiramente limitada devido ao facto de a precisão do sistema desenvolvido ser inferior à precisão do magnetómetro de céσιο mas estes resultados comprovam que a calibração do sistema está bem feita e que estes representam com exatidão o campo magnético do local (Figura 45).

6. Mapeamentos



Figura 46 – Esquemática de um percurso realizado para obtenção das medidas de campo magnético.

Para se realizar o estudo e o mapeamento manual do campo magnético numa determinada área é necessário definir o percurso a realizar nesta e a quantidade de pontos que se pretende obter, ou seja, se queremos fazer um mapeamento fino ou um mapeamento mais espaçado, isto é, com menos resolução. Sendo que esse percurso não tem necessariamente de seguir um padrão, pode ser feito de forma aleatória ao longo da área a caracterizar. Apenas por uma questão de facilitar o trabalho e de se ter uma melhor noção do espaço já medido, utilizam-se grelhas, normalmente compostas por linhas rectas (Figura 46). Nos mapeamentos realizados, devido à inexistência de anomalias grandes, utiliza-se um carro para a provocar, para que seja mais fácil perceber o correcto funcionamento do sistema e também para ajudar no posterior tratamento dos valores medidos.

6.1. Método de Kriging

Para se efectuar a caracterização de uma determinado local é necessário adquirir um número razoável de pontos numa determinada área do mesmo. Para isso com o magnetómetro de césio são registados cerca de 50 pontos na área definida, de seguida faz-se a obtenção de valores da mesma área mas utilizando o sistema desenvolvido. Tendo os valores adquiridos é posteriormente realizado o tratamento destes, seguindo os passos descritos a seguir:

Tabela 8 - Passos necessários para se efectuar a análise com o método de Kriging dos valores obtidos.

1º	Faz-se a remoção dos valores absurdos, eliminam-se os valores superiores a superiores a 40º e inferiores a 36º no caso da latitude e os valores superiores a 11º e inferiores a 7º no caso da longitude.
2º	É necessário converter as coordenadas geográficas (GPS) obtidas para coordenadas UTM, para ambos os sistemas (Figura 47).
3º	Depois pode ser aplicado o método de Kriging que é uma interpolação baseada em métodos geoestatísticos e autocorrelação dos pontos medidos (Figura 48 e Figura 49).
4º	Efetuar a subtração do plano regional aos valores medidos, este plano representa o campo magnético médio para a área em questão (Figura 50).
5º	Finalmente, basta fazer <i>plot</i> num gráfico 3D dos valores obtidos, em que os eixos X e Y são a longitude e latitude respectivamente e o eixo Z é a variação do campo magnético (B) (Figura 50 e Figura 51).

O Kriging é um método de interpolação que consiste em métodos geoestatísticos, baseado em processos Gaussianos, ou seja, em modelos estatísticos que incluem autocorrelação entre os pontos medidos [46]. Assim sendo, as técnicas geoestatísticas não têm apenas a capacidade de produzir uma superfície de previsão, mas também fornecem uma medida da certeza ou exatidão das previsões. O método Kriging assume que a distância ou a direção entre os pontos obtidos reflete uma correlação espacial que pode ser usada para explicar a variação na superfície, ajusta-se uma função matemática a um número específico de pontos ou a todos os pontos dentro de um raio específico, para determinar o valor previsto para cada local.

```

77      % calculate the sample variogram
78 -    v = variogram([x y],z,'plotit',false,'maxdist',100);
79 -    [dum,dum,dum,vstruct] = variogramfit(v.distance,v.val,[],[],[],'model','stable');
80      % now use the sampled locations in a kriging
81 -    [Zhat,Zvar] = kriging(vstruct,x,y,z,xi,yi);

```

É um processo feito em várias etapas, inclui análise estatística exploratória dos dados, modelagem de variogramas, criação da superfície e exploração de uma superfície de variância. É relativamente simples de usar em Matlab visto que basta usar algumas funções já existentes, para a modelagem de variogramas (linha 78), para a criação da superfície (linha 79) e exploração da superfície de variância e

mapeamento dos valores medidos (linha 81). Este método é apropriado quando se sabe que há uma distância espacialmente relacionada ou um desvio direcional nos dados, como é o caso concreto da caracterização de uma determinada área terrestre. [46] Este método funciona muito bem para a falta de pontos numa determinada área, ou seja, ele consegue através dos processos referidos anteriormente gerar uma representação clara da área medida, por outro lado, por vezes, quando existem demasiados pontos medidos o método torna-se pesado e os resultado da correlação entre os vários pontos não é tão bom quanto o esperado.

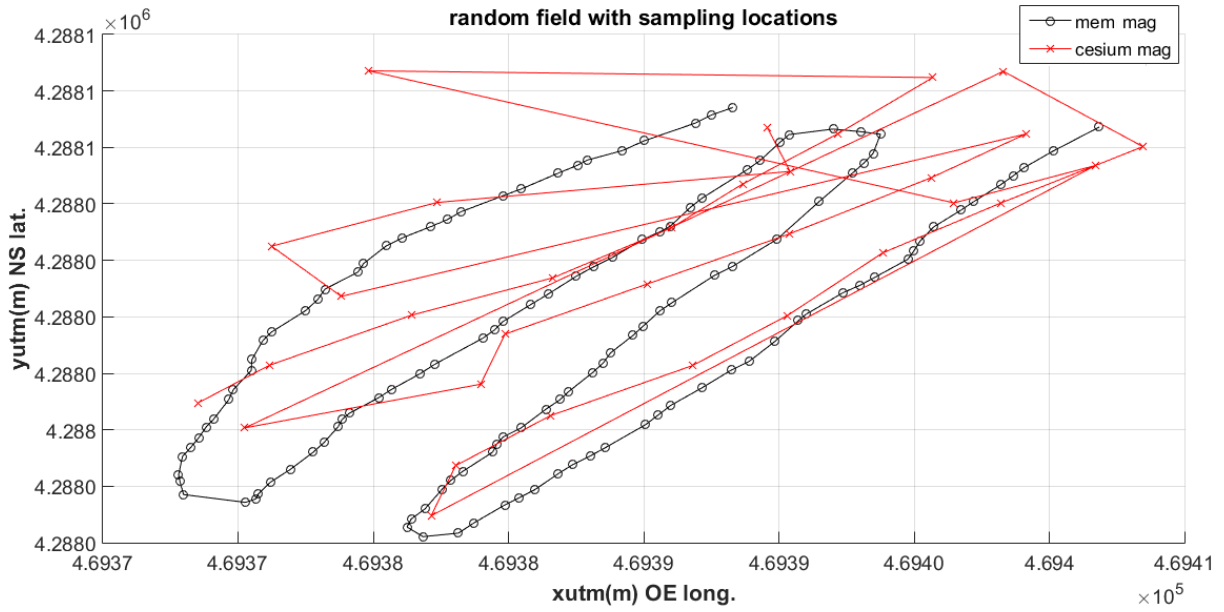


Figura 47 - Representação dos vários pontos medidos pelos dois sistemas.

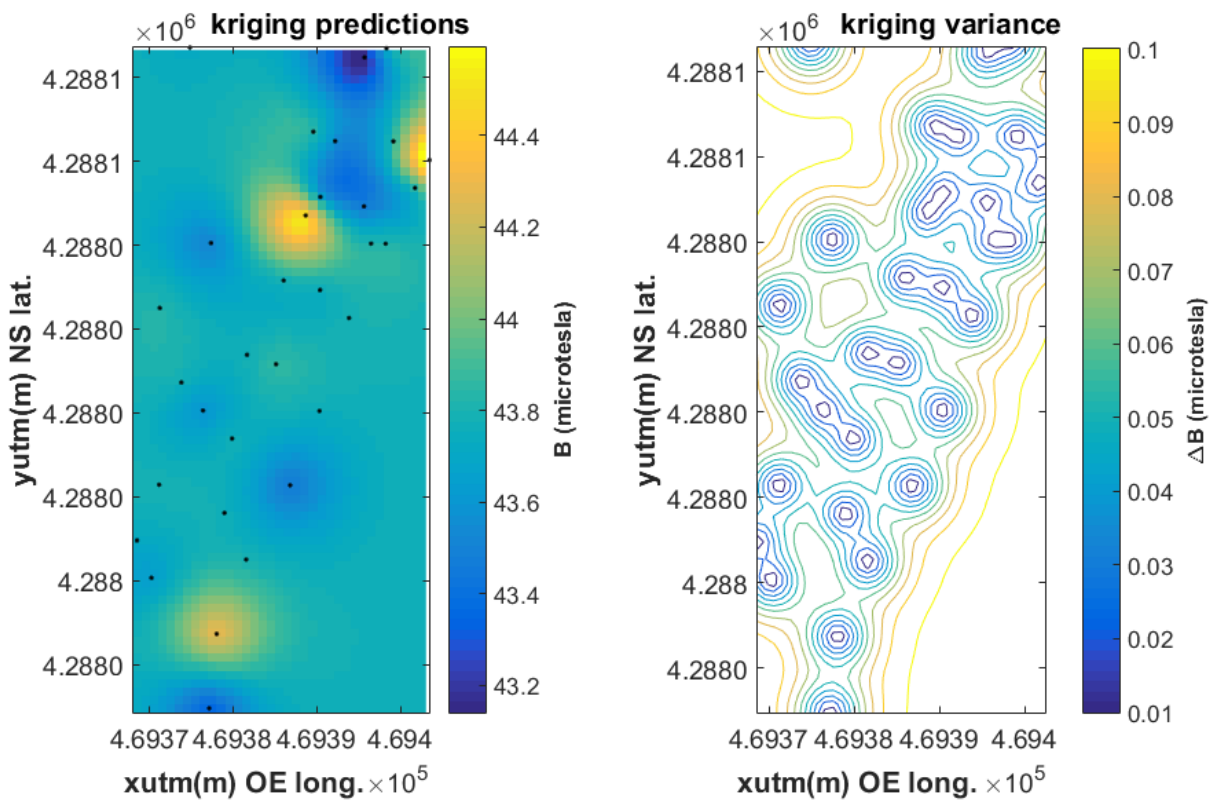


Figura 48 - Interpolação e obtenção da variância com o método de Kriging para os valores obtidos com o magnetômetro de césio.

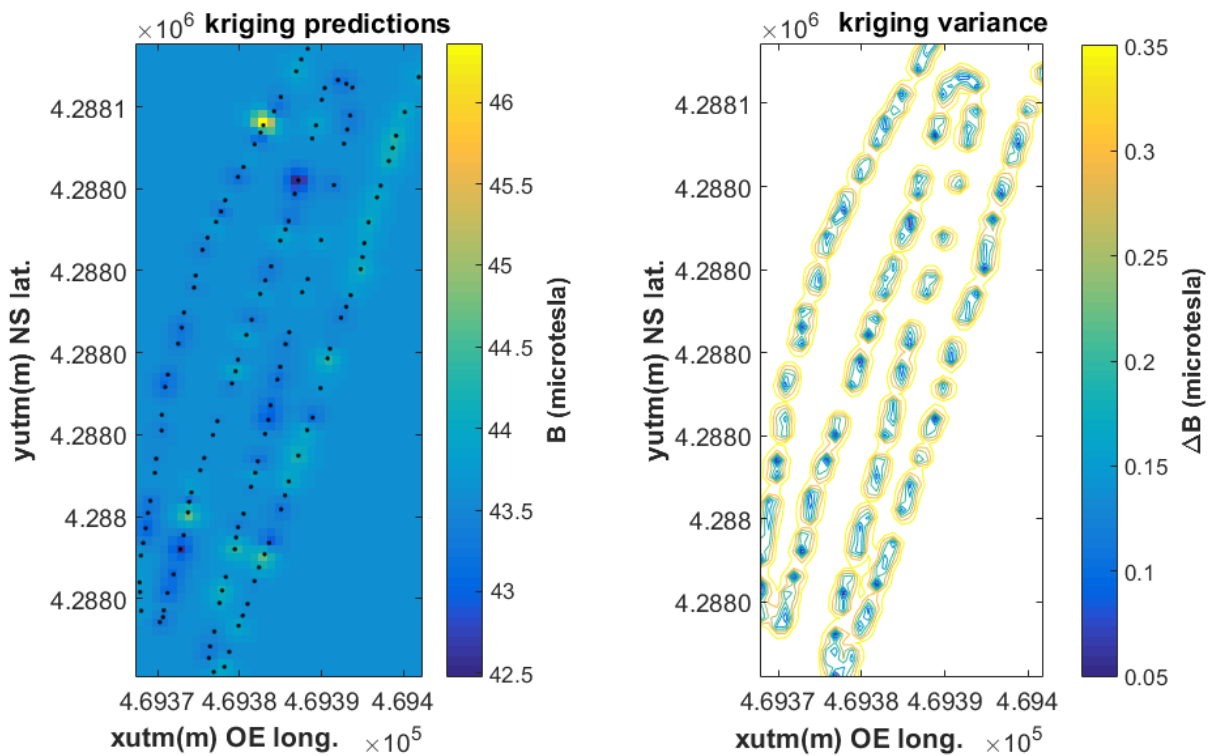


Figura 49 - Interpolação e obtenção da variância com o método de Kriging para os valores obtidos com o sistema desenvolvido.

Após a aplicação do algoritmo de tratamento dos dados obtém-se os resultados que caracterização a área escolhida, para além disso é possível perceber que os resultados obtidos pelos dois sistemas são bastante semelhantes (Figura 48 e Figura 49). Apenas se verifica alguma variação na escala visto que o processo de obtenção de dados com o magnetómetro de céσιο é muito moroso e é difícil obter tantos pontos quantos os que se conseguem obter com o sistema desenvolvido.

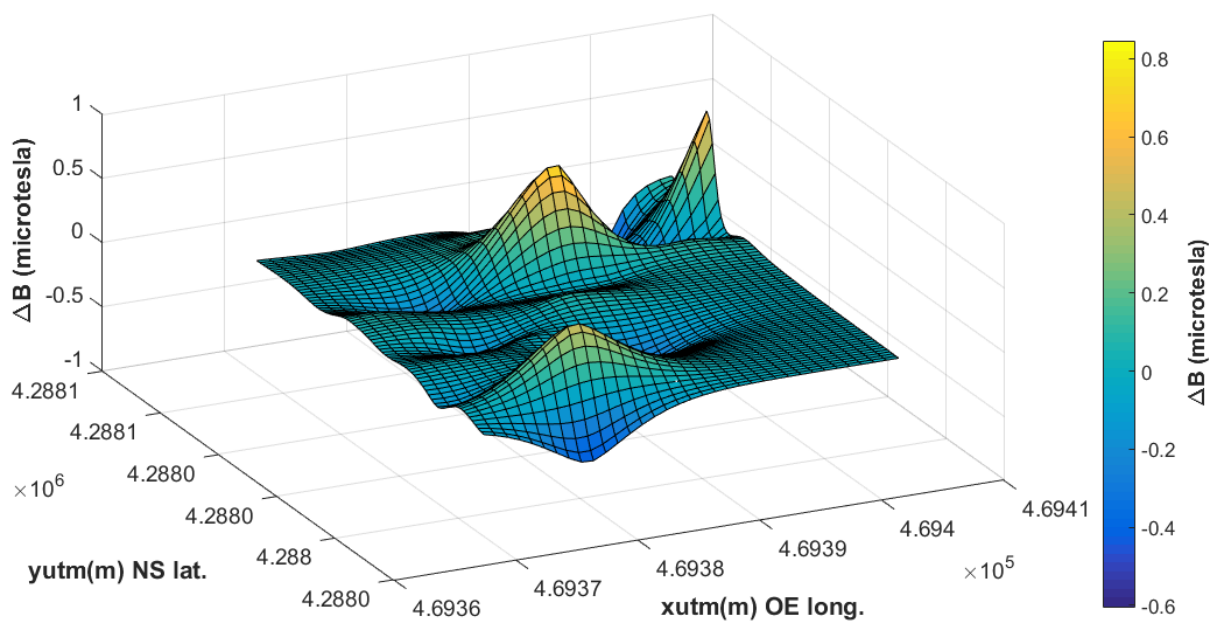


Figura 50 - Representação em gráfico 3D dos valores obtidos valores obtidos com o magnetómetro de céσιο, com a subtração do plano regional efectuada.

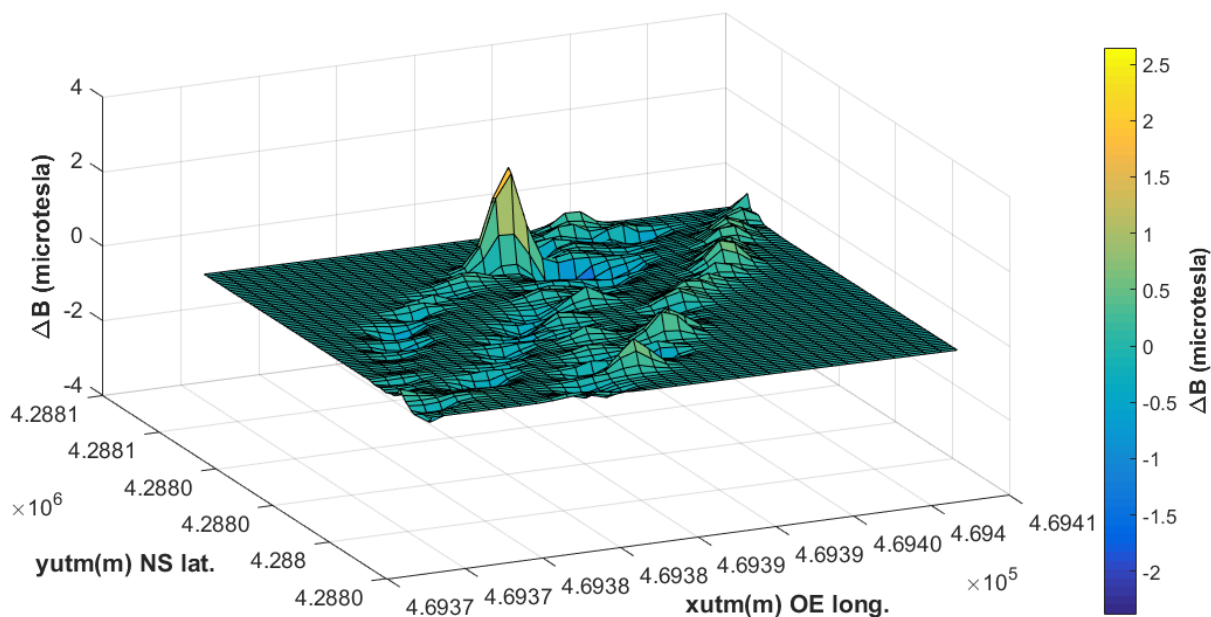


Figura 51- Representação em gráfico 3D dos valores obtidos com o sistema desenvolvido, com a subtração do plano regional efectuada.

Verifica-se então que existe claramente uma identificação da anomalia provocada pelo carro, tendo esta um valor residual, superior ao valor do campo magnético local, de aproximadamente 1,5 microTesla, onde o campo regional tem um valor médio de 43,8 microTesla e um alinhamento segundo Nordeste/Sudoeste (Figura 50 e Figura 51). Para além dessa anomalia elevada verifica-se a existência de várias anomalias residuais mais pequenas, provocadas pela acumulação de uma grande quantidade de resíduos no local, onde existiam metais ferromagnéticos, vários tipos de fibras, componentes de automóveis e compostos de características argilosas.

6.2. Mapeamentos Manuais

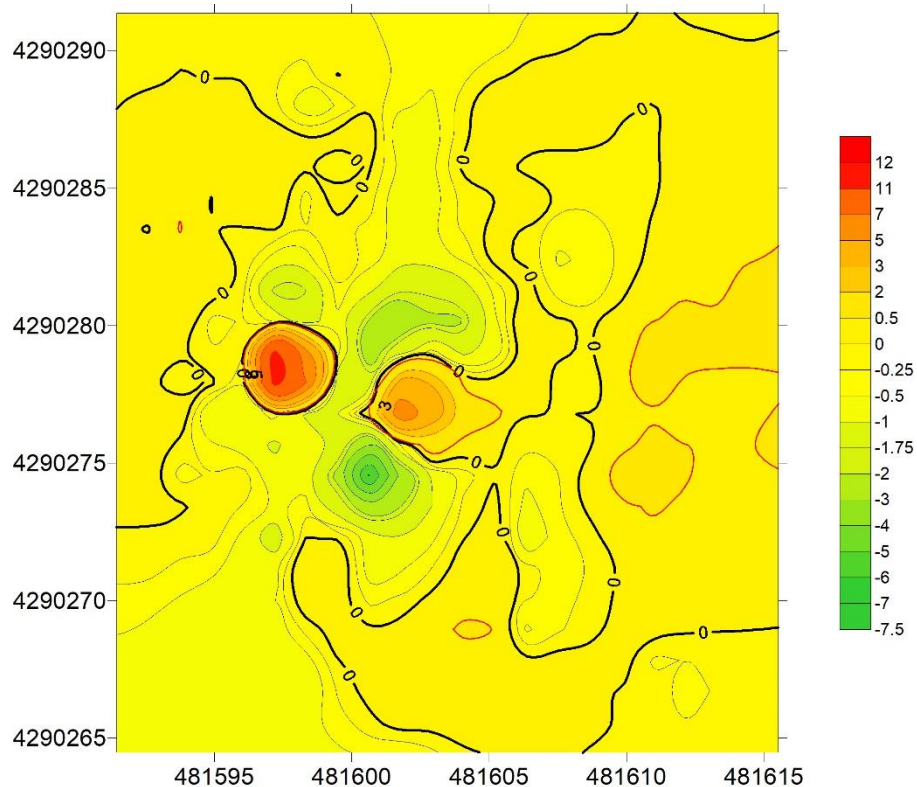


Figura 52 - Campo magnético residual da área estudada, com representação da anomalia provocada pelo carro.

Para se realizar os mapeamentos manuais nesta região percorre-se uma área de aproximadamente 500 metros quadrados onde mais uma vez um carro foi utilizado como anomalia permanente do local em questão, o mapeamento é feito de Norte para Sul e com uma distância entre varrimentos de aproximadamente um metro. Obteve-se assim uma caracterização detalhada da área analisada onde a anomalia está bem definida, verificando-se a vermelho a posição e alterações provocadas pelo carro, sendo que esta é superior ao campo magnético regional em aproximadamente 5 microTesla (Figura 52), isto porque o campo magnético regional tem um valor aproximado de 46,8 microTesla na zona estudada como se pode verificar na imagem a baixo (Figura 53).

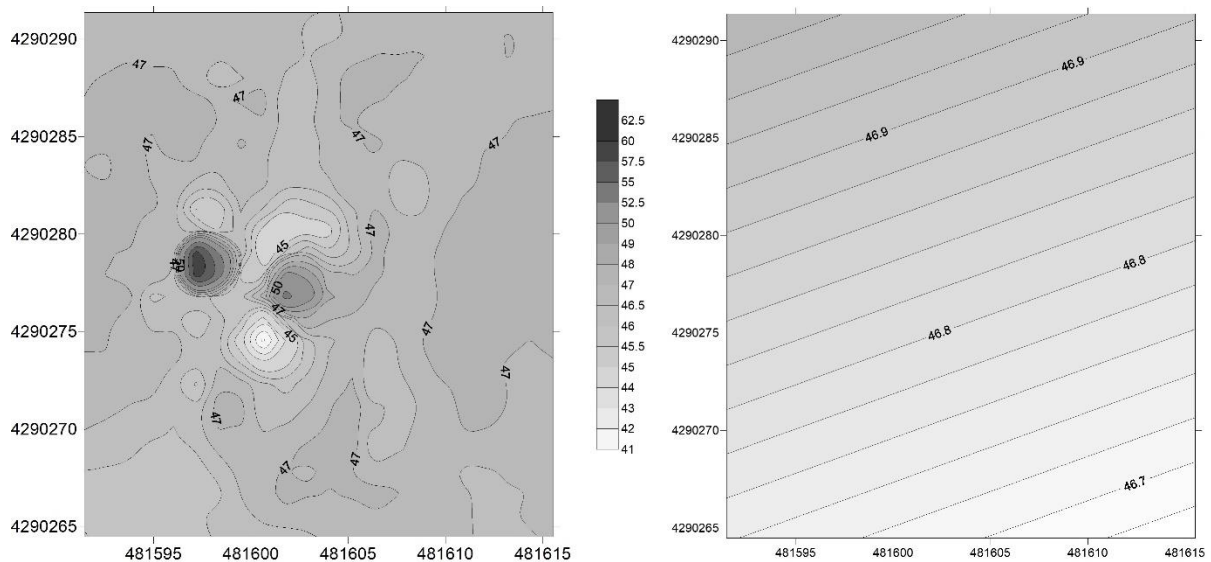


Figura 53 - Representação das linhas de iso dos valores do campo magnético e do campo regional.

Para se obterem estes resultados era possível recorrer-se ao Matlab tal como é mostrado nos mapeamentos manuais anteriores mas para se ter uma comparação entre a análise feita anteriormente e esta, recorre-se então ao *software* Origin, que é um *software* amplamente utilizado para tratamento de dados magnéticos e que dispõe de funcionalidades que facilitam a caracterização de áreas terrestre. No Origin o método seguido também é o de Kriging mas mais otimizado, onde é necessário, em primeiro lugar, converter as coordenadas geográficas (GPS) em coordenadas UTM Portugal, zona 29S, para ter uma representação X-Y isométrica da área. Depois é feita uma interpolação através do algoritmo de Kriging para a criação de uma grelha de 22 por 25 nodos que representam a área estudada. Em seguida são representadas as linhas de iso dos valores de campo magnético e determinado o valor do campo regional através do cálculo de uma superfície de 1º grau por interpolação com todos os pontos do campo. Finalmente, para ser possível obter o gradiente representado e observar de forma clara a anomalia provocada pelo carro, realiza-se a subtração do campo regional aos valores do campo magnético medidos, obtendo assim o campo residual da área.

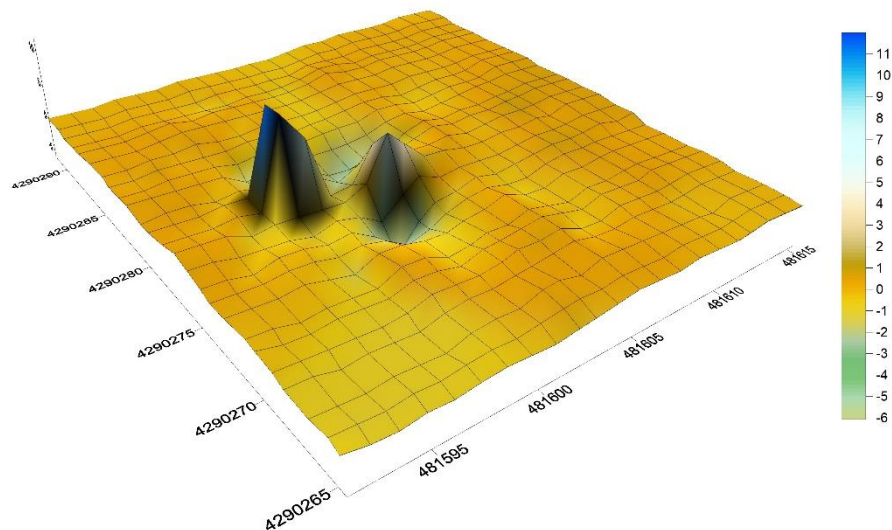


Figura 54 - Representação 3D da área estudada e da anomalia provocada pelo carro.

Através do gráfico 3D, onde X e Y são as coordenadas UTM e Z o valor residual do campo magnético B, é feita uma representação clara da anomalia provocada pelo carro, que tem um valor médio superior ao campo magnético regional de aproximadamente 5 microTesla. Sendo que esta chega mesmo a atingir um diferencial de aproximadamente 12 microTesla no seu maior pico, é numa área muito pequena mas é uma alteração bastante significativa (Figura 54).

7. Mapeamentos com Drone

7.1. Drone



Figura 55 - Drone quadrirotor (Magdrone).

O drone utilizado pode realizar a caracterização de perfis cartográficos através de um sensor LIDAR e aeromagnéticos através do sistema desenvolvido, que é integrado neste. A utilização conjunta do LIDAR, com determinação de posição através de GPS-RTK permite a realização de voos com controlo automático da distância ao solo. Quanto às suas características é um quadrirotor convencional, com estrutura em alumínio e fibra de carbono, com trem de aterragem retráctil, com 4 motores DJI E800 e hélices de duas pás de 13". Cada motor tem uma carga máxima de aproximadamente 2,1 kg (Figura 55). É alimentado com uma bateria LiPo 6S de 22,2V e 7Ah. A intensidade de corrente em voo estável é de 10A a 16A em média, em casos pontuais como correcções ou movimentos bruscos (acelerações) a corrente chega a atingir picos até 20A. A massa deste sem bateria é aproximadamente de 1600g, sendo

que a massa da bateria é de cerca de 870g. De forma a limitar a interferência electromagnética os cabos de potência passam dentro dos tubos de carbono de cada braço do drone.

A controladora é uma Pixhawk 2.4, a telemetria é feita através de Modulo 3D Robotics a 433 MHz e 100 mW, o receptor de rádio é um FrSky X8R 8/16CH Full Duplex Telemetry, enquanto o sistema de lidar é o LIDAR LightWare SF11 (0 a 40 m, precisão 1 cm) integrado no controlador Pixhawk, com aquisição contínua de distâncias e o sistema de localização fica ao cargo do GPS-RTK (Real Time Kinematics + Ublox M8P) integrado no controlador Pixhawk com precisão centimétrica também [50].

7.2. Simulação com Ardupilot e Mission Planner

- Ardupilot

O Ardupilot é um *software* de piloto automático avançado e completo, que é desenvolvido à aproximadamente cinco anos por uma equipa de diversos engenheiros e cientistas, é capaz de controlar vários tipos de veículos como aviões convencionais, multi-rotores e helicópteros, barcos e até submarinos. Possui também ferramentas de registo de dados, de análise e de simulação. É um *software* de piloto automático muito testado e utilizado de forma global, é *open source* o que significa que evoluiu diariamente e muito rápido. [47]

Através do Ardupilot é possível simular o comportamento exacto de qualquer sistema de piloto automático e até simular condições atmosféricas para que os dados obtidos sejam exactamente os mesmos aquando de um evento real.

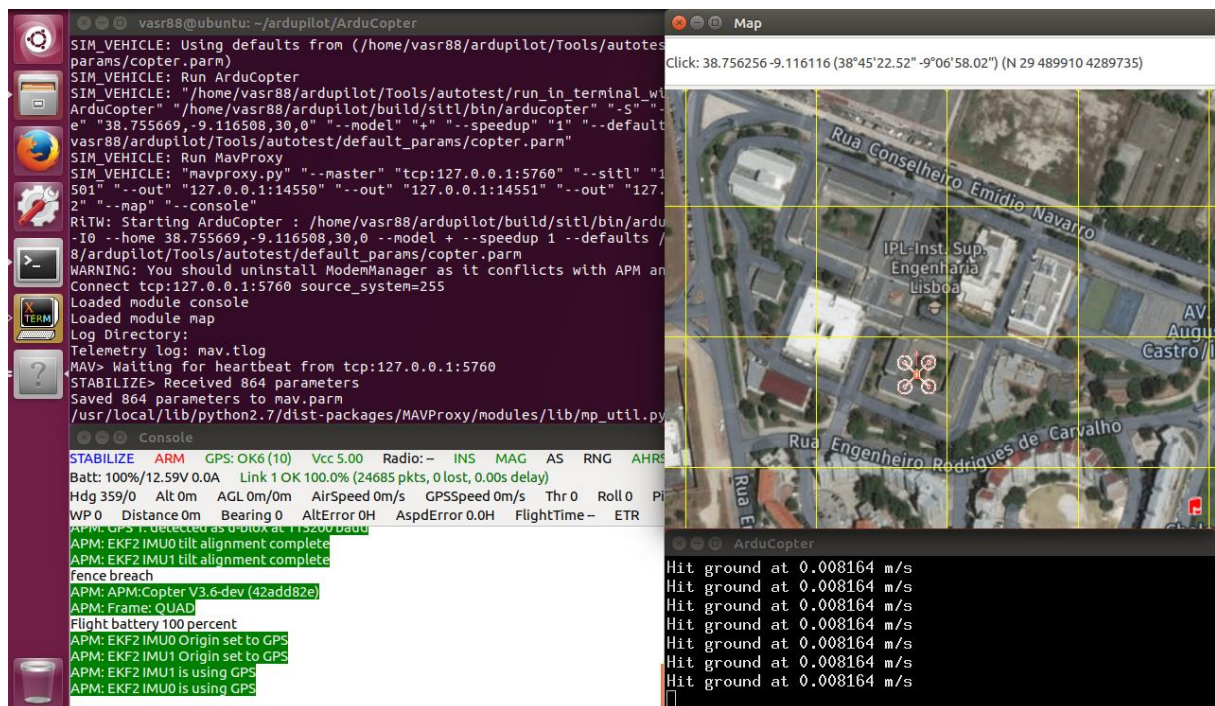


Figura 56 - Software Ardupilot a executar na máquina virtual Linux.

Para se poder utilizar o Ardupilot é necessário criar uma máquina virtual Linux na qual este é instalado e através da qual se lança o Arducopter, por outras palavras, através da qual é possível criar um “drone virtual” (Figura 56). Este “drone virtual” tem exactamente as mesmas características que o drone físico, ou seja, é simulada uma

conexão à controladora de voo e serão posteriormente enviados estímulos à mesma que levarão a que o “drone virtual” se comporte de uma determinada maneira.

Este *software* permite ao utilizador perceber se as missões que pretende realizar são ou não possíveis mediante determinadas situações, garantindo desta forma um maior sucesso das mesmas e evitando problemas na sua execução. Principalmente quando estão a ser usados sistemas de elevada sensibilidade e de elevado custo também. [47]

- **Mission Planner**

O Mission Planner é um *software* de *ground station* para o projeto de sistemas com piloto automático, é também ele *open source* e faz parte do Ardupilot. É uma *ground control station* para aviões, helicópteros, drones e carrinhos. O Mission Planner pode ser usado como um utilitário de configuração ou como um suplemento de controlo dinâmico para um veículo autónomo. Através deste é possível carregar o *firmware* (*software*) no controladora (APM, PX4, entre outras) que controla o dispositivo em questão, permite configurar e ajustar o dispositivo dependendo das necessidades, permite planear, salvar e carregue missões automáticas na controladora e carregar e analisar os *logs* de missão criados durante o percurso efectuado. [48]

Com um *hardware* de telemetria é possível monitorizar o status do dispositivo enquanto está em operação, registar dados de telemetria que contenham mais informações que os *logs* da controladora e visualizar e analisar esses *logs* posteriormente.

É um *software* versátil e completo, permite desenhar as missões a realizar no campo, simulá-las para garantir o sucesso das mesmas e para além disso permite analisar os dados quer de *logs* da controladora de voo quer da telemetria do drone, tal como referido anteriormente.

Depois de lançada a simulação do drone através do Ardupilot na máquina virtual basta aceder via TCP ao IP e porto de comunicação estabelecido com esta e de imediato se ganha o controlo do drone através do Mission Planner, a partir daí basta criar ou importar missões, caracterizadas por um ponto base de onde o drone levanta, vários *waypoints* por onde irá passar e um regresso à base onde ele irá

aterrar. Depois de realizado este processo é necessário verificar se a missão foi ou não realizada com sucesso e se sim, só depois é então possível carregar a mesma no drone real e executá-la.

Este *software* é fundamental na realização de voos de forma autónoma e controlada pois possibilita ter imensa confiança e segurança no trabalho e missões desenvolvidas bem como permite que o mesmo seja muito mais automatizado e eficiente. [48]

- **Planeamento de missões**



Figura 57 - Exemplo de um mapeamento de terreno Nordeste para Sudoeste -> Sudoeste para Nordeste.

Através da funcionalidade Plano Voo do Mission Planner é possível planear várias missões para realização das medições de campo magnético. Quer para varrimentos, quer para os mapeamentos também. Com o plano de missões é possível ter voos estáveis, quer ao nível da altitude quer ao nível da velocidade, onde estas são as principais funcionalidades usadas na configuração do drone. Para planear a missão é sempre necessário definir a Base, que é o local de onde o drone irá levantar e onde irá também aterrar. Depois definem-se os *waypoints* que se quiser para se conseguir obter um varrimento, um mapeamento, um círculo, um rectângulo, seja o que for, é possível fazer o que se quiser, basta adicionar-se os *waypoints* ao mapa. Neste projecto são feitos varrimentos em linha recta, ir e voltar no mesmo percurso e

mapeamentos também ir e voltar mas com cerca de 2 metro a 2,5 metros entre cada ida e cada volta (Figura 57). [48]

- **Simulação**

Depois de criada a missão é necessário simulá-la, ou seja, para verificar que a mesma é exequível pelo drone real é necessário verificar a sua executabilidade através do Mission Planner com a funcionalidade Simulação. Para isso basta carregar o ficheiro de Plano de Voo criado anteriormente e executar a simulação, no final analisam-se os resultados, através dos *logs* gerados e da informação obtida através do Mission Planner e Ardupilot, verificando qual o estado da bateria e qual o comportamento do drone. Caso tenham sido adicionadas condições severas quer de condições meteorológicas quer de altitude, se se verificar o sucesso da missão no Mission Planner é praticamente garantido que na realidade também a conseguiremos fazer, claro que é necessário que as condições simuladas se assemelhem à realidade ou de preferência que as reais sejam melhores. [49]

7.3. Estudo de Interferências e Mapeamentos

- **Estudo de interferências provocadas pelo drone**

Uma das grandes preocupações relativamente ao desenvolvimento deste projecto, está relacionada com as possíveis interferências/ruídos causadas na medida do campo magnético devido aos motores do drone em funcionamento, tendo em conta a grande quantidade de corrente exigida por estes em determinadas alturas. De forma a aferir se esta é ou não uma problemática, efectuam-se alguns testes, mede-se em redor do drone com o sistema desenvolvido e posteriormente analisam-se esses valores.

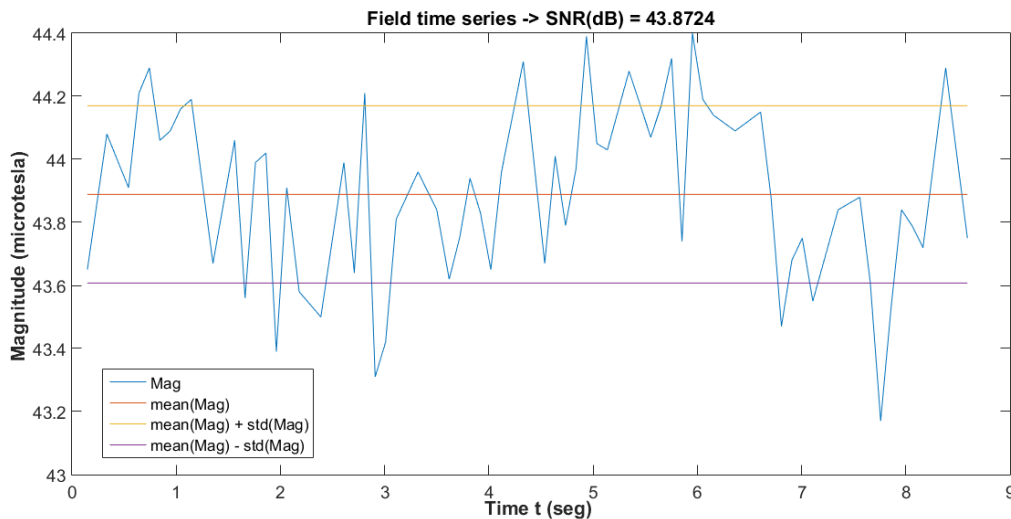


Figura 58 -Análise de interferência/ruído com motores a armar (Passar por cima).

Ao passar-se por cima do drone e medir-se com o sistema, enquanto os motores do drone estão a armar, verifica-se que não existe qualquer interferência/ruído perceptíveis que alterem significativamente os valores medidos. Notam-se algumas flutuações nos valores medidos mas que são derivadas à presença de componentes electrónicos que constituem o drone, verifica-se que o valor médio medido são cerca de 43,9 microTesla, e o desvio padrão de aproximadamente 0,3 microTesla, sendo que a relação sinal-ruído (SNR) é de 43,87 dB (Figura 58).

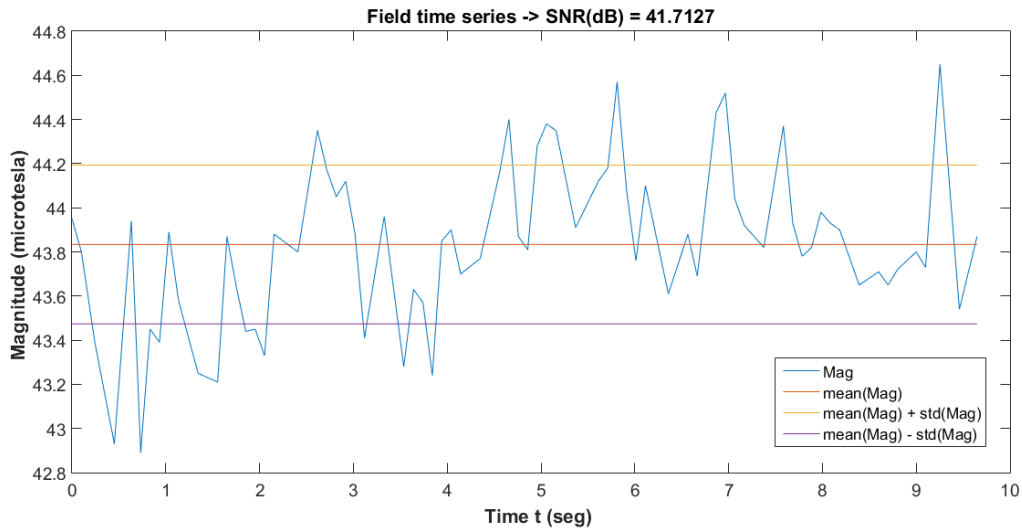


Figura 59 - Análise de interferência/ruído com motores a armar (Parar em cima).

O segundo teste foi idêntico ao primeiro mas desta vez, em vez de se passar por cima do drone a armar, parou-se em cima do mesmo e esperou-se que o processo de armar chegasse ao fim. Como é observável no gráfico anterior, também não se verificou qualquer interferência/ruído significativa, obtendo-se um valor médio medido de aproximadamente 43,65 microTesla, um desvio padrão de 0,45 microTesla e uma relação sinal-ruído de 41,67 dB (Figura 59).

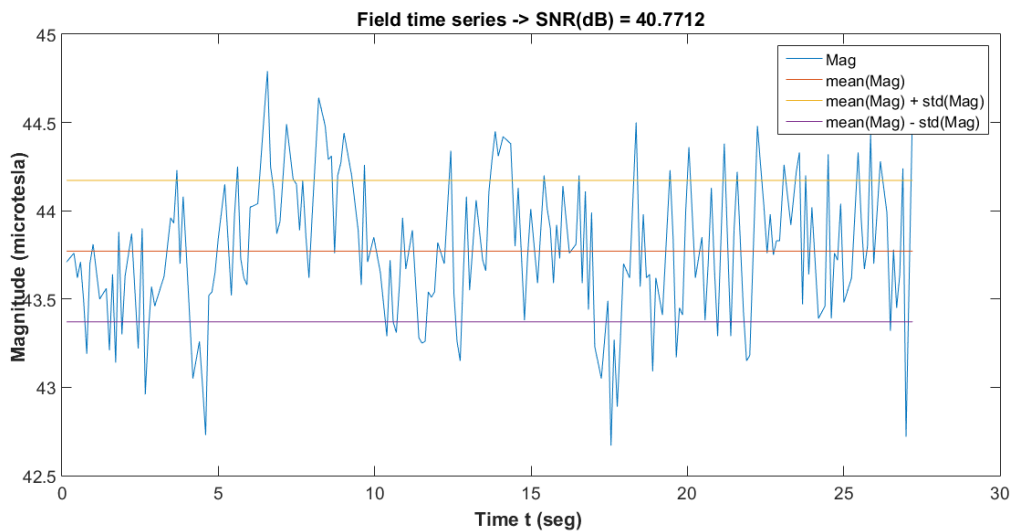


Figura 60 - Análise de interferência/ruído com motores a acelerar bruscamente.

Finalmente, realizou-se outro teste que foi medir sobre o drone enquanto os motores deste são acelerados com bastante velocidade, provocando uma grande quantidade de corrente eléctrica debitada. Devido à frequência a que os motores do

drone trabalham, não se verificam quaisquer anomalias provocadas no sistema. Como é facilmente comprovável pelos valores obtidos, um valor médio do campo magnético de 43,85 microTesla, um desvio padrão de 0,45 microTesla e uma SNR de 40,77 dB (Figura 60). Em todos os testes o valor do campo magnético manteve-se aproximadamente nos 43,8 microTesla o que comprava que para a frequência de trabalho utilizada não existem problemas relacionados com a interferência provocada pelos motores do drone em funcionamento, nem mesmo quando acelerados a elevada rotação.

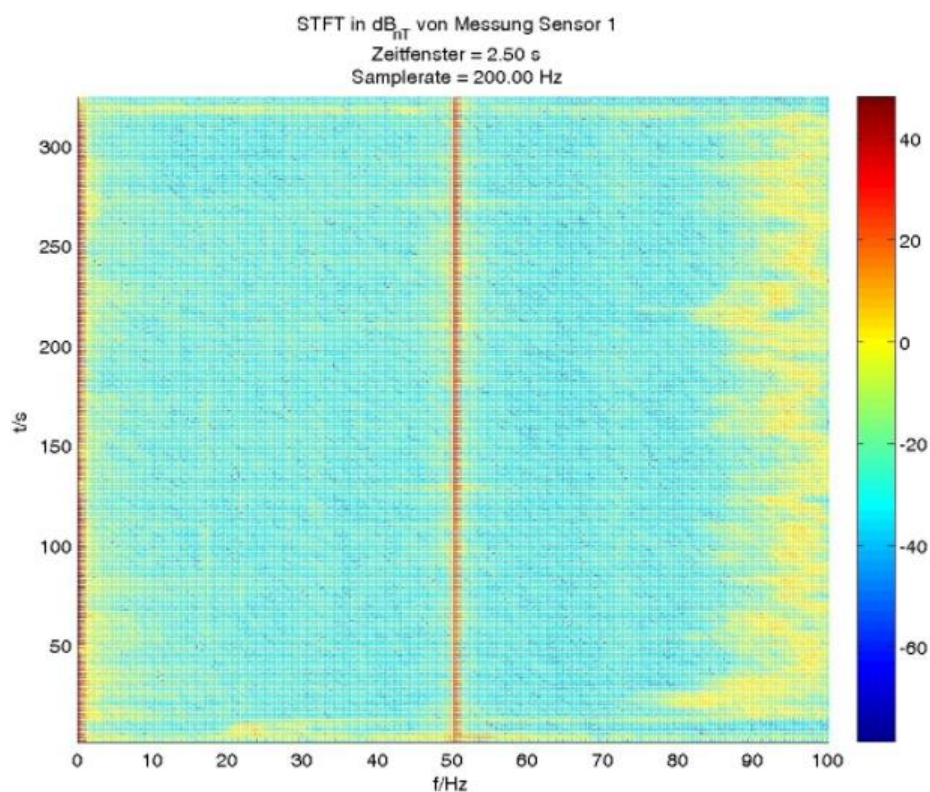


Figura 61 - Gráfico que ilustra a inexistência de interferências provocadas pelo funcionamento dos motores do drone. [17]

Não são observáveis quaisquer anomalias provocadas por estes, prova disso é também dada pelo gráfico disponibilizado por uma das empresas mencionadas anteriormente, que produz magnetômetros acopláveis a dispositivos específicos de medida, mais concretamente a Sensys. As frequências utilizadas para a obtenção de dados relativos à variação do campo magnético são normalmente entre 10 Hz e 40 Hz, ou seja, nesta zona os sinais não sofrem qualquer tipo de influência provocada pela drone, visto que as frequências de rotação dos motores ao ligarem rodam os 70 Hz a 80 Hz, assim como os 50 Hz também não são uma problemática para o sistema (Figura 61). [17]

- **Varrimentos de Nordeste para Sudoeste**



Figura 62 - Varrimentos lineares de nordeste para sudoeste e vice-versa.

Nestes varrimentos o percurso efectuado é aproximadamente linear de nordeste para sudoeste e vice-versa (Figura 62), tem uma extensão aproximada de 141 metros e o tempo médio de cada trajecto, um de ida de nordeste para sudoeste e outro de regresso de Sudoeste para Nordeste, foi de aproximadamente 117 segundos, o que dá uma velocidade média de cerca de 1,2 m/s, ou seja, 4,3 km/h. Sendo que após sincronização dos *logs* e análise, confirma-se que é também essa a velocidade registada pela controladora de voo do drone, para além disso a altitude média desta trajectória linear de aquisição é de cerca de seis metros.

A aquisição do campo magnético através do magnetómetro, ou seja, o registo das três componentes deste, é feita a uma taxa de 10Hz.

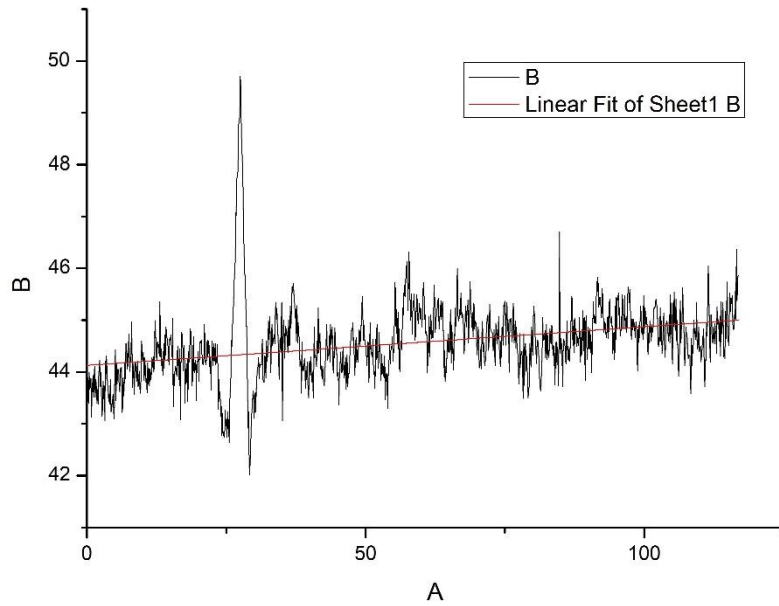


Figura 63 - Resultado da média dos dois perfis e evolução do campo regional.

O gráfico anterior (Figura 63) é o resultado da média dos dois perfis onde se evidencia um ligeiro mas contínuo aumento do campo regional de cerca de 1,5 unidades.

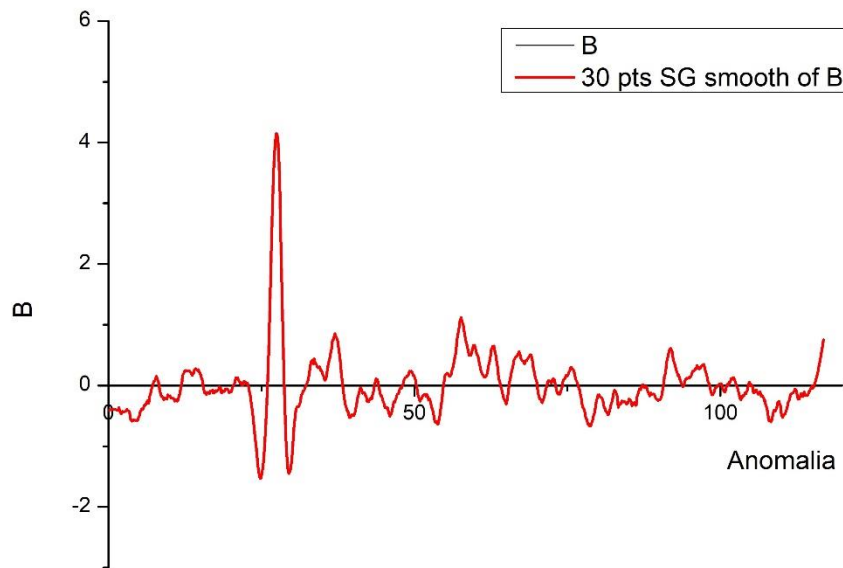


Figura 64 - Resultado da subtração do campo regional aos dois perfis.

Neste gráfico (Figura 64) é possível observar o resultado da subtração do campo regional, sintetizado, neste caso de forma muito simples através de uma linha, obtida por interpolação dos dados anteriores. Este gráfico mostra assim a anomalia provocada pelo carro, agora sem a influência do campo regional. Estes dados foram também filtrados com um filtro Savitzky-Golay que é um filtro simples instalado no *software* Origin. Este filtro é basicamente um *moving average filter*, janela de 15 pontos, que tende a suavizar as flutuações de alta frequência e a isolar melhor as de baixa frequência, mas também diminui ligeiramente a sua intensidade.

Nestes dados, ao longo do perfil há claramente anomalias de alta frequência, moduladas por variações de mais baixa frequência, estas são as que devem estar relacionadas com fontes superficiais no solo. Neste perfil é bem evidenciado a passagem nas proximidades do carro, perto da posição 25 com uma anomalia bem destacada com um valor da ordem de 4 microTesla.

- **Mapeamento de Nordeste para Sudoeste**

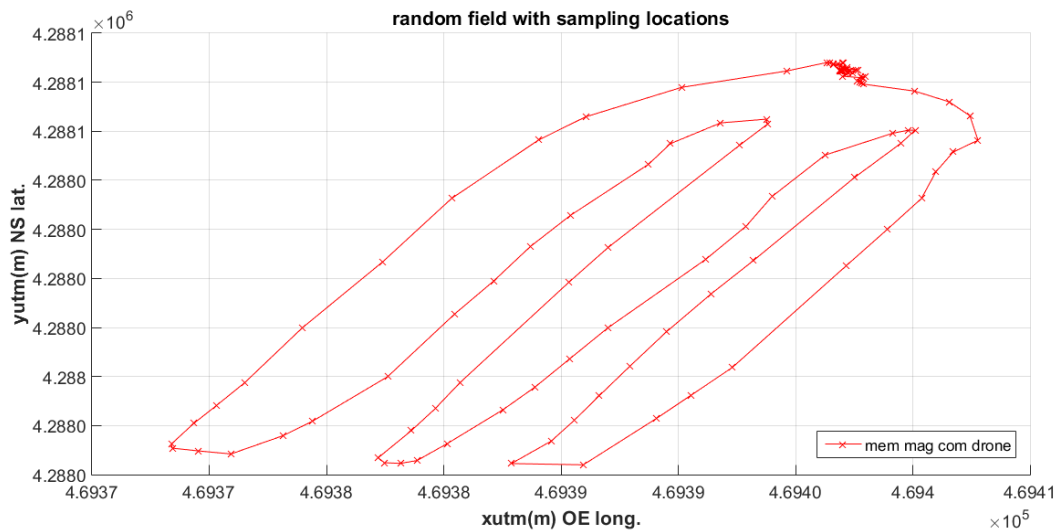


Figura 65 - Representação dos vários pontos medidos com recurso ao drone.

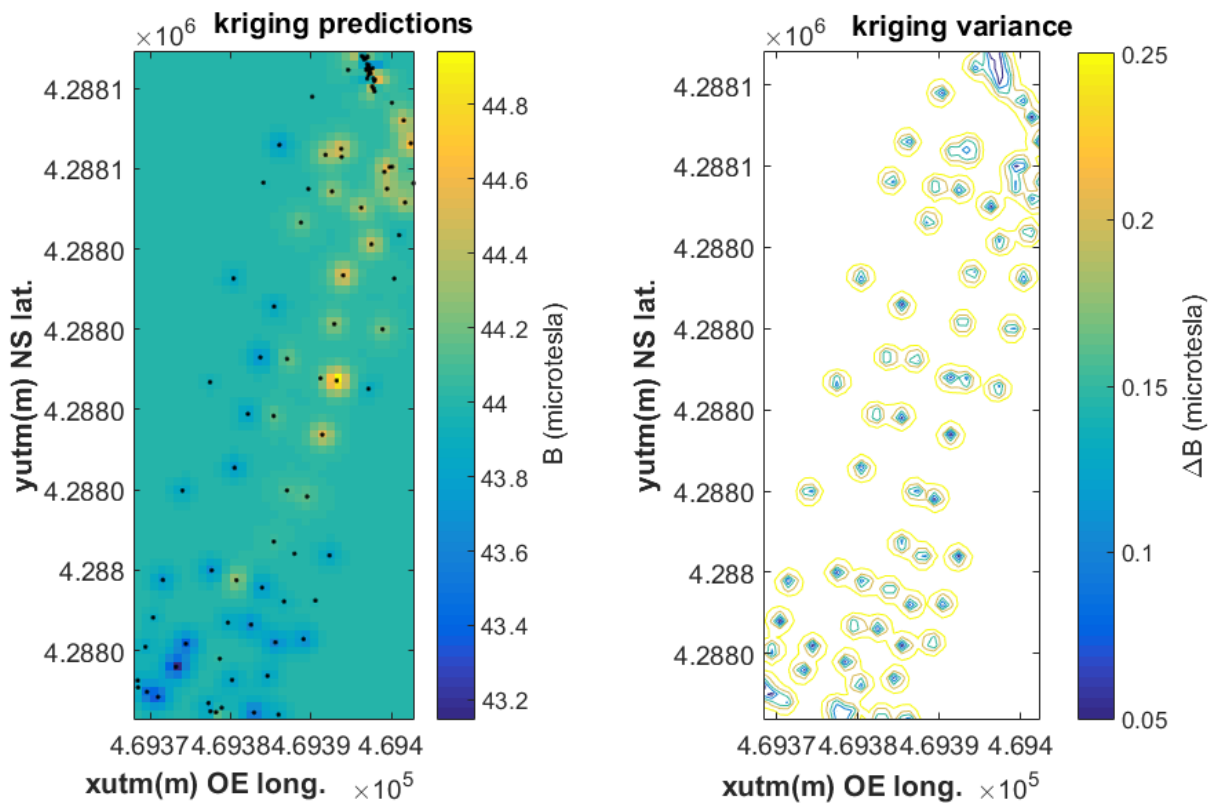


Figura 66 - Interpolação e obtenção da variância com o método de Kriging para os valores obtidos com o sistema desenvolvido com recurso ao drone.

Tal como nos mapeamentos a pé o mapeamento com o drone é feito no sentido, sensivelmente, de Nordeste para Sudoeste -> Sudoeste para Nordeste (Figura 65). Este voo é planeado e simulado no Mission Planner e posteriormente carregado na controladora de voo do drone, permitindo desta forma realizar um mapeamento completamente autónomo e sem qualquer intervenção via comando RF. O voo é realizado a 6 metros de altitude e através do tratamento e análise com o método de Kriging das medições é possível verificar que estas são muito próximas daquilo que foram os resultados quer obtidos manualmente com o magnetómetro de Césio quer obtidos manualmente com o sistema desenvolvido (Figura 66).

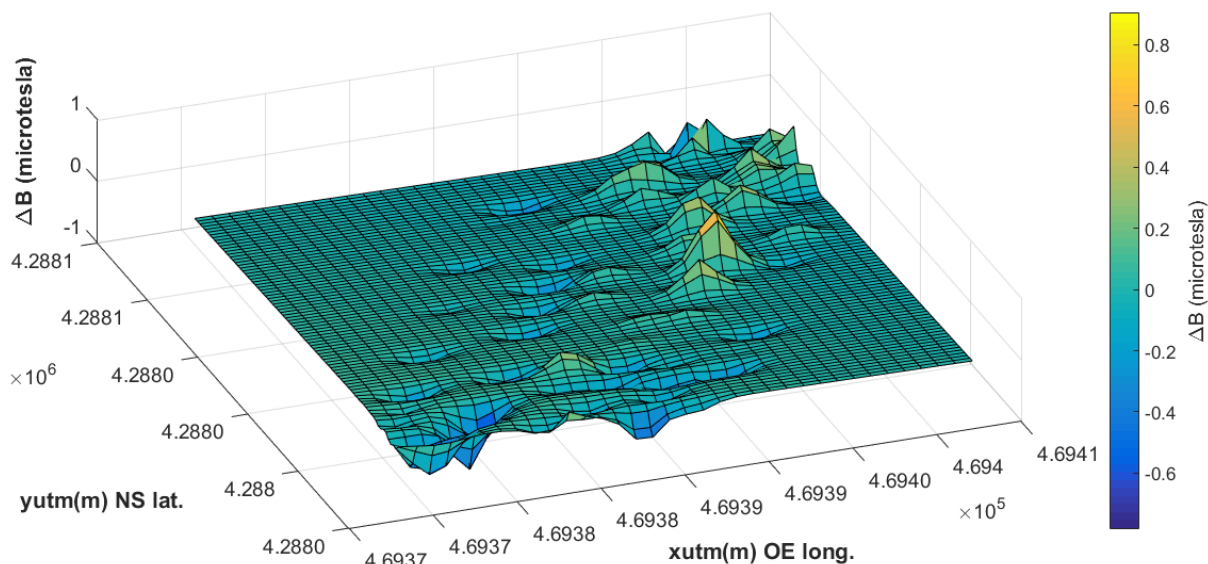


Figura 67 - Representação em gráfico 3D dos valores obtidos com o sistema desenvolvido com recurso ao drone, com a subtração do plano regional efectuada.

É claramente identificada a anomalia provocada pelo carro, sendo que esta apesar de posicionado num local diferente dos mapeamentos manuais, é claramente identificável. Tem nessa zona um valor residual, superior ao valor do campo magnético local, de aproximadamente 0,6 a 0,8 microTesla, onde o campo regional tem um valor médio de 43,8 microTesla e um alinhamento segundo Nordeste/Sudoeste (Figura 67). Para além dessa anomalia elevada verifica-se a existência de várias anomalias residuais mais pequenas, que tal como nos mapeamentos manuais são provocadas pela acumulação de uma grande quantidade de resíduos no local.

8. Discussão e Conclusões

- **Discussão**

Partindo do início, é constatável que de um modo geral a maioria dos objectivos definidos para o projecto foram alcançados quer ao nível do seu desenvolvimento e construção quer ao nível do seu teste e aplicabilidade no terreno.

Na definição do sistema e planeamento houve algumas dúvidas relativamente aos dispositivos a escolher, isto porque alguns deles ou não tinham a dimensão desejada ou não eram de integração fácil ou o seu suporte por parte do fabricante não é o melhor mas para além disso devido ao facto de alguns dispositivos inicialmente parecerem adequados, posteriormente, ao iniciar-se o desenvolvimento e implementação, verifica-se que seria melhor optar-se por outra opção. Quanto ao planeamento foi cumprido na sua grande maioria, com a metodologia de testes, de medições manuais e dos voos a ser fundamental para o trabalho realizado, com essa esquematização fica-se com um “guia” de tudo aquilo que é necessário fazer, o que é sem dúvida fundamental.

No desenvolvimento e implementação, a integração dos vários dispositivos não apresentou dificuldades, apenas a forma como estes se posicionariam gerou algum trabalho e estruturação, visto que para além de este ser um sistema independente também pode ser acoplado a qualquer sistema específico de medida. Quanto à estruturação do código, para a criação de um autómato, foi essencial para que o sistema funcione sem problemas. Nesta etapa do projecto, como inicialmente se pensou na realização da calibração no próprio dispositivo, fez-se várias tentativas de utilizar cálculo matricial no Arduino, mas as bibliotecas existentes para o cálculo matricial são bastante pesadas e ocupam muita memória, limitando desta forma a construção do autómato e a implementação das funcionalidades necessárias para o sistema, assim sendo, acabou por serem usados outros métodos de calibração

Na calibração, como esta é fundamental neste sistema, foram explorados vários mecanismos para a sua realização, sendo que os resultados obtidos para todos eles são muito semelhantes, o que gerou confiança nos vários processos de calibração, podendo ser utilizado qualquer um deles. Foram utilizados *software* já desenvolvidos e implementados, prontos a usar e outros processos provenientes de

desenvolvimentos *open source*, quer de fabricantes de sensores magnéticos quer da comunidade ligada ao estudo de campos magnéticos.

Tal como definido nos objectivos do projecto, nos testes e validação do sistema foram realizados vários estudos sobre o desempenho do sistema, foram obtidos bons resultados nos testes realizados com as bobines de Helmholtz, obtendo-se uma representação precisa do campo magnético gerado por estas, para além disso os valores obtidos em medidas de longa duração registaram uma boa estabilidade dos valores obtidos. Posteriormente, já no campo, na correlação entre o sistema desenvolvido e o magnetómetro de céσιο obtiveram-se resultados idênticos e uma correlação razoável, que foi ligeiramente limitada porque a precisão de ambos é relativamente diferente, influencia um pouco os resultados, mas de qualquer forma os valores obtidos são bons. Nos mapeamentos de comparação entre o magnetómetro de céσιο e o sistema desenvolvido constatou-se que foram observáveis e caracterizáveis de forma clara as anomalias provocadas pelo carro que alterou significativamente o campo magnético da área terrestre estudada.

Com o intuito de mapear áreas terrestres procedeu-se a medições manuais das referidas áreas e analisou-se as mesmas com recurso ao método de Kriging que trata da interpolação dos valores obtidos com recurso a métodos geoestatísticos, baseados em processos Gaussianos, resultando assim em representações gráficas que caracterizam de forma clara e assertiva o campo magnético das áreas terrestres analisadas, sendo que este método para medições onde são adquiridos muitos pontos nem sempre apresenta resultados claros.

Finalmente, nos mapeamentos com drone, repetiram-se os mapeamentos realizados de forma manual, de forma a se aferir quais as semelhanças ou diferenças entre os valores obtidos anteriormente, em mapeamentos manuais, e estes obtidos com recurso ao drone. Constatou-se então pequenas diferenças, primeiro porque a altitude a que o drone voou foi superior à altitude de realização dos mapeamentos manuais, os voos foram cerca de cinco metros mais altos e depois porque o posicionamento da anomalia (carro) é ligeiramente diferente. De qualquer forma os valores obtidos são bastante idênticos e caracterizam de forma claro a área terrestre em causa. Para a realização dos voos foram criadas missões e simulados os voos no Mission Planner, garantindo desta forma a executabilidade destas. Para além disso, foi feito o estudo das possíveis interferências provocadas pelos motores do drone em

funcionamento mas verificou-se que para a frequência de aquisição utilizada, 10 Hz, através dos testes realizados em redor do drone com o sistema desenvolvido, não existem alterações significativas que alterem os valores do campo magnético local.

- **Conclusões**

Finalizado o desenvolvimento e teste do sistema e posterior análise dos dados obtidos obteve-se não só uma ferramenta versátil mas também robusta e precisa, que permite ser utilizada em diversas aplicações, quer em mapeamento mais rudimentares, diga-se manuais, quer sendo acoplado a dispositivos específicos de medida. Para além disso é um sistema construído com custo reduzido, cerca de 100 vezes inferior ao valor dos sistemas comerciais actualmente vendidos, que são de aproximadamente 15000€, com uma eficiência boa, que permite caracterizar bem o campo magnético em diversas áreas terrestres, detectando com precisão as anomalias existentes.

É um sistema com uma sensibilidade mais reduzida que o magnetómetro de Césio utilizado para comparação mas o objectivo é precisamente desmistificar esta questão, de se ter um sistema muito preciso mas também de custo muito elevado, ou seja, criar um sistema capaz de obter dados de uma gama razoável, neste caso com uma gama de ± 16 Gauss, e que tenha um custo reduzido.

A integração do sistema neste caso concreto no drone mas em qualquer outro dispositivo é fácil e está bem estruturada, pois possibilita manter a estabilidade destes dispositivos. Quanto às possíveis interferências provocadas pelos motores deste em funcionamento, que inicialmente se pensava ser um problema, veio a confirmar-se que na gama de frequências de trabalho utilizadas não há qualquer problema desta medida.

Quanto aos dados obtidos é possível ter vários níveis de comparação, quer através de mapeamentos manuais, obtidos a uma altitude média de um metro e com o drone a aproximadamente seis metros. Os resultados obtidos são muito próximos e traduzem perfeitamente as características do campo magnético no local estudado. Definem bem a anomalia provocada pelo carro, pelo que é possível afirmar que facilmente identifica características de anomalias provocadas por rochas magnéticas e estruturas que alteram magneticamente o campo magnético envolvente.

- Datasheet do Sistema Desenvolvido

Tabela 9 – Datasheet do Sistema Desenvolvido.

Parâmetro	Especificação Típica	Unidades
Escala de Medição	±400	μTesla
Sensibilidade	0.015	μTesla
Estala de Frequência	1 – 80	Hz
Durabilidade Cartão SD	10000	inserções/remoções
Tipo de Cartão	Classe 6/ microSD	-----
Precisão GPS	2	metros
Arranque a frio GPS	27	segundos
Arranque a quente GPS	1	segundos
Sensibilidade GPS	-161dBm	dBm
Número de Canais GPS	50	-----
Capacidade da Bateria	1300/1800	mAh
Tensão de Alimentação	7,4/11	V
Durabilidade da Bateria	45/65	horas
Peso Total	300	grama
Faixa de temperatura operacional	-40 a +85	°C

- **Trabalhos Futuros**

Ao longo do desenvolvimento do projecto surgiram algumas ideias para possíveis trabalhos futuros, não só porque em alguns momentos a utilidade destes teria sido uma mais-valia mas também porque este sistema foi desenvolvido de forma modular e com o intuito de ser um sistema autónomo, o que permite ser adaptado e utilizado noutras aplicações. Assim sendo, são feitas algumas sugestões para trabalhos futuros:

Sistema de seis bobines de Helmholtz que permita gerar um campo igual em todas as direcções e sentidos dos eixos X, Y e Z do magnetómetro, de forma a viabilizar uma melhor análise da estabilidade e calibração de magnetómetros 3D.

Dispositivo específico de medida onde se possa acoplar este sistema ou outro idêntico de forma a se realizar caracterização do campo magnético *indoor*.

Aplicação de telemóvel que possibilite ligação via BLE e que permita efectuar várias operações no sistema, efectuar a calibração do magnetómetro, mostrar valores lidos em tempo real, possibilitar a alteração da frequência de aquisição, entre outros.

Referências Bibliográficas

- [1] The Fragmentation Paradox -
http://thefragmentationparadox.blogspot.pt/2014/03/electromagnetic-fields-emf-in-high_16.html
- [2] Outline of a small unmanned aerial vehicle (Ant-Plane) designed for Antarctic research, Minoru Funaki, Naohiko Hirasawa, the Ant-Plane Group, National Institute of Polar Research, Japan 2008
- [3] Designing and building an unmanned aircraft system for aeromagnetic surveying, Claire Samson, Paul Straznicky, Jeremy Laliberté, Raymond Caron, Carleton University; Stephen Ferguson, Reed Archer, Sander Geophysics Limited, Ottawa, Ontario 2010
- [4] Aeromagnetic surveying with unmanned aircraft systems by Michael Cunningham B.Sc. (Hons.) Physics, 2010 and B.Sc. (Hons.) Earth Sciences, 2014
- [5] High-Resolution Ground-Based Magnetic Survey of a Buried Volcano: Anomaly B, Amargosa Desert, NV George, O., McIlrath, J., Farrell, A., Gallant, E., Kinman, S., Marshall, A., McNiff, C., Njoroge, M., Wilson, J., Connor, C. B., Connor, L. J., Kruse, S. University of South Florida, School of Geosciences, Tampa, Florida USA, February 2015.
- [6] An aeromagnetic survey of Shinmoe-dake volcano, Kirishima, Japan, after the 2011 eruption using an unmanned autonomous helicopter Takao Koyama, Takayuki Kaneko, Takao Ohminato, Takatoshi Yanagisawa, Atsushi Watanabe, and Minoru Takeo, Earthquake Research Institute, University of Tokyo, 1-1-1 Yayoi, Bunkyo-ku, Tokyo 113-0032, Japan and IFREE, Japan Agency for Marine-Earth Science and Technology, 2-15 Natsushima-cho, Yokosuka, Kanagawa, Japan, July 2013.
- [7] PFC - Análise e Medida de Campos Electromagnéticos próximo de objectos em movimento - P203, Valter Rodrigues, Setembro 2012
- [8] MagDrone - Construção de um drone quadrirotor com posicionamento de precisão para cartografia magnética, ISEL, Dezembro 2017
- [9] http://www.ortung.com/en/?Magnetometer_measurement; <http://www.aegis-instruments.com/products/brochures/gsm-30.html>;
<https://www.hunebednieuwscade.nl/2015/03/heel-veel-hunebedden-in-denemarken/>;
<http://www.gemsys.ca/site-characterization-for-using-overhauser-magnetometer/>;

[https://www.geomatrix.co.uk/land-products/magnetic/g-857/;](https://www.geomatrix.co.uk/land-products/magnetic/g-857/)

<http://www.bartington.com/presentation/grad601-single-axis-magnetic-gradiometer-system>

[10] <http://www.sensysmagnetometer.com/en/emd2.html>

[11] http://www.drjamesnagel.com/EM_Beauty.htm

[12] <http://www.geol-amu.org/notes/m10-1-10.htm>

[13] GEOPHYSICAL MAPS OF SPAIN, Martínez Solares, J.M., Gil, A., Socias, I., Mezcuá, J., Martín, A.J. and Marín, V., Instituto Geográfico Nacional. Gral. Ibañez de Ibero, Madrid (Spain) e Servicio Regional del Instituto Geográfico Nacional. Edf. Sevilla 2, Sevilla (Spain), 2015

[14] Aeromagnetic surveying with unmanned aircraft systems, Michael Cunningham, Carleton University, Ottawa, Ontario, 2016

[15] UNMANNED AIRCRAFT SYSTEMS FOR RAPID NEAR SURFACE GEOPHYSICAL MEASUREMENTS, Johannes B. Stoll, Mobile Geophysical Technologies, Germany, 2014

[16] Aeromagnetic surveying with unmanned aircraft systems by Michael Cunningham, Carleton University Ottawa, Ontario, 2016

[17] <http://www.sensysmagnetometer.com/en/magdrone-two.html>

[18] <http://www.gemsys.ca/uavs-pathway-to-the-future/>

[19] Datasheet Pololu LIS3MDL 3-Axis Magnetometer Carrier with Voltage Regulator

[20] Datasheet Sparkfun MicroMag Magnetometer

[21] Datasheet Sparkfun Mag3110 Magnetometer

[22] Datasheet Arduino 101 Intel® Curie

[23] Datasheet Micro

[24] Datasheet Uno

[25] Datasheet Mega

[26] Datasheet Adafruit MicroSD card breakout board

[27] Datasheet Itead Studio SD Shield 3.0

[28] Datasheet Ublox GY-NEO6MV2 GPS Module

[29] Datasheet ITEAD Studio RoyalTek REB-4216 GPS Shield

[30] Datasheet TopFuel LiPo 25C-ECO-X 1300mAh 2S

[31] Datasheet TopFuel LiPo 25C-ECO-X 1800mAh 3S

- [32] Lib LIS3MDL - <https://github.com/pololu/lis3mdl-arduino>
- [33] Lib SD - <https://github.com/arduino-libraries/SD>
- [34] Lib TinyGPS++ - <http://arduiniana.org/libraries/tinygpsplus/>
- [35] Calibração - https://teslabs.com/articles/magnetometer-calibration/#mjx-eqn-eq_quad_gen
- [36] MagMaster - <https://diydrone.com/profiles/blogs/advanced-hard-and-soft-iron-magnetometer-calibration-for-dummies?id=705844%3ABlogPost%3A1676387&page=4#comments>
- [37] https://commons.wikimedia.org/wiki/File%3AXYZ-DIS_magnetic_field_coordinates.svg
- [38] <https://www.ngdc.noaa.gov/geomag/WMM/DoDWMM.shtml>
- [39] <http://www.geom.uiuc.edu/docs/reference/CRC-formulas/node61.html>
- [40] Complete Triaxis Magnetometer Calibration in the Magnetic Domain, Valérie Renaudin, Muhammad Haris Afzal and Gérard Lachapelle, Position Location and Navigation (PLAN) Group, Schulich School of Engineering, University of Calgary, 2500 University Drive N.W., Calgary, Canada
- [41] Least squares ellipsoid specific fitting, Qingde Li e J.G. Griffiths Dept. of Comput. Sci., Hull Univ., UK 2004 - <http://ieeexplore.ieee.org/document/1290055/>
- [42] Calibration of a magnetometer in combination with inertial sensors, Manon Kok, Thomas B. Schön and Fredrik Gustafsson of Division of Automatic Control, Linköping University, Sweden and Jeroen D. Hol and Henk Luinge of Xsens Technologies B.V., Enschede, the Netherlands, 2012 - <http://ieeexplore.ieee.org/document/6289882/>
- [43] <https://www.didaktik.physik.uni-muenchen.de/elektronenbahnen/fr/b-feld/B-Feld/Helmholtzspulenpaar.php>
- [44] <http://mundogeo.com/blog/2013/06/05/coordenadas-topograficas-x-coordenadas-utm/>
- [45] ftp://geom.geometrics.com/pub/mag/DataSheets/G-859_DataSheet-SP.pdf
- [46] 1. Burrough, P. A. Principles of Geographical Information Systems for Land Resources Assessment. New York: Oxford University Press. 1986. 2. Heine, G. W. "A Controlled Study of Some Two-Dimensional Interpolation Methods." COGS Computer Contributions 3 (no. 2): 60–72. 1986. 3. McBratney, A. B., and R. Webster. "Choosing Functions for Semi-variograms of Soil Properties and Fitting Them to Sampling

Estimates." Journal of Soil Science 37: 617–639. 1986. **4.**Oliver, M. A. "Kriging: A Method of Interpolation for Geographical Information Systems." International Journal of Geographic Information Systems 4: 313–332. 1990. **5.**Press, W. H., S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery. Numerical Recipes in C: The Art of Scientific Computing. New York: Cambridge University Press. 1988. **6.**Royle, A. G., F. L. Clausen, and P. Frederiksen. "Practical Universal Kriging and Automatic Contouring." Geoprocessing 1: 377–394. 1981. -
<http://desktop.arcgis.com/en/arcmap/10.3/tools/3d-analyst-toolbox/how-kriging-works.htm>

[47] Ardupilot - <http://ardupilot.org/>

[48] Mission Planner - <http://ardupilot.org/planner/>

[49] Planeamento de Missões - <http://ardupilot.org/planner/docs/common-mission-planning.html>

[50] Datasheet MagDrone - Construção de um drone quadrirotor com posicionamento de precisão para cartografia magnética.

[51] Magneto - <https://sites.google.com/site/sailboatinstruments1/home>

Anexo A2

- **Código Autómato**

```
#include <Wire.h>
#include <LIS3MDL.h>
#include <SPI.h>
#include <SD.h>
#include <TinyGPS++.h>
#define CONFIG 0
#define STANDBY 1
#define COMPENSACAO 2
#define TESTES 3
#define GRAVAR 4
#define BUTTON1 2
#define BUTTON2 3
#define BUTTON3 4
#define RED 5
#define GREEN 6
#define BLUE 7
LIS3MDL mag;
int state = STANDBY;
//#define ARDUINO_SD_CS 4 //SD Shield 3.0
#define ARDUINO_SD_CS 10 //Adafruit Breakout Micro-SD
int fileCount = 0;
int testCount = 0;
int countValues = 0;
unsigned long initialTime = millis();
boolean calibrated = false;
//arduino example code for getting the calibrated magnetometer data
//calibrated_values[3] is the global array where the calibrated data will be
placed
//calibrated_values[3]: [0]=Xc, [1]=Yc, [2]=Zc
float calibrated_values[3];
float magBias[3],magScale[3];
float mRes = 10.*4912./32760.0;
//transformation(float uncalibrated_values[3]) is the function of the
magnetometer data correction
//uncalibrated_values[3] is the array of the non calibrated magnetometer data
//uncalibrated_values[3]: [0]=Xnc, [1]=Ync, [2]=Znc
```

```

float uncalibrated_values[3];
//////////
// Log File Defintions //
//////////
// Keep in mind, the SD library has max file name lengths of 8.3 - 8 char prefix,
// and a 3 char suffix.
// Our log files are called "gpslogXX.csv, so "gpslog99.csv" is our max file.
#define LOG_FILE_PREFIX "gpslog" // Name of the log file.
#define MAX_LOG_FILES 100 // Number of log files that can be made
#define MAX_COUNT_VALUES 1500 // Number of log values that can be made for
calibration
#define LOG_FILE_SUFFIX "csv" // Suffix of the log file
char logFileName[13]; // Char string to store the log file name
// Data to be logged:
#define LOG_COLUMN_COUNT 13
char * log_col_names[LOG_COLUMN_COUNT] = {"milis",
"longitude", "latitude", "altitude", "speed", "course",
"date", "time", "satellites", "X", "Y", "Z", "Mag"}; // log_col_names is
printed at the top of the file.
//////////
// Log Rate Control //
//////////
#define LOG_RATE 10 // 10 Hz
unsigned long lastLog = 0; // Global var to keep of last time we logged
unsigned long startTime = 0; // Global var to get start time log
unsigned long saveSDTime = 0; // Global var to get start time log
//////////
// TinyGPS Definitions //
//////////
TinyGPSPlus tinyGPS; // tinyGPSPlus object to be used throughout
#define GPS_BAUD 9600 // GPS module's default baud rate
#define SERIAL_BAUD 115200 // Serial default baud rate
//////////
// Arduino GPS Shield Serial Port Definitions //
//////////
// If you're using an Arduino Uno, Mega, RedBoard, or any board that uses the
// 0/1 UART for programming/Serial monitor-ing, use SoftwareSerial:
#include <SoftwareSerial.h>
#define ARDUINO_GPS_RX 9 // GPS TX, Arduino RX pin
#define ARDUINO_GPS_TX 8 // GPS RX, Arduino TX pin
SoftwareSerial ssGPS(ARDUINO_GPS_RX, ARDUINO_GPS_TX); // Create a
SoftwareSerial
// Set gpsPort to either ssGPS if using SoftwareSerial or Serial1 if using an

```

```

// Arduino with a dedicated hardware serial port
// #define gpsPort ssGPS // Alternatively, use Serial1 on the Leonardo
// Define the serial monitor port. On the Uno, Mega, and Leonardo this is 'Serial'
// on other boards this may be 'SerialUSB'
#define SerialMonitor Serial
File logFile; // Open the log file
void setup()
{
  SerialMonitor.begin(SERIAL_BAUD);
  Wire.begin();
  pinMode(BUTTON1, INPUT_PULLUP);
  pinMode(BUTTON2, INPUT_PULLUP);
  pinMode(BUTTON3, INPUT_PULLUP);
  pinMode(RED, OUTPUT);
  pinMode(GREEN, OUTPUT);
  pinMode(BLUE, OUTPUT);
  if (!mag.init())
  {
    SerialMonitor.println("Failed to detect and initialize magnetometer!");
    while (1);
  }
  mag.enableDefault();
  delay(500);
  ssGPS.begin(GPS_BAUD);
  delay(500);
  SerialMonitor.print("Initializing SD card...");
  // see if the card is present and can be initialized:
  if (!SD.begin(ARDUINO_SD_CS)) {
    SerialMonitor.println("Card failed, or not present");
    // don't do anything more:
    return;
  }
  SerialMonitor.println("Card initialized.");
  delay(500);
}
void loop()
{
  automato();
}
void medirValores()
{
  mag.read();
  uncalibrated_values[0] = mag.m.x / 100.0;
}

```

```

uncalibrated_values[1] = mag.m.y / 100.0;
uncalibrated_values[2] = mag.m.z / 100.0;
// Calculate the magnetometer values in milliGauss
// Include factory calibration per data sheet and user environmental corrections
//uncalibrated_values[0] = (float)mag.m.x*mRes - magBias[0]; // get actual
magnetometer value, this depends on scale being set
//uncalibrated_values[1] = (float)mag.m.y*mRes - magBias[1];
//uncalibrated_values[2] = (float)mag.m.z*mRes - magBias[2];
transformation(uncalibrated_values);
}
void magValues()
{
if(countValues==0){
logFile = SD.open("calibValues.csv", FILE_WRITE);
logFile.println("X,Y,Z");
}
mag.read();
uncalibrated_values_for_calib[0] = mag.m.x / 100.0;
uncalibrated_values_for_calib[1] = mag.m.y / 100.0;
uncalibrated_values_for_calib[2] = mag.m.z / 100.0;
logFile.print(uncalibrated_values_for_calib[0]);
logFile.print(',');
logFile.print(uncalibrated_values_for_calib[1]);
logFile.print(',');
logFile.print(uncalibrated_values_for_calib[2]);
logFile.println();
logFile.close();
logFile = SD.open(logFileName, FILE_WRITE);
}
void mostrarValores()
{
//Serial.flush();
if ((lastLog + 1000/LOG_RATE) <= millis())
{
lastLog = millis(); // Update the lastLog variable
SerialMonitor.print(millis() - startTime);
SerialMonitor.print(",");
SerialMonitor.print(tinyGPS.location.lng(),7);
SerialMonitor.print(",");
SerialMonitor.print(tinyGPS.location.lat(),7);
SerialMonitor.print(",");
SerialMonitor.print(tinyGPS.altitude.meters());
SerialMonitor.print(",");
}
}

```

```

SerialMonitor.print(tinyGPS.speed.kmph());
SerialMonitor.print(",");
SerialMonitor.print(tinyGPS.course.deg());
SerialMonitor.print(",");
SerialMonitor.print(tinyGPS.date.value());
SerialMonitor.print(",");
SerialMonitor.print(tinyGPS.time.value());
SerialMonitor.print(",");
SerialMonitor.print(tinyGPS.satellites.value());
SerialMonitor.print(",");
SerialMonitor.print(calibrated_values[0]);
SerialMonitor.print(",");
SerialMonitor.print(calibrated_values[1]);
SerialMonitor.print(",");
SerialMonitor.print(calibrated_values[2]);
SerialMonitor.print(",");
SerialMonitor.print(sqrt(sq(calibrated_values[0]) + sq(calibrated_values[1])
+ sq(calibrated_values[2])));
SerialMonitor.println();
}
}
void transformation(float uncalibrated_values[3])
{
//calibration_matrix[3][3] is the transformation matrix
//replace M11, M12,...,M33 with your transformation matrix data
//inicial calibration
/* double calibration_matrix[3][3] =
{
{1.98, 0.144, -0.213},
{-0.056, 2.047, 0.099},
{0.073, 0.011, 2.047}
};
//bias[3] is the bias
//replace Bx, By, Bz with your bias data
double bias[3] =
{
24.986,
-15.469,
-7.664
}; */
//local calibration (Trajouce)
double calibration_matrix[3][3] =
{

```

```

{1.548, 0.032, 0.014},
{0.07, 1.61, 0.023},
{0.006, -0.043, 1.646}
};
//bias[3] is the bias
//replace Bx, By, Bz with your bias data
double bias[3] =
{
16.907,
-7.197,
-8.028
};
/*//local calibration (Venda Nova)
double calibration_matrix[3][3] =
{
{1.494, 0.02, -0.022},
{0.051, 1.522, 0.102},
{-0.004, -0.06, 1.558}
};
//bias[3] is the bias
//replace Bx, By, Bz with your bias data
double bias[3] =
{
16.687,
-7.782,
-7.733
}; */
//calculation
for (int i=0; i<3; ++i) uncalibrated_values[i] = uncalibrated_values[i] -
bias[i];
float result[3] = {0, 0, 0};
for (int i=0; i<3; ++i)
for (int j=0; j<3; ++j)
result[i] += calibration_matrix[i][j] * uncalibrated_values[j];
for (int i=0; i<3; ++i) calibrated_values[i] = result[i];
}
void automato()
{
switch(state){
case STANDBY:
whiteON();
SerialMonitor.println(state);
if(digitalRead(BUTTON1)==LOW){

```

```

SerialMonitor.println("Compensacao");
count = 0;
state = COMPENSACAO;
}
else if(digitalRead(BUTTON2)==LOW){
SerialMonitor.println("Test" + String(testCount) + ".");
state = TESTES;
lastLog = 0;
startTime = millis();
}
else if(digitalRead(BUTTON3)==LOW){
// open the file. note that only one file can be open at a time,
// so you have to close this one before opening another.
updateFileName(); // Each time we start, create a new file, increment the
number
logFile = SD.open(logFileName, FILE_WRITE);
printHeader(); // Print a header at the top of the new file
state = GRAVAR;
lastLog = 0;
startTime = millis();
}
break;
case COMPENSACAO:
redON();
if(!calibrated){
if(countValues<MAX_COUNT_VALUES)
magValues();
countValues++;
}else{
logFile.close();
calibrated = true;
}
}
else{
redOFF();
delay(500);
}
if(digitalRead(BUTTON2)==LOW || digitalRead(BUTTON3)==LOW){
logFile.close();
whiteON();
delay(1000);
state = STANDBY;
}
}

```

```

break;
case TESTES:
greenON();
if(digitalRead(BUTTON1)==LOW || digitalRead(BUTTON3)==LOW){
whiteON();
delay(1000);
state = STANDBY;
}
medirValores();
mostrarValores();
//delay(100);
break;
case GRAVAR:
blueON();
medirValores();
gpsTracker();
//delay(100);
if(digitalRead(BUTTON1)==LOW || digitalRead(BUTTON2)==LOW){
logFile.close();
whiteON();
fileCount++;
delay(1000);
state = STANDBY;
}
break;
}
}
void whiteON(){
digitalWrite(RED,HIGH);
digitalWrite(GREEN,HIGH);
digitalWrite(BLUE,HIGH);
}
void redON(){
digitalWrite(RED,HIGH);
digitalWrite(GREEN,LOW);
digitalWrite(BLUE,LOW);
}
void redOFF(){
digitalWrite(RED,LOW);
digitalWrite(GREEN,LOW);
digitalWrite(BLUE,LOW);
}
void greenON(){

```

```

digitalWrite(RED,LOW);
digitalWrite(GREEN,HIGH);
digitalWrite(BLUE,LOW);
}
void blueON(){
digitalWrite(RED,LOW);
digitalWrite(GREEN,LOW);
digitalWrite(BLUE,HIGH);
}
void magCalibration(float * dest1, float * dest2) {
uint16_t ii = 0, sample_count = 0;
int32_t mag_bias[3] = {0, 0, 0}, mag_scale[3] = {0, 0, 0};
int16_t mag_max[3] = {0x8000, 0x8000, 0x8000}, mag_min[3] = {0x7FFF, 0x7FFF,
0x7FFF}, mag_temp[3] = {0, 0, 0};
SerialMonitor.println("Mag Calibration: Wave device in a figure eight until
done!");
sample_count = 1500; // at 100 Hz ODR, new mag data is available every 10 ms
for(ii = 0; ii < sample_count; ii++) {
mag.read(); // Read the mag data
mag_temp[0] = mag.m.x;
mag_temp[1] = mag.m.y;
mag_temp[2] = mag.m.z;
for (int jj = 0; jj < 3; jj++) {
if(mag_temp[jj] > mag_max[jj])
mag_max[jj] = mag_temp[jj];
if(mag_temp[jj] < mag_min[jj])
mag_min[jj] = mag_temp[jj];
}
delay(135); // at 8 Hz ODR, new mag data is available every 125 ms
}
// Get hard iron correction
mag_bias[0] = (mag_max[0] + mag_min[0])/2; // get average x mag bias in counts
mag_bias[1] = (mag_max[1] + mag_min[1])/2; // get average y mag bias in counts
mag_bias[2] = (mag_max[2] + mag_min[2])/2; // get average z mag bias in counts
dest1[0] = (float) mag_bias[0]*mRes; // save mag biases in G for main program
dest1[1] = (float) mag_bias[1]*mRes;
dest1[2] = (float) mag_bias[2]*mRes;
// Get soft iron correction estimate
mag_scale[0] = (mag_max[0] - mag_min[0])/2; // get average x axis max chord
length in counts
mag_scale[1] = (mag_max[1] - mag_min[1])/2; // get average y axis max chord
length in counts
mag_scale[2] = (mag_max[2] - mag_min[2])/2; // get average z axis max chord

```

```

length in counts
float avg_rad = mag_scale[0] + mag_scale[1] + mag_scale[2];
avg_rad /= 3.0;
dest2[0] = avg_rad/((float)mag_scale[0]);
dest2[1] = avg_rad/((float)mag_scale[1]);
dest2[2] = avg_rad/((float)mag_scale[2]);
SerialMonitor.println("Mag Calibration done!");
}
byte writeSDCard()
{
if (logFile)
{ // Print longitude, latitude, altitude (in feet), speed (in mph), course
// in (degrees), date, time, and number of satellites.
logFile.print(millis() - startTime);
logFile.print(',');
logFile.print(tinyGPS.location.lng(), 7);
logFile.print(',');
logFile.print(tinyGPS.location.lat(), 7);
logFile.print(',');
logFile.print(tinyGPS.altitude.meters(), 1);
logFile.print(',');
logFile.print(tinyGPS.speed.kmph(), 1);
logFile.print(',');
logFile.print(tinyGPS.course.deg(), 1);
logFile.print(',');
logFile.print(tinyGPS.date.value());
logFile.print(',');
logFile.print(tinyGPS.time.value());
logFile.print(',');
logFile.print(tinyGPS.satellites.value());
logFile.print(',');
logFile.print(calibrated_values[0]);
logFile.print(',');
logFile.print(calibrated_values[1]);
logFile.print(',');
logFile.print(calibrated_values[2]);
logFile.print(',');
logFile.print(sqrt(sq(calibrated_values[0]) + sq(calibrated_values[1]) +
sq(calibrated_values[2])));
logFile.println();
logFile.close();
logFile = SD.open(logFileName, FILE_WRITE);

```

```

return 1; // Return success
}
return 0; // If we failed to open the file, return fail
}
// printHeader() - prints our eight column names to the top of our log file
void printHeader()
{
if (logFile) // If the log file opened, print our column names to the file
{
int i = 0;
for (; i < LOG_COLUMN_COUNT; i++)
{
logFile.print(log_col_names[i]);
if (i < LOG_COLUMN_COUNT - 1) // If it's anything but the last column
logFile.print(','); // print a comma
else // If it's the last column
logFile.println(); // print a new line
}
}
}
// updateFileName() - Looks through the log files already present on a card,
// and creates a new file with an incremented file index.
void updateFileName()
{
int i = 0;
for (; i < MAX_LOG_FILES; i++)
{
memset(logFileName, 0, strlen(logFileName)); // Clear logFileName string
// Set logFileName to "gpslogXX.csv":
sprintf(logFileName, "%s%d.%s", LOG_FILE_PREFIX, i, LOG_FILE_SUFFIX);
if (!SD.exists(logFileName)) // If a file doesn't exist
{
break; // Break out of this loop. We found our index
}
else // Otherwise:
{
SerialMonitor.print(logFileName);
SerialMonitor.println(" exists"); // Print a debug statement
}
}
SerialMonitor.print("File name: ");
SerialMonitor.println(logFileName); // Debug print the file name
}

```

```

void gpsTracker()
{
  if ((lastLog + 1000/LOG_RATE) <= millis())
  { // If it's been LOG_RATE milliseconds since the last log:
    if (tinyGPS.location.isValid()) // If the GPS data is valid
    {
      lastLog = millis(); // Update the lastLog variable
      if (writeSDCard()) // Log the GPS data
      {
        SerialMonitor.println("GPS logged."); // Print a debug message
      }
      else // If we failed to log GPS
      { // Print an error, don't update lastLog
        SerialMonitor.println("Failed to log new GPS data.");
      }
    }
    else // If GPS data isn't valid
    {
      // Print a debug message. Maybe we don't have enough satellites yet.
      SerialMonitor.print("No GPS data. Sats: ");
      SerialMonitor.println(tinyGPS.satellites.value());
    }
  }
  // If we're not logging, continue to "feed" the tinyGPS object:
  while (ssGPS.available())
  tinyGPS.encode(ssGPS.read());
}

```