

Sistema móvel de telemetria para automóveis

António Ricardo Teixeira Pessanha Carvalho

**Dissertação para obtenção do Grau de Mestre em
Engenharia Informática, Área de Especialização em
Tecnologias do Conhecimento e Decisão**

Orientador

Prof. Doutor Lino Manuel Baptista Figueiredo

Co-orientador

Eng.º Ricardo Manuel Soares Anacleto

Júri

Presidente

Prof. Doutor Luís Miguel Moreira Lino Ferreira

Vogais

Prof. Doutor Ângelo Manuel Rego e Silva Martins

Prof. Doutor Lino Manuel Baptista Figueiredo

Eng.º Ricardo Manuel Soares Anacleto

Dedico este trabalho à minha esposa Sabine e à minha querida filha Inês, que nunca me faltaram com apoio, carinho e compreensão.

Resumo

Este trabalho compreende uma análise crítica e reflexiva sobre o que atualmente existe no âmbito dos sistemas de navegação integrados (*e.g.* sistemas de posicionamento global com sistemas de navegação por inércia). O objetivo deste estudo vai também no sentido de desenvolver para um dispositivo móvel, um sistema de telemetria para automóveis baseado na plataforma Android fazendo uso de conceitos estudados e tecnologias existentes.

Pretende-se demonstrar a potencialidade da integração de um sistema de posicionamento por satélite, com um sistema de navegação por inércia, em dispositivos cada vez mais acessíveis ao utilizador comum, tais como *tablets*, *smartphones* e outros equipamentos dotados de dispositivos MEMS (sistemas microelectromecânicos) de baixo custo usando a plataforma Android. Este trabalho pretende explorar a forma como podemos fundir os registos dos vários sensores, tais como o acelerómetro, magnetómetro e giroscópio, para determinar a orientação do dispositivo e assim integrar esta informação de maior frequência, com a informação disponibilizada pelo GNSS de menor frequência, tendo como objetivo final a determinação em tempo real do posicionamento do dispositivo, destacando as forças e fraquezas de cada um dos sistemas de navegação.

Palavras-chave: GNSS, INS, filtro de Kalman, filtro complementar, Android, localização.

Abstract

This work includes a critical and reflective analysis about what exists in integrated navigation systems (*e.g.* Global Positioning Systems and Inertial Navigation Systems) area. This study will also focus in the development of a telemetry system for a mobile device, based on the Android platform, making use of existing technologies and concepts studied.

We intend to demonstrate the capability of integrating a global positioning satellite system (GNSS) with an inertial navigation system (INS), on mobile devices such as tablets, smartphones and other equipment with low-cost MEMS devices (microelectromechanical systems) using the Android platform. This work intends to explore the capability of merging data from various sensors, such as accelerometer, magnetometer and gyroscope, to determine the device's orientation and integrate more frequently this information with the data provided by the GNSS, where the ultimate aim is to determine in real-time the localization of the mobile device and highlighting the strengths and the weaknesses of each navigation systems.

Keywords: GNSS, INS, Kalman filter, complementary filter, Android, localization.

Agradecimentos

Gostaria de agradecer a todas as pessoas que de alguma forma contribuíram para a realização deste trabalho.

Agradeço à minha esposa, Sabine Pessanha pelo apoio que me deu nas horas difíceis, pela compreensão tida nos momentos alargados de ausência necessários ao estudo e sobretudo pela paciência demonstrada neste longo percurso.

À minha filha Inês Pessanha pelo carinho e alegria que sempre distribuiu nos momentos certos.

Aos meus orientadores Professor Doutor Lino Manuel Baptista Figueiredo e Engenheiro Ricardo Manuel Soares Anacleto pela disponibilidade, conselhos, orientação determinante e pelo grande trabalho de revisão efetuado ao presente trabalho.

Aos docentes da instituição pela dedicação e pela contribuição dada na evolução do meu conhecimento.

Aos meus pais Carmen Carvalho e António Carvalho pelo apoio e carinho dado ao longo da minha longa vida académica e por nunca terem desistido de acreditar em mim.

À minha restante família pela motivação constante e presença nos momentos de maior necessidade.

Aos poucos mas bons amigos pelo apoio e compreensão pela ausência e pouca dedicação que lhes dei.

A todos, o meu muito obrigado!

Índice

1	Introdução	1
1.1	Contexto e objetivos	2
1.2	Dispositivos móveis	3
1.3	Organização do trabalho	4
2	Estado da arte	7
2.1	<i>Global Navigation Satellite System</i> (GNSS)	7
2.1.1	<i>Global Positioning System</i> (GPS)	8
2.1.2	<i>Global Navigation Satellite System</i> (GLONASS)	9
2.1.3	GALILEO	9
2.1.4	Formato NMEA (<i>National Marine Electronics Association</i>)	10
2.2	Sistema de navegação por inércia ou sistema de navegação inercial	12
2.2.1	Sistemas de plataforma estabilizada	13
2.2.2	Sistemas do tipo <i>Strapdown</i>	14
2.3	Vantagens e desvantagens no contexto do uso isolado do GNSS e INS	14
2.4	Integração dos sistemas GNSS/INS	17
2.4.1	Filtro de Kalman	20
2.4.2	Filtro complementar	22
2.5	Estratégias para a integração de dados	23
2.6	Sistemas de navegação integrados para automóveis	26
2.6.1	Trusted vehicle navigator	29
2.6.2	Uninterrupted Portable Car Navigation System Using GPS, Map and Inertial Sensors Data	31
2.6.3	Vbox 3i	34
2.6.4	Ctrack	36
2.6.5	GNSS/INS Integration in Vehicular Urban Navigation	38
2.6.6	Improved Vehicle Positioning in Urban Environment through Integration of GPS and Low-Cost Inertial Sensors	41
3	Orientação e referenciais de navegação	45
3.1	Representação da orientação	45
3.2	Sistemas de coordenadas ou referenciais	48
3.2.1	Cálculo da orientação inicial	50
3.2.2	Determinação da aceleração e velocidade	50
4	Desenvolvimento de aplicações para dispositivos móveis	53
4.1	Plataforma Android	55
4.1.1	Arquitetura	55
4.1.2	Desenvolvimento para Android	57
4.1.3	Sensores nos dispositivos móveis	58
4.1.4	Deteção do movimento e orientação do dispositivo com Android	60

5	Aquisição e análise de dados dos sensores de um dispositivo móvel	63
5.1	Processamento de sinais e erros dos sensores.....	64
5.2	Análise do acelerómetro	65
5.2.1	Calibração do acelerómetro	66
5.2.2	Atenuação do ruído	71
5.3	Análise do giroscópio	77
5.4	Análise do magnetómetro.....	79
5.5	Filtro complementar de orientação	80
5.6	Cálculo da gravidade com o dispositivo em movimento.....	87
6	Desenvolvimento da solução	93
6.1	Restrições impostas ao desenvolvimento	94
6.2	Activities.....	95
6.3	Módulo INS	102
6.4	Resultados	105
6.4.1	Cenário 1.....	105
6.4.2	Cenário 2.....	106
6.4.3	Cenário 3.....	108
6.4.4	Cenário 4.....	109
7	Conclusão	111
	Referências.....	117

Lista de Figuras

Figura 1 - Algoritmo simplificado do sistema de navegação inercial numa plataforma estabilizada. (Woodman, 2007)	12
Figura 2 – IMU de plataforma estabilizada (Woodman, 2007).....	13
Figura 3 - Algoritmo simplificado para sistema de navegação inercial com plataforma <i>Strapdown</i> (Woodman, 2007).....	14
Figura 4 - Princípio básico do processamento com filtro de Kalman. (Adaptado de Brown, et al., 1997)	22
Figura 5 - Estratégia de integração INS/GPS <i>Loosely coupled</i> . (Adaptado de Petovello, et al., 2007)	24
Figura 6 - Estratégia de integração INS/GPS <i>Tightly coupled</i> . Adaptado de (Petovello, et al., 2007)	25
Figura 7 - Estratégia de integração INS/GPS <i>Deeply coupled</i>	26
Figura 8 - Aplicação e requisitos de performance dos acelerómetros actualmente tendo em conta a intensidade das acelerações medidas e a largura de banda. (Kraft, 2000)	27
Figura 9 - Estruturas e princípio de funcionamento de um acelerómetro MEMS capacitivo....	28
Figura 10 - <i>Yaw, Pitch</i> e <i>Roll</i> (giroscópio) e aceleração (acelerómetro) nos diversos eixos.	28
Figura 11 – Trusted Positioning Platform (TPP) (Positioning, 2012).....	30
Figura 12 - Algoritmo de deteção do modo de transporte. (Adaptado de Hunter, et al., 2010)31	
Figura 13 - Descrição conceitual das fontes de informação de navegação e o fluxo de dados do sistema portátil de navegação para automóvel. Os elementos de <i>hardware</i> são sombreados a verde. (Davidson, et al., 2009)	32
Figura 14 – Vbox 3i + IMU	34
Figura 15 – Esquema I/O da Vbox. (Racelogic, 2011)	35
Figura 16 - Mapeamento do posicionamento do veículo (GPS - Vermelho; GPS+INS - AZUL). (Racelogic, 2011).....	36
Figura 17 - Consumo energético Ctrack vs GPS vs Wifi. (Thiagarajan, et al., 2011)	37
Figura 18 - Arquitetura do sistema Ctrack. (Thiagarajan, et al., 2011).....	38
Figura 19 - Descrição conceptual das fontes de informação para navegação e fluxo de dados do sistema portátil de navegação automóvel. (Davidson, et al., 2009).....	42
Figura 20 - A matriz rotação R é utilizada para identificar a orientação atual relativamente à inicial.	46
Figura 21 - Ângulos <i>pitch, roll</i> e <i>yaw</i>	48
Figura 22 - 6 graus de liberdade (6DOF) num sistema de navegação inercial do tipo <i>Strapdown</i>	48
Figura 23 - Referencial local e global.	49
Figura 24 - Efeito da força centrípeta na direção do vetor velocidade.....	51
Figura 25 - Venda de Smartphones por sistema operativo no segundo trimestre de 2013. (Gartner, 2013).....	55
Figura 26 - Arquitetura Android. (Perez, s.d.)	56
Figura 27 - Ciclo de vida de uma atividade. (Adaptado de Open Source Project, 2012)	58

Figura 28 - Sistemas de coordenadas do Android. (Adaptado de Open Source Project, 2012)	61
Figura 29 - Definição dos eixos no Android.....	62
Figura 30 - Smartphone LG 2X P990 e suas características principais.	63
Figura 31 - Constatação do sinal dos eixos do <i>smartphone</i> colocado em cima de uma superfície plana para o primeiro cenário.	65
Figura 32 - Output da aceleração para os eixos assinalados como a) x, b) y e c) z do acelerómetro sem tratamento (análise temporal de 5 segundos).	65
Figura 33 - Comparativo da medição do sensor acelerómetro nos eixos assinalados como a) X, b) Y e c) Z antes da calibração (Raw) e pós calibração (Calibrado) para o cenário 1.	69
Figura 34 - Comparativo da medição do sensor acelerómetro nos eixos assinalados como a) X, b) Y e c) Z antes da calibração (Raw) e pós calibração (Calibrado) para o cenário 2.	70
Figura 35 - Comparativo da medição do sensor acelerómetro nos eixos assinalados como a) X, b) Y e c) Z antes da calibração (Raw) e pós calibração (Calibrado) para o cenário 3.	70
Figura 36 - Comportamento do sinal após aplicação da média móvel com 5 (verde) e 10 (vermelho) pontos nos eixos assinalados como a) X, b) Y e c) Z para o cenário 1.	72
Figura 37 - Registo da aceleração nos eixos assinalados como a) X, b) Y e c) Z e visualização dos mesmos registos com o filtro passa baixo com $\alpha = 0.2$	74
Figura 38 - Comparativo dos registos de aceleração no eixo X para o cenário 1 com o filtros média móvel (MM) a 5 e 10 pontos e o filtro passa baixo com $\alpha=0.2$	74
Figura 39 - Registo da aceleração nos eixos assinalados como a) X, b) Y e c) Z e visualização dos mesmos registos com o filtro passa baixo com $\alpha = 0.35$	75
Figura 40 - Registo da aceleração no eixo Y e visualização dos mesmos registos com o filtro passa baixo com $\alpha = 0.55$	76
Figura 41 - Cálculo da velocidade com a aceleração do eixo Y previamente calibrado e utilização de filtros Média Móvel e passa baixo.	77
Figura 42 - a) Velocidade angular ($^{\circ}/s$). b) angulo ($^{\circ}$) após integração.	78
Figura 43 - Output do magnetómetro nos eixos x, y e z.	79
Figura 44 - Fusão sensorial com filtro complementar.....	81
Figura 45 - Comparativo para a obtenção do <i>Yaw</i> com a utilização do método da API Android <i>getRotationMatrix(acc,mag)</i> e o método desenvolvido com recurso ao filtro complementar <i>getRotationMatrixFromOrientation(a,p,r)</i>	82
Figura 46 - Comparativo para a obtenção do <i>Pitch</i> com a utilização do método da API Android <i>getRotationMatrix(acc,mag)</i> e o método desenvolvido com recurso ao filtro complementar <i>getRotationMatrixFromOrientation(a,p,r)</i>	82
Figura 47 - Comparativo para a obtenção do <i>Roll</i> com a utilização do método da API Android <i>getRotationMatrix(acc,mag)</i> e o método desenvolvido com recurso ao filtro complementar <i>getRotationMatrixFromOrientation(a,p,r)</i>	83
Figura 48 - Comparativo na utilização do três métodos na determinação do YAW.	83
Figura 49 - Obtenção da aceleração dos eixos X, Y e Z no referencial global com o auxilio da matriz rotação obtida com o filtro complementar.	85
Figura 50 - Obtenção da aceleração dos eixos X, Y e Z no referencial global com o auxilio da matriz rotação obtida com o auxilio do sensor virtual <i>TYPE_ROTATION_VECTOR</i>	86

Figura 51 - Comparativo dos registos do acelerómetro linear do Android e do calculado com o auxílio do filtro complementar para o eixo X.....	87
Figura 52 - Comparativo dos registos do acelerómetro linear do Android e do calculado com o auxílio do filtro complementar para o eixo Y.....	87
Figura 53 - Comparativo dos registos do acelerómetro linear do Android e dos calculados com o auxílio do filtro complementar para o eixo Z.....	87
Figura 54 - Aceleração sentida no eixo Y durante o trajeto.....	89
Figura 55 - Registo da gravidade no eixo Y.	90
Figura 56 - Aceleração linear (sem efeito da gravidade) determinada pelo sensor virtual <i>TYPE_LINEAR_ACCELERATION</i> , aceleração menos a gravidade determinada pelo método 1 (giroscópio) e aceleração menos a gravidade determinada pelo método 2 (filtro passa alto). 91	91
Figura 57 - Registo da velocidade obtida do GPS, velocidade determinada pela aceleração total menos a gravidade calculada pelo método 1 e velocidade determinada pela aceleração total menos a gravidade calculada pelo método 2.....	91
Figura 58 - Dispositivo móvel utilizado para análise do sistema, fixo no tablier do automóvel.	94
Figura 59 - Principais <i>activities</i> da aplicação.....	95
Figura 60 - Ecrã inicial da aplicação.(1- botão para aceder ao ecrã de calibração; 2- botão para aceder às definições; 3- botão para arrancar o sistema).....	96
Figura 61 - Ecrã de calibração. (1- Valores atuais de calibração; 2- botão de saída; 3- Botão para continuar).....	96
Figura 62 - Processo de calibração. (1- barra de progresso; 2- Texto de apoio)	97
Figura 63 - Parâmetros da aplicação.....	98
Figura 64 – Caixa de diálogo enquanto a aplicação obtém sinal GPS.....	100
Figura 65 - Caixa de diálogo enquanto o INS inicializa.....	100
Figura 66 - Visualização do percurso efetuado. A vermelho o percurso estimado pelo INS e a azul o percurso obtido pelo GPS. O primeiro botão permite ao utilizador efetuar o log e envio de dados via UDP. O segundo botão permite aceder à segunda <i>view</i>	101
Figura 67 - A aplicação desenvolvida apresenta informações sobre a direcção e velocidade do INS e GPS. Com os dois botões em baixo é possível ao utilizador voltar atrás ou simular falha GPS para análise.....	101
Figura 68 - 4 fases envolvidas no sistema INS.....	102
Figura 69 - Resumo dos processos inerentes ao INS implementado.....	103
Figura 70 – Trajeto relativo ao cenário 1.	106
Figura 71 - Trajeto relativo ao cenário 2.....	107
Figura 72 - Trajeto relativo ao cenário 2 após colocação do dispositivo móvel no centro do veículo para limitar as interferências magnéticas.	108
Figura 73 - Trajeto relativo ao cenário 3.....	108
Figura 74 - Trajeto relativo ao cenário 4.....	109
Figura 75 - Trajeto relativo ao cenário 4.....	110

Lista de Tabelas

Tabela 1 – Tabela comparativa dos GNSS GPS, GALILEO e GLONASS (factos de 2011).....	8
Tabela 2 - Significado dos campos da mensagem do tipo RMC.....	11
Tabela 3 - Sensores utilizados no desenvolvimento da aplicação.	59
Tabela 4 - Estatística do output do acelerómetro nos diferentes eixos.	66
Tabela 5 - Orientações do dispositivo móvel durante o processo de calibração.....	67
Tabela 6 - Estatísticas das amostras do acelerómetro nas diferentes posições (1,2 e 3).	68
Tabela 7 - Estatísticas dos valores obtidos do acelerómetro pós filtragem com o filtro média móvel nas diferentes posições (1,2 e 3).....	72
Tabela 8 - Estatísticas dos valores obtidos do sensor magnetómetro.....	80

Acrónimos

ARW	<i>Angle Random Walk.</i>
ASCII	<i>American Standard Code for Information Interchange.</i>
BPSK	<i>Binary Phase Shift Keying.</i>
CDMA	<i>Code Division Multiple Access.</i>
DR	<i>Dead-reckoning.</i>
ECEF	<i>Earth Centered Earth Fixed.</i>
ECI	<i>Earth-centered Inertial.</i>
EKF	<i>Extended Kalman Filter.</i>
ENU	<i>East, North, Up.</i>
ERB	<i>Estação Rádio Base.</i>
GNSS	<i>Global Navigation Satellite System.</i>
GPS	<i>Global Positioning System.</i>
GSM	<i>Global System for Mobile Communications.</i>
IMU	<i>Inertial Measurement Unit.</i>
INS	<i>Inertial Navigation System.</i>
IRNSS	<i>Indian Regional Navigational Satellite System.</i>
MEMS	<i>Micro Electro Mechanical Systems.</i>
MHZ	<i>Megahertz.</i>
NMEA	<i>National Marine Electronics Association.</i>
PRN	<i>Pseudo Random Noise.</i>
UTC	<i>Tempo Universal Coordenado.</i>
VRW	<i>Velocity Random Walk.</i>
WGS-84	<i>World Geodetic System 1984.</i>

1 Introdução

A evolução tecnológica dos dispositivos móveis, impulsionada sobretudo pelos grandes fabricantes num ambiente extremamente competitivo com o intuito de comercializar equipamentos cada vez mais multifuncionais, tem aberto um novo mercado de aplicações para *smartphones* ou *tablets*.

A presença de recetores GNSS (*Global Navigation Satellite System*), ligação *Wi-fi*, acelerómetros (sensores de aceleração), giroscópios (sensores de rotação) e o aumento da capacidade de processamento, fazem com que estes aparelhos fiquem aptos a correr aplicações mais robustas, providenciando novas funcionalidades em aplicações de lazer e de entretenimento. Estes aparelhos requerem sistemas operacionais mais complexos, e com capacidade de gerir estes novos componentes.

Apesar dos dispositivos móveis, nomeadamente os *smartphones*, conterem cada vez mais dispositivos MEMS (*MicroElectroMechanical Systems*) tais como acelerómetros e giroscópios, a sua integração com sistemas de GNSS não é habitual, pois normalmente são usados para jogos ou aplicações de entretenimento que não requerem este tipo de integração. Isto acontece também porque os sensores presentes nos *smartphones*, são geralmente MEMS de baixo custo com características de erro complexas e difíceis de modelar (Georgy, 2010).

No desenvolvimento de soluções de navegação, o recetor GNSS presente nos *smartphones* de hoje em dia permite receções entre 1 a 10 hz (Schmidt & Phillips, 2011), dependendo do GPS interno ou externo, ou seja, uma a dez atualizações de posicionamento absoluto por segundo o que pode ser insuficiente, derivado a erros e desvios não assinalados na trajectória e quando pós-processados e disponibilizados num mapa digital. Estes erros deverão ser tratados recorrendo à integração com outros sensores. Outros obstáculos para o uso isolado de sistemas GNSS são os *blackouts* ou falhas e degradação de sinal GNSS provenientes de diversas fontes de interferência, ruídos e erros de *multipath* (Skog, 2007) (Schultz, 2006). Para resolver estes e outros problemas de degradação das soluções de navegação, recorre-se geralmente à informação de sensores disponíveis, tais como acelerómetros e giroscópios cuja configuração se dá pelo nome de IMU (*Inertial Measurement Unit*) que por sua vez é parte integrante de um INS (*Inertial Navigation System*). Dependendo da precisão, disponibilidade, continuidade e integridade que se pretende oferecer, diferentes filtros de fusão de dados poderão ser adotados, nomeadamente o filtro complementar (Higgins, 1975) (Neto, et al., 2009), o filtro de Kalman ou o EKF (*Extended Kalman Filter*) (Filho, 2007) (Skog, 2007) .

1.1 Contexto e objetivos

Embora a tecnologia GNSS, nos últimos anos, esteja a atravessar a melhor fase de desenvolvimento e seja disponibilizada num número cada vez maior de plataformas móveis, a maior desvantagem no seu uso, o bloqueio e degradação de sinal, vai continuar a existir, ainda que o sistema GLONASS Russo, o sistema GALILEO Europeu ou o sistema COMPASS Chinês (operacional mas ainda regional), fiquem totalmente operacionais.

A combinação do GNSS com o INS prova ser uma boa solução em momentos de bloqueio de sinal do GNSS ou como forma de compensação da reduzida taxa de leitura dos GNSS (Petovello, et al., 2007) .

O sistema integrado fornece de forma contínua, posição, velocidade e direção a uma taxa de leitura elevada. Estas vantagens têm vindo a conduzir à integração do GNSS com o INS em aplicações militares e civis, assim como o rápido progresso no desenvolvimento de sensores MEMS, fazendo com que a integração de sensores de inércia de tecnologia MEMS com GNSS seja especialmente atrativa devido ao tamanho reduzido, baixo custo e baixo consumo de energia que os mesmos providenciam.

A existência de recetores GNSS e sensores INS de baixo custo na maioria dos dispositivos móveis de hoje em dia, possibilita o desenho e desenvolvimento de sistemas de navegação integrados proveitosos para uso em aplicativos que necessitem de disponibilizar localização precisa, contínua e em tempo real em alternativa a uma solução, por exemplo, baseada na integração de leituras do GNSS e determinação contínua do trajeto recorrendo a um algoritmo *map-matching* (White, et al., 2000). A precisão da grande maioria dos sensores INS, presentes nestes dispositivos, ainda é bastante limitada, existindo uma grande acumulação de erros em soluções integradas se estes não forem alimentados com informação dos satélites por mais de alguns segundos.

O objetivo deste trabalho é o desenvolvimento de um sistema móvel para automóveis, baseado em métodos de integração das leituras dos sensores do INS (giroscópio, acelerómetro e magnetómetro) com as de um recetor GNSS presente numa vasta gama de dispositivos móveis tais como *smartphones* e *tablets*, permitindo registar num mapa os trajetos realizados, o tempo demorado e as variações registadas pelos sensores nos 3 eixos. Esta informação gerada deverá ainda ser armazenada de forma persistente para que o utilizador tenha a acesso à informação e assim poder efetuar uma leitura detalhada.

De forma mais detalhada, propõem-se então os seguintes objetivos:

- Analisar o uso de dispositivos MEMS em aplicações de navegação;
- Analisar, recorrendo a ferramentas de cálculo numérico, nomeadamente o MATLAB¹, os diferentes registos providenciados pela plataforma Android de um dispositivo móvel;

¹ MATLAB - O MATLAB é um programa de cálculo numérico que pode ser usado interactivamente. A sua estrutura de dados fundamental é a matriz, que pode ter elementos reais ou complexos. Embora na sua versão base o MATLAB já possua um vasto conjunto de funções de carácter genérico, existem várias bibliotecas de funções adicionais (designadas por *toolboxes*) que expandem as suas capacidades em domínios de aplicação mais específicos.

- Propor, implementar e testar algoritmos que permitem a compensação de erros de posicionamento gerados pelo sistema de navegação por inércia, auxiliado por um recetor GNSS (integração) garantindo um desempenho adequado nos períodos de ausência ou degradação do sinal GNSS;
- Adaptar o algoritmo para utilização em dispositivos móveis recorrendo à plataforma Android tendo em conta a necessidade de leitura e processamento em tempo real, as capacidades limitadas dos sensores presentes e a arquitetura do *hardware*;
- Providenciar uma interface do sistema ao utilizador que efetue a devida análise gráfica de forma intuitiva;
- Providenciar métodos de armazenamento da informação de forma persistente para posterior análise.

1.2 Dispositivos móveis

Dispositivos móveis ou sistemas computacionais móveis são termos genéricos que se referem a uma variedade de dispositivos que permitem aos utilizadores acederem a dados e informação local e global a partir de qualquer localização. Estes equipamentos são cada vez mais populares devido às inúmeras funcionalidades oferecidas, preço relativamente baixo, facilidade de transporte e reduzida dimensão.

Devido à grande variedade de dispositivos móveis existentes, não existe uma definição clara e unívoca sobre o que são, mas sim critérios estabelecidos que auxiliam na sua determinação. Os critérios que melhor definem um equipamento como sendo um dispositivo móvel poderão ser (Sandoval, et al., 2004):

- A capacidade de se ligar à Internet (ou outra rede de dados);
- Suporta Inputs do utilizador e interação;
- Oferece múltiplas funcionalidades;
- Leve, portátil e pequeno;
- Permitir aos programadores o desenvolvimento de aplicações nativas.

Dispositivos que encaixam nestes critérios podem ser os *smartphones*, *tablets* e *netbooks*.

Os dispositivos móveis, quando comparados com os computadores convencionais, ainda apresentam algumas limitações tais como a capacidade de memória, poder de processamento, tamanho do ecrã e bateria (Sandoval, et al., 2004).

Muitos dos dispositivos móveis estão atualmente equipados com camaras fotográficas, recetor GNSS, *Bluetooth*, acelerómetros, giroscópios e outros sensores. Devido à complexidade e rápida evolução, alguns *standards* para a computação móvel estão ainda em

1 Introdução

fase de especificação. Os sistemas operativos² que vêm instalados nos dispositivos móveis atuais, fornecem APIs (*Application Programming Interface*) que especificam padrões e rotinas por forma a disponibilizarem as suas funcionalidades aos programadores.

As funcionalidades de um dispositivo não são apenas atribuídas ao *hardware*. A discussão acerca destas deve ter em conta qual o sistema operativo a ser usado e que plataformas de desenvolvimento estão disponíveis no mercado. O equipamento é obviamente importante, mas as diferenças das funcionalidades disponibilizadas pelo *software* de um *smartphone* por exemplo, podem ser decisivas na escolha.

Alguns dos principais sistemas operativos para dispositivos móveis existentes atualmente no mercado são o Android, o BlackBerry OS, o iOS e o Windows Phone.

Resumidamente, existe um conjunto de fatores técnicos que são preponderantes para o sucesso do trabalho aqui desenvolvido. Em primeiro lugar, os sensores integrados com os dispositivos móveis tais como o acelerómetro e giroscópio, que fazem parte de uma lista mais alargada de sensores que hoje em dia estão disponíveis, nomeadamente o compasso digital, sensor de proximidade, sensor de luz, entre outros. Em segundo lugar, a capacidade de processamento e armazenamento disponível nos dispositivos móveis. Os processadores destes dispositivos são cada vez mais potentes tendencialmente *multi-core*. A capacidade de armazenamento é baseada em memória *flash*, recurso que será de grande importância para o armazenamento e posterior análise dos registos efectuados no âmbito deste trabalho. Em terceiro lugar o sistema operativo e linguagem de programação.

O desenvolvimento da aplicação de análise vai ser efetuado para dispositivos que correm o sistema operativo Android com recurso à linguagem de programação Java com a inclusão de bibliotecas específicas presentes na API de desenvolvimento para Android.

1.3 Organização do trabalho

Este documento está organizado em capítulos que permitirão ao leitor um entendimento gradual sobre os assuntos referidos.

O capítulo 1 visa contextualizar e apresentar os principais objetivos no desenvolvimento do presente trabalho. Pretende-se também introduzir alguns conceitos relacionados com dispositivos móveis.

O capítulo 2 introduz o estado da arte atual onde são empregues os conceitos necessários ao desenvolvimento deste trabalho. Inicialmente é efetuada uma introdução sobre os principais objetos de estudo tais como os sistemas de navegação, arquiteturas e técnicas de integração GNSS/INS. Em seguida é elaborada uma apresentação sobre sistemas e investigações mais relevantes neste campo onde são referidos os aspetos de integração não só sobre o GNSS/INS mas também sobre outras possibilidades de integração com vista a obtenção dos mesmos resultados.

² Um sistema operativo é um programa ou conjunto de programas cuja principal função é gerir os recursos de um sistema (gerir tarefas de processamento, gerir memória, etc) e fornecem uma interface entre o dispositivo e o utilizador.

O capítulo 3 é responsável pela introdução a conceitos de orientação e referenciais de navegação importantes para entender a complexidade da análise e desenvolvimento efetuados posteriormente. Os conceitos de orientação permitem definir as diferentes representações da mesma (local ou global). A introdução sobre os referenciais existentes vai permitir introduzir as formas de determinação da aceleração e velocidade requeridas pelo sistema INS.

O capítulo 4 introduz a plataforma usada no desenvolvimento (Android) da aplicação e caracteriza os principais componentes da mesma. São também especificadas as formas de obtenção dos dados relevantes para posterior processamento e integração.

O capítulo 5 especifica em detalhe os sinais provenientes dos diferentes sensores do dispositivo móvel e a forma como são tratados com vista a obtenção do objetivo final que passa por reduzir os erros característicos dos sensores. É ainda definida a forma de fusão sensorial através de um filtro complementar e estipulados diferentes métodos de obtenção do vetor gravidade com o dispositivo em movimento.

O capítulo 6 detalha as principais atividades no desenvolvimento da aplicação e os resultados finais. São especificados os principais interfaces gráficos, funcionalidades e resultados finais providenciados pela aplicação espelhados em cenários reais.

O capítulo 7 sintetiza os aspectos relevantes do presente trabalho e apresenta as conclusões finais deste estudo, as suas limitações e recomendações para trabalhos futuros.

2 Estado da arte

Neste capítulo, pretende-se abordar os aspetos mais importantes a serem considerados relativamente ao tema da dissertação, assim como conceitos necessários à compreensão e contextualização desta investigação e desenvolvimento. Os conceitos abordados são os sistemas de navegação (GNSS e INS), as suas vantagens, desvantagens e limitações, usados na análise e desenvolvimento do tema, as estratégias de integração conhecidas dos sistemas de navegação referidos, os filtros responsáveis pela fusão de dados (uma das fases de integração dos sistemas de navegação) e uma apresentação sobre algumas soluções que empreguem estes conceitos e abordagens em dispositivos móveis ou que permitam portabilidade.

2.1 *Global Navigation Satellite System (GNSS)*

GNSS é um termo genérico utilizado para denominar um sistema de posicionamento de escala global com recurso a satélites. Um GNSS é formado por uma constelação de satélites que enviam sinais de posicionamento e tempo permitindo identificar com uma certa precisão a longitude, latitude e orientação do dispositivo recetor. Existem vários sistemas GNSS em funcionamento ou em fase de desenvolvimento tais como o GPS, GLONASS e o GALILEO, este último próximo a entrar em funcionamento. Outros sistemas de localização por satélite estão também a ser desenvolvidos, tal como o COMPASS-beidu chinês, o IRNSS indiano e o QZSS japonês tendo sido projetados inicialmente como sistemas regionais de navegação (RNSS – Regional Navigation Satellite Systems), ou seja, cobrem apenas o seu território. Desta forma, nesta secção, destacam-se apenas os sistemas que foram projetados como sistemas globais de navegação por satélite.

Na tabela 1 podemos visualizar um comparativo entre os três sistemas (GPS, GLONASS e GALILEO), distinguindo as principais características, nomeadamente o número total de satélites (em funcionamento ou previstos), o número de planos orbitais de cada um dos sistemas, a altitude orbital dos satélites, o grau de inclinação relativamente ao equador dos satélites, as frequências das ondas portadoras, a modulação usada e o sistema de referência geodésico³.

³ A definição da posição de um ponto requer o estabelecimento de um sistema de referência. Sobre esse sistema de referência, podem ser definidos diversos tipos de coordenadas sobre os quais se efetuam os cálculos envolvendo posições, distâncias, desníveis e ângulos. (Matos, 2007)

Tabela 1 – Tabela comparativa dos GNSS GPS, GALILEO e GLONASS (factos de 2011).

	GPS	GLONASS	GALILEO
Satélites	24+3	21+3	27+3
Planos orbitais	6	3	3
Altitude Orbital	20200	19100	23616
Inclinação	55°	64.8°	56°
Frequência	1575.42 Mhz 1227.6 Mhz	1246-1257 Mhz 1602-1616 Mhz	1164-1300 Mhz 1559-1591 Mhz
Modulação	CDMA	FDMA	CDMA
Sistema de referência geodésico	WGS-84	PE-90	GTRF

Neste tipo de sistemas de navegação, cada satélite transmite repetidamente um sinal com informação sobre a sua posição orbital e qual o instante temporal em que o sinal foi emitido. O recetor irá calcular as distâncias aparentes entre o recetor e o satélite, com base na diferença entre o tempo de chegada (medido com o relógio do recetor) e o tempo de emissão do sinal (medido com o relógio do emissor). A posição é então obtida através da trilateração⁴, realizada ao estabelecer a geometria do problema com base nas posições dos satélites e as distâncias entre eles e o recetor.

2.1.1 *Global Positioning System (GPS)*

O GPS é um sistema de posicionamento global, relativo ou absoluto, utilizável em qualquer ponto da Terra ou nas suas imediações atmosféricas. Atualmente, e cada vez mais, é utilizado nos mais diversos domínios, tais como em sistemas de navegação aéreo e terrestre, cartografia, geodesia, topografia, georeferenciamento, agricultura de precisão, controlo de movimento de placas tectónicas entre outros (Sousa, 2004) (Monico, 2000).

O GPS tem como principal objetivo transmitir informações de tempo e distância permitindo navegação de precisão (posição, velocidade e direção) contínua e global, em tempo real e sob quaisquer condições atmosféricas.

O sistema GPS é composto por uma rede de 27 satélites, 24 operacionais + 3 extra, para o caso de avaria de algum dos primeiros, distribuídos em 6 planos orbitais. Comparando o tempo em que os sinais foram transmitidos dos satélites e o tempo que demoraram a ser recebidos, o recetor GPS calcula a distância a cada satélite, sendo no mínimo processada a distância de três ou mais satélites (depende do recetor GPS), que resultará na sua posição na superfície da terra. Quanto mais satélites, mais preciso fica o cálculo do posicionamento geográfico.

Com estas distâncias medidas, o recetor também poderá calcular a velocidade média em determinado trajeto, distância percorrida, altitude do local determinado para além de outros parâmetros.

⁴ Trilateração no cálculo do posicionamento consiste na intersecção de 3 esferas definidas pela distância de 3 satélites. Esta intersecção gera 2 pontos no espaço e 1 deles é descartado por estar localizado no espaço sideral, muito distante da terra.

Os satélites GPS transmitem sobre a forma de duas portadoras em banda L⁵, L1= 1575,42 MHz e L2= 1227,6 MHz. Estas portadoras são moduladas, em BPSK⁶ (*Binary Phase Shift Keying*), por um sinal resultante da combinação de códigos pseudoaleatórios (PRN - *Pseudo Random Noise*) com uma outra sequência conhecida como Mensagem de Navegação (*Data Signal*) (Sousa, 2005) (Kumar, 2004). Na modernização da rede de satélites GPS, está incluída a utilização de uma terceira frequência L5 de 1176,45 MHz para uso civil.

2.1.2 Global Navigation Satellite System (GLONASS)

O sistema GLONASS foi criado na década de 80 do século XX pela União Soviética com características semelhantes ao GPS. O sistema GLONASS foi projetado para ter 24 satélites incluindo três de reserva distribuídos em 3 planos orbitais com 8 satélites por plano. O GLONASS esteve perto da sua configuração final por volta da metade da década de 90, do século XX, mas por questões económicas e de manutenção, nem todos os satélites ficaram a funcionar.

Em 2002 a Rússia possuía apenas 8 satélites operacionais. No entanto, e após a recuperação económica, um total de 14 satélites estavam já operacionais no final do ano 2005 com o lançamento de satélites aperfeiçoados tais como o GLONASS-M e o GLONASS-k com esperança média de vida de 7 e 10 anos respetivamente. Em Outubro de 2011, o sistema estava totalmente operacional oferecendo cobertura global.

O sistema GLONASS disponibiliza serviços de navegação diferenciados tanto para uso civil como para uso militar. Os satélites do GLONASS também transmitem dois sinais na banda L (L1 e L2) com o mesmo nível de potência do GPS, e a banda de frequência é aproximadamente de 20 a 30 MHz maior que a do GPS.

O intuito dos responsáveis pelo sistema de navegação por satélite GLONASS é que este venha a ser interoperável com outros sistemas GNSS, nomeadamente o GALILEO. Para isso deverá existir uma alteração das características do sinal do GLONASS forçando ao uso da técnica de divisão por código (CDMA)⁷ em detrimento da divisão de frequência (FDMA)⁸ (Assis, 2010).

2.1.3 GALILEO

O GALILEO é o sistema Europeu de navegação por satélites que está em construção prevendo-se a entrada em operação no final de 2013. Prevê-se que o sistema seja interoperável com os outros dois sistemas de navegação globais (GPS e GLONASS).

⁵ Grupo de rádio-frequências que se estende de 390MHz ate 1550MHz.

⁶ Neste caso (modulação por sinal binário) os dois estados da portadora modulada diferenciam-se porque assumem, consoante um ou outro valor, do sinal modulador, 2 valores diferentes para a fase.

⁷ CDMA (Code Division Multiple Access) é um sistema de acesso múltiplo a um determinado recurso para permitir o acesso simultâneo, e sem interferências, de um grande número de utilizadores, a uma única faixa de radiofrequências, através da atribuição de códigos de utilização diferentes a cada utilizador.

⁸ FDMA (Frequency Division Multiple Access) é um sistema de acesso múltiplo a um determinado recurso, para permitir o acesso simultâneo, e sem interferências, de um grande número de utilizadores, a uma única faixa de radiofrequências, através da atribuição de frequências diferentes a cada um deles.

Este sistema tem como vantagem possuir uma constelação de 30 satélites incluindo 3 de reserva possibilitando assim uma maior cobertura do globo incluindo as zonas polares. Para além desta vantagem, a disponibilidade do serviço é garantida (redundância) e nas circunstâncias mais extremas, informará o utilizador a ocorrência de falha num dos satélites em poucos segundos. Estas características tornarão o Galileo adequado a aplicações onde a segurança é crucial, tais como na circulação de comboios ou aterragem de aeronaves.

O primeiro satélite experimental do GALILEO, GIOVE-A foi lançado a 28 de dezembro de 2005, com o objetivo de consolidar tecnologias críticas já desenvolvidas. O segundo satélite (GIOVE-B) foi lançado a 27 de abril de 2008. O GIOVE-B emprega um relógio ultra estável e preciso que pode ser a grande promessa no âmbito de sistemas GNSS futuros (ESA, 2008).

Todos os satélites serão lançados em órbita a 26616 Km (acima das órbitas dos satélites GPS e GLONASS) dispostos ao longo de 3 planos orbitais.

O sistema disponibiliza a mesma variedade de informações que o sistema americano modulada em 3 portadoras de frequências L1, L2 e L3 (1.575,42 MHz, 1.227,60 MHz e 1.381,05 MHz respetivamente). O Galileo envia informações de uso civil nas duas primeiras frequências, de forma redundante, as quais podem ser combinadas para o refinamento da posição calculada.

2.1.4 Formato NMEA (*National Marine Electronics Association*)

A interface com os recetores GNSS é normalmente efetuada de dois modos: no modo binário ou usando mensagens no formato NMEA 0183. O formato de dados normalizado NMEA 0183 é amplamente utilizado por um conjunto de equipamentos, conferindo assim interoperabilidade entre sistemas de navegação. Este formato foi proposto pela *National Marine Electronics Association* dos Estados Unidos em 1983 e embora fosse concebido inicialmente para uso em instrumentação marítima, este formato tornou-se na interface normalizada para os recetores GNSS.

São definidos na norma NMEA especificações do sinal, especificações do protocolo de transmissão de dados e um formato de mensagens específico. As mensagens são constituídas por até um máximo de 82 caracteres ASCII. Todas as mensagens NMEA são formatadas em frases que começam pelo carácter ASCII “\$” e terminam com os caracteres ASCII “ <CR> <LF> ”. Um endereço de 5 caracteres ocorre logo após o carácter “\$” onde os dois primeiros caracteres distinguem o emissor da mensagem (GP – GPS, GL – GLONASS) e os três últimos distinguem o formato e tipo de mensagem. Após estes primeiros campos, estão os dados propriamente ditos e que são separados pelo carácter “,”. A mensagem termina quando for encontrado o carácter “*” que é seguido de dois dígitos hexadecimais usados para validação da mensagem (checksum). Podemos verificar isto através de um exemplo demonstrado a seguir:

```
$GP<identificador da mensagem><dados><*checksum><CR><LF>
```

Os tipos de mensagem definidas (GSA, GSV, GLL, GGA, RMC e outros) surgem de acordo com necessidades específicas.

Para o desenvolvimento deste trabalho, o interesse vai para o tipo de mensagem RMC (*Recommended Minimum sentence C*) proveniente do sistema GPS, pois atende às especificações e parâmetros adotados. Apesar de tudo, no desenvolvimento para Android, esta preocupação de efetuar o tratamento da mensagem não existe, pois a classe utilizada para efetuar o desmembramento dos valores já é parte nativa do SDK. No entanto é interessante analisar a que tipo de informação temos acesso. Os campos presentes neste tipo de mensagem, apresentam toda a informação necessária para posterior integração com um INS. A tabela 2, apresenta o significado de cada um dos campos presentes neste tipo de mensagem com o formato especificado a seguir:

\$GPRMC,<1>,<2>,<3>,<4>,<5>,<6>,<7>,<8>,<9>,<10>,<11>*<12><13>.

Tabela 2 - Significado dos campos da mensagem do tipo RMC.

Campo	Nome	Descrição
1	Tempo UTC	Hora UTC no formato hhmmss.sss
2	Estado	A= <i>Valid Position</i> V= <i>Navigation receiver warning</i>
3	Latitude	Formato: dd mm.mmmm
4	Indicador Norte / Sul	N= Norte S= Sul
5	Longitude	Formato: ddd mm.mmmm
6	Indicador Este / Oeste	E= Este W= Oeste
7	Velocidade	Formato: Nós
8	Orientação	Formato: Graus
9	Data	Dados UTC no formato dd mm yy
10	Variação magnética	Em graus ou em direcção
11	Modo	A = Modo Autónomo D = Modo Diferencial E = Modo Estimado M = Modo de Entrada Manual N = Dados Inválidos S = Modo de Simulação
12	Checksum	Código de erro da mensagem, no caso de ocorrer
13	Fim de mensagem	Indica o fim da mensagem

Esta norma permite ainda que os fabricantes de *hardware* definam as suas mensagens proprietárias. Todas estas mensagens deverão começar pela letra “P” seguida de três caracteres identificadores do fabricante que controla a mensagem.

2.2 Sistema de navegação por inércia ou sistema de navegação inercial

O funcionamento de um INS é baseado nas leis do movimento. Se o campo gravítico⁹ através da trajetória for conhecido, então a posição pode ser determinada através da dupla integração da aceleração e da velocidade respetivamente (figura 1) (Cardeira, 2009) (Sousa, 2004). A primeira integração da aceleração fornece a velocidade e a segunda integração fornece o deslocamento, mas de qualquer forma as condições iniciais de integração têm de ser conhecidas. Por isso, e ao contrário do sistema GNSS, o INS é um sistema de posicionamento relativo, ou seja, a posição é determinada em relação a outro ponto cujas coordenadas devem ser conhecidas.

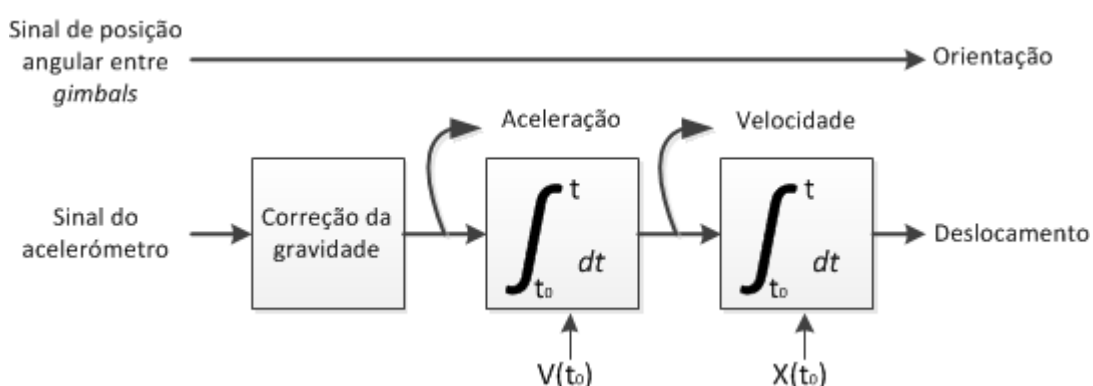


Figura 1 - Algoritmo simplificado do sistema de navegação inercial numa plataforma estabilizada. (Woodman, 2007)

Por outro lado, a informação relativa à gravidade pode ser extraída se a posição e a velocidade do veículo forem conhecidas.

O fundamento da navegação inercial baseia-se, portanto, na integração de acelerações observadas através da montagem de acelerómetros num veículo, avião, robot ou pessoa. O ângulo e distância são usados para obter a posição atual. A frequência de *output* de dados neste tipo de sistema é consideravelmente maior do que num sistema GNSS genérico.

Para uma melhor compreensão dos fenómenos físicos por detrás dos sistemas de navegação, é útil ter presente as leis de Newton. A 1ª Lei de Newton, também conhecida por Lei da Inércia, afirma que um objeto em repouso, tende a ficar em repouso e que um objeto em movimento uniforme tende a permanecer em movimento uniforme se a resultante das forças externas aplicadas for nula. Já a 2ª Lei de Newton afirma que num referencial inercial a aceleração a de uma massa m é proporcional à resultante das forças aplicadas F_i conforme a equação (1) (Cardeira, 2009) (Figueiredo, et al., 2007).

$$F_i = ma \quad (1)$$

⁹ Força gravítica que atua por unidade de massa, colocada num ponto, à distância R da massa criadora do campo.

2.2 Sistema de navegação por inércia ou sistema de navegação inercial

Dispondo de um sensor que tenha a capacidade de medir a aceleração, seria possível calcular a variação de velocidade e posição através de sucessivas integrações matemáticas em relação ao tempo. A aceleração pode ser medida com dispositivos denominados por acelerómetros que normalmente são instalados nos sistemas de navegação inercial tendo a capacidade de medir acelerações segundo um eixo (X,Y ou Z). Mecanicamente, são instalados para que os seus eixos sensíveis sejam mutuamente ortogonais.

Para se conseguir navegar em relação a um referencial inercial, é fundamental conhecer as direções segundo as quais, os eixos dos acelerómetros estão a fazer as medidas de aceleração. Os sensores que permitem medir os movimentos de rotação de um corpo, são designados por giroscópios, e permitem obter a orientação dos acelerómetros em todos os instantes. Assim é possível transformar as medidas de aceleração para o referencial em relação ao qual se está a navegar e posteriormente integrar em relação ao tempo, uma e duas vezes para se obter a velocidade e posição, respetivamente (Cardeira, 2009).

Os IMU são tipicamente constituídos por um giroscópio triaxial e um acelerómetro triaxial medindo a velocidade angular e a aceleração linear respetivamente. Quase todos os tipos de IMU são classificados dentro de duas grandes classes: sistemas de plataforma giro-estabilizada (*gimbal systems*) esquematizado na figura 2, onde os sensores estão montados numa plataforma que mantém a mesma orientação independentemente da dinâmica a que o objeto está sujeito e em sistemas onde os acelerómetros e giroscópios estão fixos na estrutura cuja dinâmica se pretende descrever (*strapdown systems*) (Woodman, 2007).

2.2.1 Sistemas de plataforma estabilizada

Na configuração de plataforma giro-estabilizada, os sensores inerciais estão montados numa plataforma que está isolada de qualquer movimento de rotação. Ligado a esta plataforma estão os braços (*gimbals*) que garantem liberdade nos três eixos de rotação. Estes sinais são enviados para os motores que rodam os braços de forma a cancelar a rotação, mantendo a plataforma alinhada com a estrutura principal. Para seguir a orientação do dispositivo, os ângulos entre cada braço adjacente podem ser medidos utilizando detetores angulares. Para calcular a posição do dispositivo é necessário proceder à dupla integração dos sinais dos acelerómetros montados na plataforma.

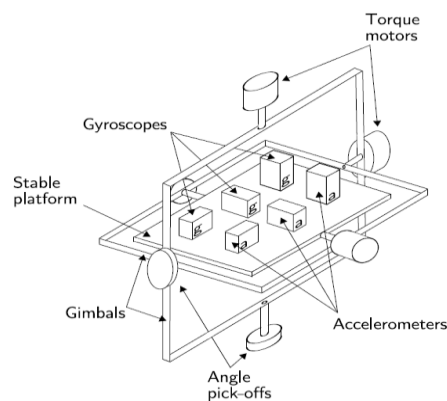


Figura 2 – IMU de plataforma estabilizada (Woodman, 2007).

2.2.2 Sistemas do tipo *Strapdown*

Um sistema de navegação do tipo *strapdown* faz uso de uma IMU que não está isolada das rotações do veículo onde foi instalada.

Em configurações deste tipo, os acelerómetros e giroscópios estão fixos na estrutura que está sujeita ao movimento, e assim, à saída dos sensores, temos a informação relativa ao movimento da estrutura, onde os sensores estão fixados. Para efeitos de navegação é necessário transformar estes sinais em coordenadas globais, através da integração da velocidade angular devolvida pelo giroscópio e posteriormente integrar os sinais provenientes dos acelerómetros. O esquema deste algoritmo pode ser observado na figura 3.

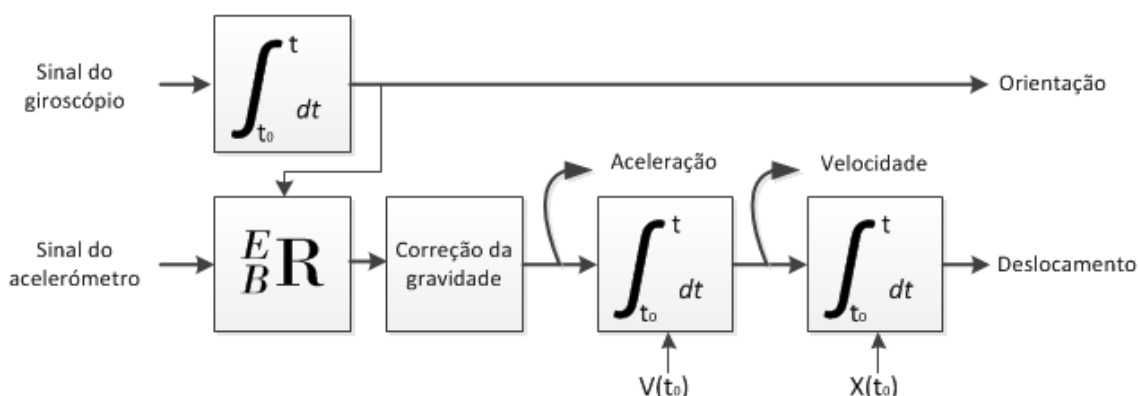


Figura 3 - Algoritmo simplificado para sistema de navegação inercial com plataforma *Strapdown* (Woodman, 2007).

2.3 Vantagens e desvantagens no contexto do uso isolado do GNSS e INS

Um sistema de navegação integrado INS/GNSS consiste na combinação de dois sistemas diferentes, unidos pelo algoritmo de processamento de dados. Tal combinação permite juntar as características positivas de cada um e compensar as características negativas. Desta forma convém salientar os aspetos positivos e negativos de cada um.

A utilização de sistemas GNSS apresenta várias vantagens tais como (Monico, 2000):

- Funciona em qualquer condição climática;
- Não necessita de intervisibilidade entre os pontos (GPS diferencial¹⁰);
- Alcance ilimitado;
- Funciona a qualquer hora do dia.

¹⁰ O GPS Diferencial (ou DGPS) é um sistema que permite aumentar a precisão dos dados e assim diminuir a margem de erro. Nesta solução existe um posto fixo de localização conhecida. Este corrige os erros enviando um sinal aos recetores e está constantemente a corrigir os dados recebidos por este. Para tal é necessária uma antena DGPS associada ao recetor GPS.

2.3 Vantagens e desvantagens no contexto do uso isolado do GNSS e INS

Um GNSS funciona bem no registo de trajetos de longa duração e na obtenção com elevada precisão da velocidade e posição. No entanto existem algumas desvantagens na utilização deste sistema de forma isolada (Schmidt, 2011):

- Necessita de perfeita visibilidade entre os satélites acima da linha do horizonte;
- Necessita no mínimo de 3 ou 4 satélites visíveis (dependendo do recetor);
- Existem interferências com o sinal (interferência ionosférica¹¹, troposférica¹²);
- Dificuldades de funcionamento em zonas muito arborizadas e edificadas ou onde o sinal não consiga chegar de forma desobstruída;
- Não autónomo;
- Multipath¹³;
- Baixa taxa de refrescamento.

Existem também vantagens e desvantagens no uso de sistemas INS que são descritas abaixo (Sousa, 2004) (Cardeira, 2009).

- As indicações de posição e velocidade são instantâneas e contínuas, permitindo a sua utilização em sistemas caracterizados por dinâmicas com grande largura de banda;
- É completamente autónoma uma vez que se baseia em medidas de aceleração linear e angular, que são obtidas no interior do próprio veículo. É um sistema que não irradia ondas eletromagnéticas e não está sujeito a sabotagem exterior;
- A informação de navegação pode ser obtida para todas as latitudes, incluindo as regiões polares, independentemente das condições meteorológicas e sem necessidade de estações terrestres;
- Um sistema inercial disponibiliza informação de velocidade e posição em relação a um referencial terrestre assim como a orientação do veículo em relação ao referencial terrestre local.

Tendo em conta que irá ser usado um INS baseado em leituras de um acelerómetro, giroscópio e magnetómetro de tecnologia MEMS, as maiores desvantagens deste sistema apresentam-se sobre a forma das características dos erros destes equipamentos.

Relativamente aos giroscópios, as principais origens de erros compreendem (Woodman, 2007):

¹¹ Interferências ionosféricas são anomalias que causam alterações na amplitude e na fase das ondas de rádio.

¹² A refração troposférica afeta a modulação do código e a fase com a mesma intensidade, causando uma diminuição na velocidade de propagação dos sinais.

¹³ O erro de *multipath* resulta na reflexão dos sinais em certos tipos de superfícies, nomeadamente água e metais.

- Erro de tendência (offset) ou erro sistemático: Um erro tendencioso sistemático ε , quando integrado, causa um erro angular que cresce linearmente com o tempo. $\theta(t) = \int_0^t \varepsilon \cdot dt = \varepsilon \cdot t$. Este erro pode ser estimado procedendo a uma média ponderada da saída do giroscópio de um longo intervalo de tempo e subtrair esse valor pelo valor de saída do giroscópio;
- Ruído branco: Os giroscópios têm saídas perturbadas por ruído branco¹⁴ causados por eventos termomecânicos gerando variações de frequência muito superiores à taxa de amostragem. A integração deste ruído introduz um erro cujo desvio padrão é $\sigma_\omega(t) = \sigma\sqrt{\delta t} \cdot t$ crescendo proporcionalmente com \sqrt{t} . Este erro é descrito pelo fabricante como ARW (*angle random walk*). Suponha-se que obtemos uma amostra a 1Hz e cujo ARW tenha um coeficiente de 3,5, logo podemos achar o valor do desvio padrão em graus deduzindo $\sigma_\omega|\Delta t = 1\text{sec} = 3.5 \text{ }^\circ/\sqrt{\text{hr}}/\sqrt{1\text{sec}} = 0.058 \text{ }^\circ/\text{sec}$;
- *Flicker noise*: Todos os dispositivos integrados provocam alterações na tensão de saída do sensor variando o erro sistemático proporcionalmente inverso à frequência (1/f). Geralmente vem especificado em $^\circ/\text{h}$ ou $^\circ/\text{s}$ em dispositivos menos precisos;
- Efeitos da temperatura: Flutuações de temperatura modificam o erro sistemático que é proporcional à raiz quadrada do tempo;
- Erros de calibração: Estes erros caracterizam-se por erros de escala que são apenas possíveis de identificar quando sujeito a uma rotação.

As características dos erros dos acelerómetros são as seguintes (Woodman, 2007):

- Erro de tendência (offset) ou erro sistemático: É um erro que se mantém constante em termos de magnitude ao longo de todas as observações. O erro da posição é fruto da dupla integração que cresce com o quadrado do tempo. $p(t) = \iint_0^t \varepsilon \cdot dt \cdot dt = \varepsilon \cdot \frac{t^2}{2}$;
- Ruído branco: As observações de um acelerómetro são perturbadas por um ruído branco. Este ruído introduz um erro no cálculo da velocidade cujo desvio padrão $\sigma_s(t) \approx \sigma \cdot t^{\frac{3}{2}} \cdot \sqrt{\frac{\delta t}{3}}$ cresce proporcionalmente com $t^{\frac{3}{2}}$;
- *Flicker noise*: Este ruído introduz um *Random walk*¹⁵ de segunda ordem cuja incerteza aumenta proporcionalmente com $t^{\frac{3}{2}}$ no cálculo da velocidade e um *Random Walk* de terceira ordem cuja incerteza aumenta proporcionalmente com $t^{\frac{5}{2}}$ no cálculo da posição;
- Efeitos da temperatura: Diferenças de temperatura causam alterações no sinal de *output*. Esta relação não é linear e depende do dispositivo usado. Causa erros de posição que crescem com o quadrado do tempo;

¹⁴ Ruído branco representa a sequência de variáveis aleatórias não correlacionadas de média zero.

¹⁵ Fomalização que traduz a ideia intuitiva da tomada de vários passos consecutivos, cada qual numa direção aleatória.

- Erros de calibração: Os erros de calibração (escala e alinhamento) apenas são visíveis quando o dispositivo está sujeito a uma aceleração. No entanto, devido ao efeito da gravidade, estes podem ser visíveis.

As características dos erros dos magnetómetros são as seguintes (Woodman, 2007):

- Declinação magnética: A declinação magnética é a diferença em graus, entre o norte magnético e o norte geográfico. Dependendo da região em que se está a navegar, erros graves podem surgir na determinação do *heading* (direcção ou rumo). Este problema pode ser mitigado com a obtenção dos valores corretos para o local.
- Distorção magnética: A existência de materiais que geram campos magnéticos que se sobrepõem ao campo magnético da Terra, perturbam a leitura. São materiais ferromagnéticos com propriedades magnéticas permanentes. Esta distorção, também é provocada pela existência de equipamento eléctrico, nomeadamente a existência de cablagem eléctrica na proximidade do dispositivo de leitura, motores eléctricos ou qualquer outro dispositivo que gere campos magnéticos corrompendo a leitura obtida pelo magnetómetro.
- Erros de escala: A sensibilidade nestes dispositivos não é perfeita na obtenção da leitura do campo magnético tendo como consequência leituras diferentes quando sujeitos ao mesmo campo magnético.

Para se determinar com maior exatidão o posicionamento, uma solução técnica apropriada e económica, é fundir os registos de sistemas de navegação por inércia baseados em tecnologia MEMS com os do sistema GNSS.

2.4 Integração dos sistemas GNSS/INS

Se um sistema de navegação utiliza mais do que um sensor simultaneamente, então este sistema faz uso de métodos de navegação integrada.

No caso das aplicações de GNSS, com o uso de um acelerómetro, permitimos que o *software* detete arranques, travagens e curvas mais rapidamente. Estas variáveis podem ser usadas para tornar o *software* de navegação mais inteligente, detetando quando se perdeu uma curva ou mantendo uma estimativa aproximada da localização quando se perde o sinal dos satélites por alguns segundos.

Uma possível aplicação é a determinação da posição entre duas estimativas da posição dadas pelo GNSS, as quais podem estar espaçadas por um intervalo de tempo relativamente longo (*e.g.*, em túneis).

Ao processo de estimar a posição atual com base no conhecimento de uma posição anterior, velocidade, tempo decorrido e percurso, chama-se DR (*Dead Reckoning*¹⁶). Este é o processo inerente aos sistemas INS, processo que produz estimativas.

Neste trabalho pretende-se usar um acelerómetro, um giroscópio e um magnetómetro, presentes já na grande maioria dos dispositivos móveis, para medir distâncias e determinar orientações e assim proporcionar uma integração com o sistema GNSS, neste caso o GPS. O grande problema no uso de sistemas INS, prende-se com a diferença entre as estimativas e a realidade da posição, a qual tem efeitos cumulativos. Estes erros cumulativos são derivados das características dos erros (referidas na secção 2.3) dos acelerómetros e giroscópios que por sua vez baseiam o cálculo da posição atual no conhecimento da posição anterior. Para colmatar esta falha dever-se-á providenciar uma recalibração periódica, neste caso efetuada pelo GPS (INS auxiliado por GPS). A esta fase dá-se o nome de inicialização do sistema INS que será responsável por duas operações:

- A atribuição de coordenadas e velocidade iniciais;
- A determinação da orientação da plataforma (dispositivo móvel) relativamente a um referencial de navegação.

Os desvios dos acelerómetros e giroscópios acumuláveis e que se deterioram com o tempo e outros erros sistemáticos em conjunto com os erros de orientação, podem ser minimamente determinados por um filtro de Kalman¹⁷. A diferença entre as estratégias para a integração de dados de GNSS e INS num único sistema, reside no tipo de informação que é dividida entre as unidades ou módulos individuais. Os tipos mais comuns de estratégias de integração encontradas na literatura, são chamados de *Loosely coupled*, *Tightly coupled* e *Deeply coupled* (Schmidt, 2011) (Petovello, *et al.*, 2007) (Filho, 2007).

A fusão de dados proveniente de diversos sensores pode ser vista como um processo de integração de dados de sensores distintos para a estimativa de parâmetros e estados necessários para a auto-localização, mapeamento, planeamento e controlo da trajetória de um veículo (Kam, *et al.*, 1997).

Os sensores são usados para medir quantidades físicas e o seu erro é dado pela diferença entre o valor medido (*pseudo-range*) e o valor real (*true-range*). Já o GNSS permite uma medição mais exata tendo em conta que não existem erros cumulativos, mas mesmo este está sujeito a problemas conforme especificado na secção 2.3.

O objetivo deste trabalho consiste no uso de tecnologia MEMS presente nos telemóveis de hoje em dia, em detrimento das tecnologias convencionais dos sensores de inércia. Os convencionais são geralmente mais caros e mais sensíveis apresentando melhores resultados, mas o objetivo é precisamente o uso de sensores de baixo custo presentes em equipamentos móveis.

¹⁶ Expressão derivada de "Deduced Reckoning" (Contagem Deduzida). Procedimento para determinação da localização atual de um veículo através da análise das informações sobre sua velocidade e curso conhecido. (Skog, 2007)

¹⁷ Tem o propósito de utilizar medições de grandezas realizadas ao longo do tempo (contaminadas com ruído e outras incertezas) e gerar resultados que tendam a se aproximar dos valores reais das grandezas medidas e valores associados.

Para a obtenção de observações dignas de serem usadas na integração com o GPS, as unidades de medida de inércia (IMU) disponibilizam métodos de filtragem das respectivas observações para atenuação ou correção de erros. Os filtros neste tipo de observações vão atenuar os ruídos e perturbações de alta frequência.

Para essa filtragem inicial e/ou como parte do processo de integração, usam-se geralmente os seguintes filtros:

- Filtro média móvel, filtro média ponderada ou passa baixo (*low pass filter*) e média centrada para atenuação de ruído e diminuição das perturbações de alta frequência.
- Filtro de Kalman e *Extended Kalman filter* (EKF) (Welch & Bishop, 2006) (Skog, 2007);
- Filtro complementar (Cardeira, 2009) (Higgins, 1975);
- Filtro de Occam (Lima, 2005);
- Filtro de partículas (Giremus & Tournet, 2006).

O filtro de Kalman é uma técnica de predição que atua sobre sistemas dinâmicos discretos ao longo do tempo determinando o estado verdadeiro do processo num sistema com ruídos. O estado atual do sistema é representado por um vetor de números reais, sendo que, a cada incremento do tempo, um operador linear é aplicado para determinar o próximo estado do sistema (Welch & Bishop, 2006). O EKF é uma evolução do modelo de Kalman, que visa a utilização de equações (processo ou medição) não lineares, ao contrário do modelo de Kalman original (J. Sum, 1996).

O termo “filtro complementar” é normalmente utilizado como referência a um qualquer algoritmo que tenha como objetivo a fusão de dados redundantes ou similares de diferentes sensores para alcançar uma estimativa ótima de uma determinada variável (*e.g.* giroscópio e acelerómetro). Num sistema típico de duas leituras de dados, uma delas providencia informação com ruído de alta frequência, que será filtrada através de um filtro passa-baixo. A outra produzirá dados com ruído de baixa frequência, sendo, portanto, filtrado através de um filtro passa-alto. Se estes filtros são matematicamente complementares, então o sinal filtrado será a reconstrução completa da variável que está a ser monitorizada, eliminando os ruídos.

O filtro de Occam baseia-se na aplicação de algoritmos de compressão com perda de detalhes sobre sinais contendo ruídos, onde a perda é igual à intensidade do ruído, logo a perda de detalhes e o ruído tendem a serem cancelados ao invés de serem somados (Natarajan, 1994). A aplicação do filtro de Occam sobre as observações de um sistema de navegação inercial procura diminuir o nível de ruído presente nestas observações antes do processo de integração.

Os filtros de partículas são uma forma de implementação dos filtros *bayesianos* recursivos. Podem ser aplicados a quaisquer modelos de transição de estado e de medida, não estando sujeitos a restrições de linearidade e/ou ruído gaussiano (Ristic, 2004).

Os filtros mais usados para o tipo de integração aqui discutido, baseado em sistemas de navegação distintos, são os filtros de partículas e de kalman (Petovello, et al., 2007).

Tendo em conta o nível de complexidade e capacidade computacional, neste trabalho será usado o filtro complementar para obtenção da orientação. Em termos comparativos, podemos descrever a forma de uso do filtro complementar e do filtro de Kalman. O uso dos filtros média móvel são descritos no capítulo 5 referente ao tratamento de sinais.

2.4.1 Filtro de Kalman

Após ter sido formulado por Rudolf. E. Kalman, há mais de 40 anos, diversas variações deste algoritmo têm sido empregues com sucesso em várias áreas da ciência e em muitas aplicações de engenharia. O filtro de Kalman é um algoritmo considerado ótimo no sentido de fornecer variância mínima na estimação de variáveis de estado de um sistema de posicionamento. O termo filtro deve-se ao facto de se ter difundido rapidamente em aplicações de engenharia que condicionam (filtram) sinais eletrónicos que estejam sujeitos a interferências ou a degradações. De facto trata-se de um algoritmo para processar medidas discretas (*inputs*) e estimar valores ótimos (*outputs*)

As duas principais características da solução Kalman são a possibilidade de modelação vetorial dos processos aleatórios em consideração e o processamento recursivo das medições com ruído (*inputs*). O facto de ser um algoritmo recursivo torna o filtro de Kalman, do ponto de vista computacional, bastante eficiente porque toda a informação necessária para a estimação do estado do sistema numa determinada iteração está contida na última estimativa.

Importa agora fazer referência ao princípio de Kalman. O filtro de Kalman é um algoritmo eficiente, capaz de estimar as variáveis de estado de sistemas lineares a partir de medidas ruidosas dos sensores que compõem o sistema. Por esta razão o filtro de Kalman é bastante utilizado na suavização de *outputs* de equipamentos de medição e mesmo na fusão de diferentes fontes de dados para sistemas de navegação, para além de outros propósitos.

O objetivo do filtro de Kalman num sistema dinâmico é estimar, a cada instante e de forma ótima, a saída do sistema. Após uma fase de inicialização, o filtro alterna entre uma fase de previsão e uma fase de correção de uma forma cíclica.

Todas as variáveis que o filtro de Kalman utiliza para estimar uma solução ótima, estão agrupadas num vetor x , chamado vetor de estado conforme a equação (2) (note-se o T de transposta pois é uma matriz $(n \times 1)$).

$$x = [x_1 \ x_2 \ x_3 \ x_4 \ \dots \ x_n]^T \quad (2)$$

O vetor de estado x possui não só os parâmetros desejados mas também aqueles que foram adicionados ao filtro para introduzir redundância ao problema.

O vetor de medidas z (observações) é um agrupamento de todas as variáveis cuja medida está disponível.

$$z = [z_1 \ z_2 \ z_3 \ z_4 \ \dots \ z_l]^T \quad (3)$$

É frequente que a dimensão (n) do vetor de estados x seja maior que a dimensão (l) do vetor de medidas z . Ambos os vetores estão relacionados linearmente de forma que a partir do vetor de estados, calcula-se o vetor de variáveis.

Antes de expor as equações do filtro de Kalman, convém descrever o propósito das várias fases que o compõem. Imagine-se um dispositivo móvel fixo numa plataforma (e.g. um automóvel) munido de um sistema de medição da posição, a deslocar-se a uma velocidade constante s (para simplificar o problema), num ambiente ruidoso. Poder-se-ia definir um modelo para descrever a posição conforme a equação (4):

$$x_k = x_{k-1} + s \cdot \Delta t + \omega_k, \quad (4)$$

onde x_k é dependente da posição anterior x_{k-1} , da velocidade s e do ruído ω_k que se considera branco, Gaussiano¹⁸, de média zero e com variância σ_ω^2 . Supondo ainda que o modelo que melhor descreve a relação entre as medidas efetuadas z_k e a posição atual do dispositivo é o mostrado na equação (5):

$$z_k = x_k + v_k, \quad (5)$$

onde v_k é representativo dos valores aleatórios das características do ruído da medida (observação) com as mesmas características de ω_k mas com variância σ_v^2 .

A fase de inicialização é efetuada com base nas condições iniciais, estabelecendo-se a posição inicial do automóvel, \hat{x}_0 , cuja incerteza é caracterizada pela sua variância σ_0^2 .

A fase de previsão é responsável pela estimativa da posição do automóvel no instante $k = 1$, antes de se obter as medidas do dispositivo. Assim sendo, prevê-se que o automóvel estará na posição \hat{x}_1^- , sendo esta uma estimativa do estado *a priori* que se calcula recorrendo à equação (6):

$$\hat{x}_1^- = \hat{x}_0 + s \cdot \Delta t + 0.4 \quad (6)$$

Assume-se que o ruído ω_k é 0.4 para o instante $k = 1$. Pode-se também calcular a incerteza relativamente à posição estimada através da equação (7):

$$\sigma_1^2 = \sigma_0^2 + \sigma_w^2 \quad (7)$$

Usando este modelo de sistema foi possível atualizar no tempo a posição estimada com base no estado anterior $k - 1$ existindo um aumento da incerteza proporcional ao número de previsões sucessivas, ignorando as medições disponibilizadas pelo dispositivo. Para diminuir a incerteza, recorre-se às medições.

A fase de correção, com base na previsão anterior \hat{x}_1^- e na medição z_1 efetuada pelo dispositivo, vai estimar a posição do automóvel \hat{x}_1^+ no instante $k = 1$, imediatamente a seguir à medição. Para determinar esta posição (\hat{x}_1^+), é efetuada uma média ponderada entre a posição estimada \hat{x}_1^- e a medição z_1 conforme equação (8).

$$\hat{x}_1^+ = \hat{x}_1^- + K(z_1 - \hat{x}_1^-). \quad (8)$$

¹⁸ O ruído branco gaussiano é um ruído simultaneamente branco (aquele em que não há uma correlação com o tempo) e gaussiano (que representa uma distribuição de Gauss) combinando as características de ambos.

O ganho de Kalman define a importância a ser aplicada à medição efetuada, pois depende das variâncias do erro da medição e estimativa feita na fase de previsão. Os processos aqui discutidos podem ser visualizados no resumo do algoritmo de Kalman na figura 4.

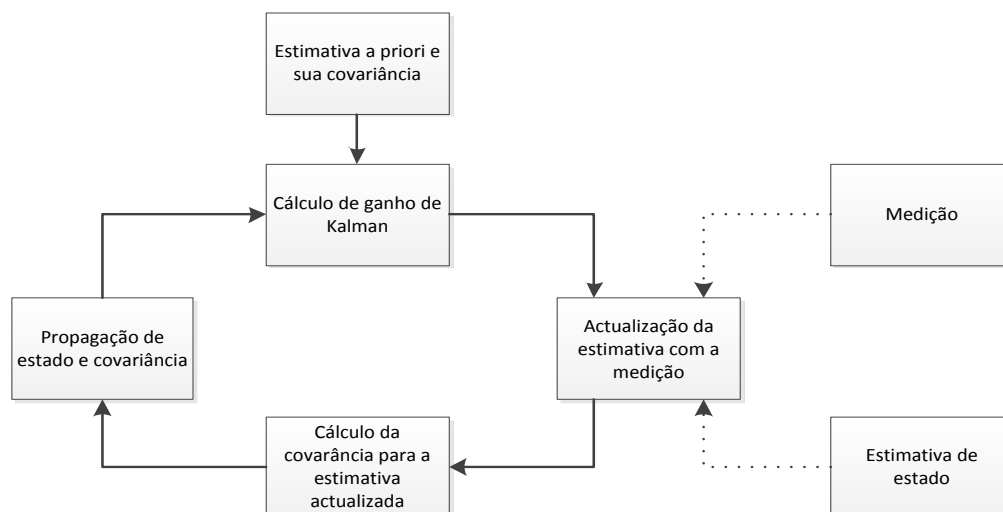


Figura 4 - Princípio básico do processamento com filtro de Kalman. (Adaptado de Brown, et al., 1997)

2.4.2 Filtro complementar

Na maioria dos casos analisados, os sistemas baseados em navegação inercial realizam a fusão dos registos dos diversos sensores utilizando o filtro de Kalman. A principal dificuldade encontrada na utilização de tal método é o processo complexo na definição e obtenção de bons modelos para os sensores assim como a capacidade computacional exigida na manipulação de matrizes (Neto, et al., 2009).

Tendo em conta estes aspetos é conhecida na literatura uma outra técnica menos complexa que dá pelo nome de filtro complementar (Colton, 2007) (Higgins, 1975).

Este tipo de filtragem permite integrar dois sinais suficientemente confiáveis nas suas características mais precisas, com diferentes faixas de frequência para se formarem valores mais precisos no domínio do tempo. Existem diversas vantagens na utilização deste método que passam pela baixa complexidade na implementação e a qualidade dos resultados obtidos, no nosso caso para a obtenção da orientação.

Ao fundir registos inerciais provenientes de giroscópios e acelerómetros com os registos provenientes do sensor magnetómetro, o filtro produz estimativas precisas de orientação que são centrais nos objetivos deste trabalho.

Imagine-se que x e y são observações com ruído de um qualquer sinal z , e \hat{z} é a estimativa produzida pelo filtro. Assume-se que o ruído de y é maioritariamente de alta frequência e o ruído de x é na sua grande maioria de baixa frequência. Então a função $G(s)$ pode destacar-se como um filtro passa baixo por forma a filtrar o ruído de alta frequência de y e $[1 - G(s)]$ o complemento, *i.e.*, um filtro passa alto que filtra o ruído de baixa frequência de x .

No caso de observações sem ruído ou sem erro a equação (9) representaria uma estimativa perfeita.

$$\hat{z} = z[1 - G(s)] + zG(s) = z \quad (9)$$

2.5 Estratégias para a integração de dados

Em termos computacionais é importante definir o grau de conhecimento e relacionamento entre os diferentes componentes de um sistema. O termo “arquitetura de integração” aqui empregue, pretende somente definir as técnicas de integração entre sistemas de navegação.

Os sistemas ou arquiteturas de integração atuais, geralmente são classificados como *Loosely coupled*, *Tightly Coupled* e *Deeply Coupled* (Schmidt, 2011) e pretendem descrever a forma como os dados que chegam de um sistema, neste caso INS e GPS, são fundidos num sistema de navegação integrado.

Esta terminologia tem variado no seu significado ao longo do tempo e por vezes existe alguma confusão sobre a sua classificação correta. Hoje em dia as arquiteturas de integração INS/GPS contêm elementos de diversos esquemas de integração o que torna uma tarefa difícil classificá-los como apenas *Loosely coupled*, *Tightly Coupled* e *Deeply Coupled* (Lachapelle & Petovello, 2007).

As características dos erros gerados nos sistemas INS e GPS são complementares. Quando os dados são fundidos, as estimativas do GPS (posição e velocidade) são usados para calibrar o INS. Por sua vez o INS disponibiliza estimativas com uma grande frequência (direção, posição e velocidade) necessária ao controlo e orientação do veículo.

Na estratégia *Loosely Coupled*, os sistemas INS e GPS operam como sistemas de navegação independentes. A informação proveniente dos dispositivos é então fundida usando um *software* e formando assim uma terceira solução de navegação. Normalmente são usados filtros de Kalman estendidos (EKF) para levar a cabo o trabalho de fusão de registos, apesar de atualmente serem já usados outros mecanismos não lineares tais como filtro de partículas (Giremus & Tourneret, 2006).

O propósito de uma integração *loosely coupled* é extrair os atributos desejáveis do INS e do GPS e suprimir os atributos indesejáveis. Existem algumas variações deste tipo de integração que são referidas como configurações *feed-forward* e *feedback*. Em configurações do tipo *feedback* os erros estimados pelo EKF são realimentados para corrigir os dados dos giroscópios e acelerómetros (mostrado na linha descontínua da figura 5). Quando este *feedback* não existe, então a configuração é do género *feed-forward*. Na figura 5, no caso de configuração *feed-forward*, seria então retirada a linha que mostra a realimentação. Quando dispositivos INS de baixa qualidade são utilizados (os quais apresentam maior taxa de erros), quase sempre a configuração *feedback* é usada. A configuração *feed-forward* é mais usada com dispositivos INS de grande qualidade e fiabilidade.

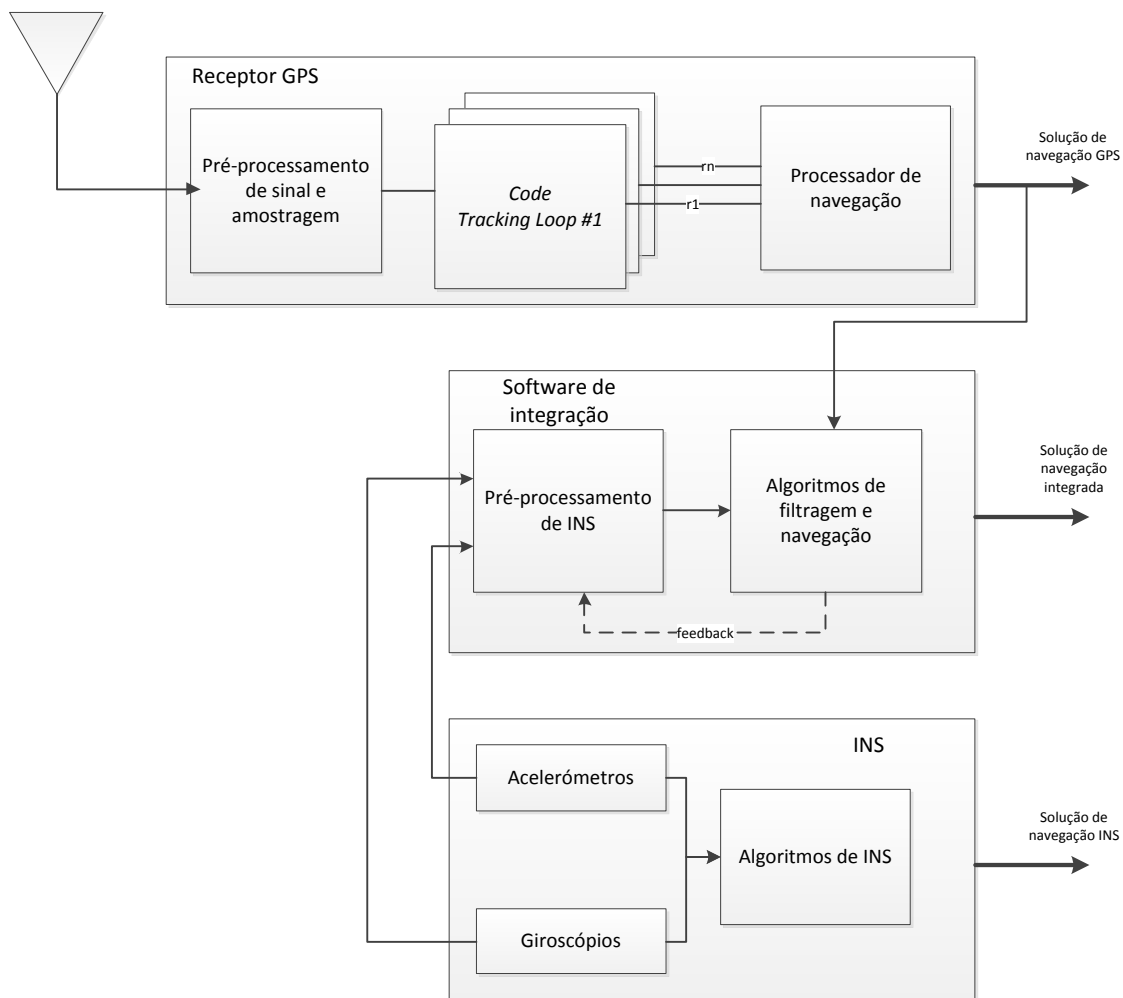


Figura 5 - Estratégia de integração INS/GPS *Loosely coupled*. (Adaptado de Petovello, et al., 2007)

Na estratégia *Tightly coupled*, cujo esquema está representado na figura 6, o INS e GPS estão reduzidos às suas funções sensoriais, ou seja, as observações dos acelerómetros e giroscópios assim como as pseudo-distâncias são usadas para gerar uma solução integrada de navegação.

De uma forma geral, o uso de uma solução do tipo *Loosely coupled* é menos precisa que uma solução do tipo *Tightly coupled*. Outra vantagem da integração do tipo *Tightly coupled* é o facto de poder-se extrair informação útil do recetor GPS mesmo em situações onde menos de quatro satélites são visíveis. Por outro lado, *Loosely coupled* tem a vantagem de redundância, pois tanto o GPS como o INS são soluções independentes de navegação.

Uma definição alternativa de *Tightly coupled* compreende as mesmas características da definição clássica referida atrás, acrescentando a particularidade de existir realimentação no sistema GPS com a informação proveniente do filtro de fusão, aumentando assim a performance do GPS.

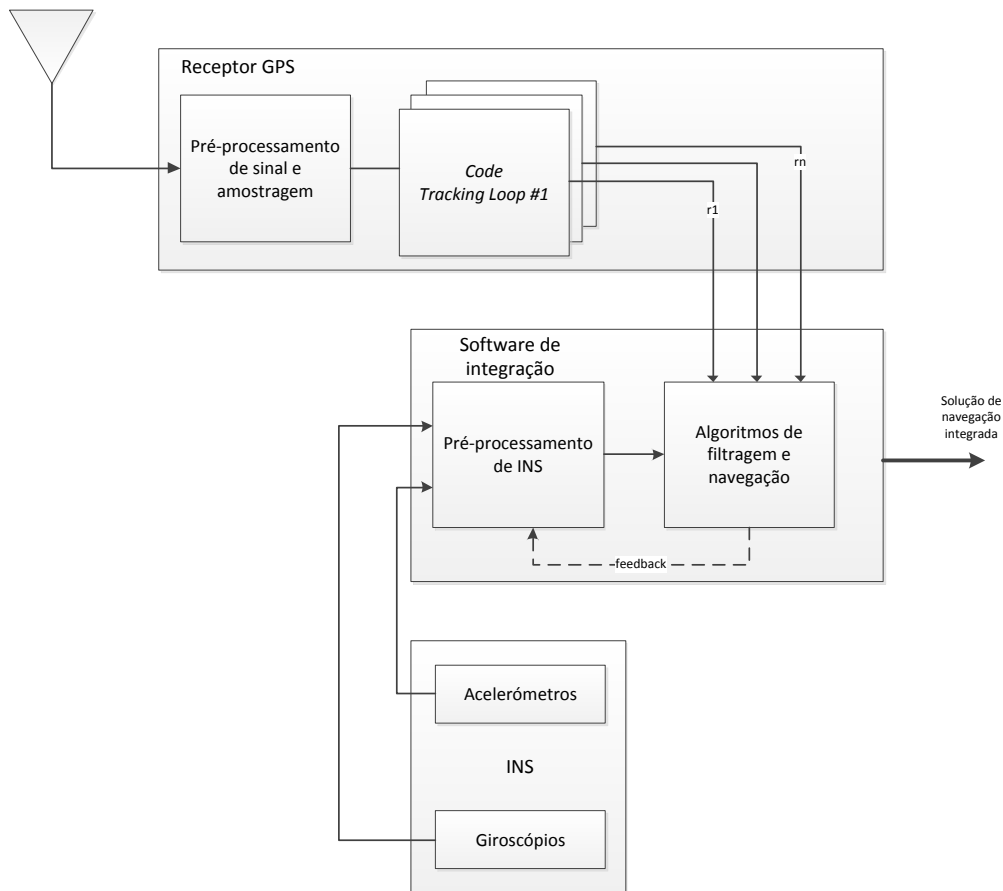


Figura 6 - Estratégia de integração INS/GPS *Tightly coupled*. Adaptado de (Petovello, et al., 2007)

Outra estratégia utilizada como arquitetura de integração dos sistemas INS e GPS é a *Deeply coupled*. Existem duas importantes características que a distinguem das integrações referidas anteriormente: a tradicional arquitetura do recetor constituída por *tracking-loops*¹⁹ é substituída por algo semelhante a um VDLL – *vector delay-locked loop*²⁰ (que pode ser visto na figura 7 dentro de uma caixa com a linha descontínua) e a utilização do INS ou informação proveniente deste, constitui parte integral do recetor GPS, ou seja, o GPS não pode ser visto como uma parte independente do INS.

Uma das vantagens deste sistema é o aumento da robustez quando há falhas por interferência ou dinâmica do veículo. Por outro lado, a desvantagem óbvia desta forma de integração é a extrema complexidade na integração da informação do GPS e INS dentro do recetor.

¹⁹ Um dos objetivos do recetor é sincronizar uma réplica do código PRN local com o sinal a receber de forma a calcular a distância percorrida do sinal a receber da forma mais precisa possível. Os *tracking loops* são usados em ciclos fechados para de forma contínua acompanhar o código e os parâmetros do sinal a receber.

²⁰ DLL é um circuito digital similar a um PLL (phase locked loop), cuja principal diferença é a ausência de um oscilador de tensão controlada substituído por uma linha de atraso.

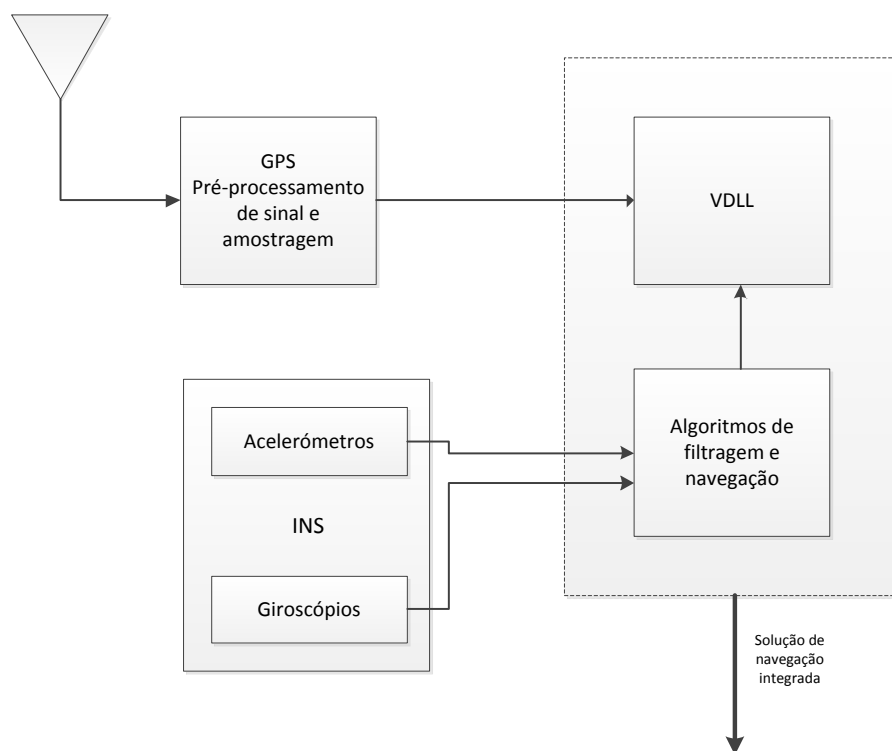


Figura 7 - Estratégia de integração INS/GPS Deeply coupled.

Resumidamente, indo de uma estratégia de integração do tipo *Loosely coupled* para *Deeply coupled*, vamos ganhando robustez no que toca a interrupções provenientes de interferência ou dinâmica do veículo (falhas de posicionamento pois a taxa de leitura de GPS é muito baixa). No entanto, com o ganho desta robustez, vem a perda de simplicidade, redundância e independência dos sistemas INS e GPS.

2.6 Sistemas de navegação integrados para automóveis

Antigamente, a tecnologia e a aplicação de acelerómetros estava condicionada a áreas onde o custo era pouco relevante, tais como as áreas aeroespaciais e militares. Hoje em dia, com a evolução da tecnologia, nomeadamente MEMS, os acelerómetros com sensibilidade tri-axial (medição ao longo de 3 eixos) estão reduzidos em muitos casos a apenas um componente de dimensões extremamente reduzidas.

A tecnologia MEMS explora, por exemplo, as propriedades mecânicas do silicó para criar estruturas móveis que no caso dos sensores MEMS, detetam movimento (aceleração e vibração) em direções distintas (Figueiredo, et al., 2007) (Kraft, 2000). Os requisitos dos sensores inerciais variam de forma significativa conforme os fins para os quais são destinados. A figura 8 mostra a performance necessária para cada tipo de aplicação. Normalmente os sistemas que necessitam de maior performance são aqueles que são destinados a aplicações militares pois combinam a tecnologia GPS com a alta performance dos sistemas de medição

inercial, referências temporais precisas e comunicações radiofrequência para se obter uma localização e orientação precisa.

Um acelerómetro linear de tecnologia MEMS é baseado numa estrutura em silicone, com interdigitações e em forma de pente composta por dentes fixos e móveis (figura 9). A aceleração é obtida pela medição dos deslocamentos de elementos móveis que estão associados aos eixos. O movimento medido pelo sensor é então convertido num sinal analógico ou digital. As suas aplicações são imensas e vão desde plataformas de estabilidade para video-cameras, realidade virtual, monitorização de embates, brinquedos eletrónicos, *air-bags*, entre outros cujos requisitos de performance variam de acordo com as suas aplicações (figura 8).

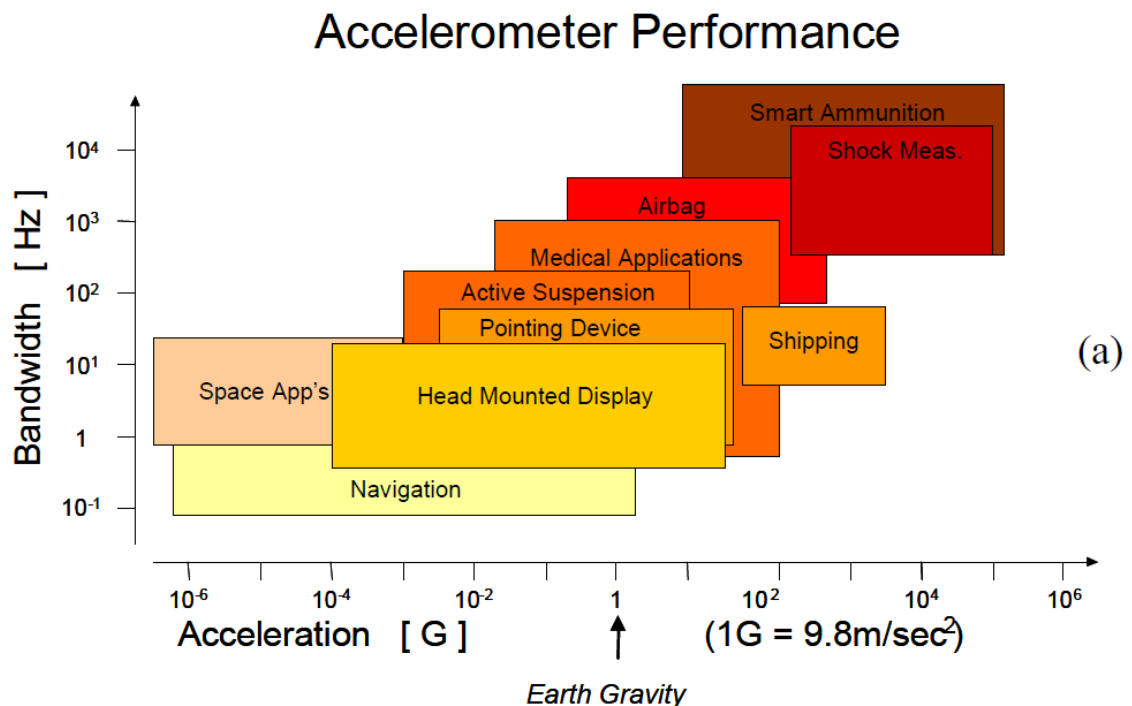


Figura 8 - Aplicação e requisitos de performance dos acelerómetros actualmente tendo em conta a intensidade das acelerações medidas e a largura de banda. (Kraft, 2000)

Um giroscópio baseado em tecnologia MEMS é um dispositivo que permite medir a velocidade de rotação de um objeto em torno do centro de rotação ou dito de outra forma, a velocidade a que um objeto gira em relação a um sistema de referência inercial conhecido como velocidade angular. A velocidade angular pode ser integrada em relação ao tempo para determinar a posição angular do objeto. Dependendo da orientação de montagem, a rotação é medida em relação a um dos três eixos principais da plataforma: *yaw*, *pitch* ou *roll*²¹ (figura 10).

²¹ Pitch, Roll e Yaw, são termos usados geralmente na aeronáutica representam os movimentos de rotação nos eixos transversal, longitudinal e vertical respetivamente.

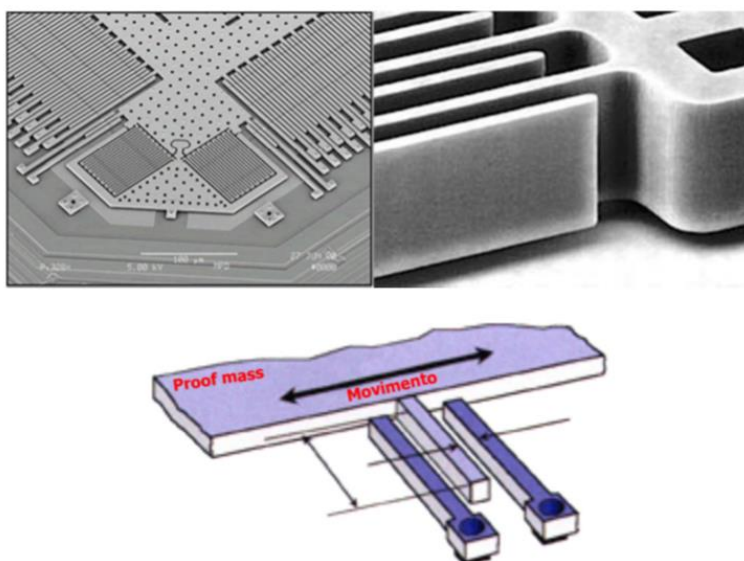


Figura 9 - Estruturas e princípio de funcionamento de um acelerómetro MEMS capacitivo.

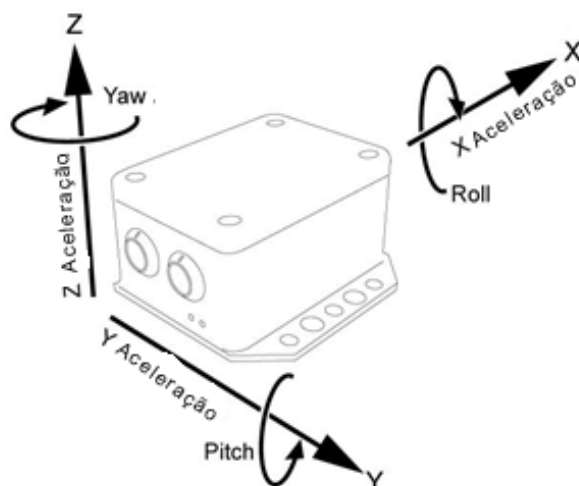


Figura 10 - Yaw, Pitch e Roll (giroscópio) e aceleração (acelerómetro) nos diversos eixos.

Neste contexto, será elaborada uma introdução sobre alguns trabalhos e equipamentos existentes dentro do âmbito dos objetivos deste trabalho.

Existem diversos sistemas portáteis de localização GPS mas muitos destes falham na integração com dispositivos contínuos de localização tais como acelerómetros e/ou giroscópios de forma a disponibilizar localização em zonas de ausência ou interferência de sinal GPS.

Este estudo tem em conta não só os desenvolvimentos para a plataforma Android (também objeto de desenvolvimento neste trabalho), mas também para outras plataformas que façam uso de métodos equivalentes de integração de GPS + INS. Pretende-se também demonstrar o uso da tecnologia não só no caso aqui retratado (sistemas móveis para veículos terrestres) mas também noutros domínios distintos que façam uso integrado.

2.6.1 Trusted vehicle navigator

O sistema *Trusted vehicle navigator* (Syed, 2009) é um sistema disponibilizado pela empresa *Trusted Positioning inc.* fornecendo um módulo de navegação portátil e métodos de operação distintos.

A empresa tem por missão resolver o problema com os sistemas de navegação e posicionamento atuais: a disponibilidade de navegação e posicionamento contínuo, preciso e de baixo custo para ambientes *indoor* e *outdoor* tanto para empresas como para consumidores finais.

A plataforma oferecida pela empresa, integra sensores tais como os sensores de inércia MEMS (externos ou presentes em dispositivos móveis) com atualizações disponibilizadas por (GNSS ou WiFi) para providenciar de forma contínua, precisa e de baixo custo, navegação e posicionamento em todos os ambientes.

Tem como características o seguinte:

- *Software* de integração multissensorial;
- Módulos personalizados para necessidades específicas;
- Recalibração autónoma em dispositivos MEMS de baixo custo;
- Suporte para outros diferentes tipos de sensores tais como magnetómetros, barómetros e sensores de velocidade;
- Filtros não lineares avançados capazes de operações em tempo-real em processadores de baixo desempenho.

A portabilidade do sistema desenvolvido pela empresa é uma das principais características da solução. Resumidamente faz uso dos dados de sensores de baixo custo (acelerómetros, magnetómetros, barómetros) independentes ou presentes em dispositivos móveis e de dados provenientes de *GNSS* e *WiFi* para gerar informação e integrá-la através de uma plataforma a que dão o nome de TPP (*Trusted Positioning Platform*) tal como exemplificado na figura 11.

É ainda possível integrarem módulos OBDII (*OnBoard Diagnostics*) dos veículos para obterem parâmetros tais como velocidade.

O sistema é composto por um recetor GPS que envia um primeiro conjunto de informações de navegação e por um sensor de inércia (ou outro tipo de sensor auxiliar) para gerar um segundo conjunto de informações de navegação. Esta informação de navegação é usada por um processador dedicado de modo a identificar o “modo de transporte / trânsito” e orientação. A informação do modo de transporte (“a pé”, “no veículo”, “indeterminado” e “a pé e no veículo”) e orientação assim como a informação de navegação, produzem uma solução de navegação filtrada onde resulta a posição, velocidade e direção.

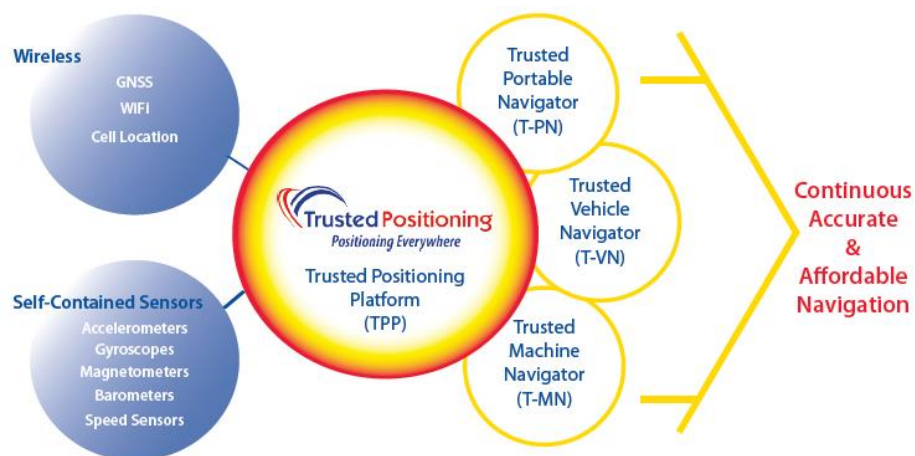


Figura 11 – Trusted Positioning Platform (TPP) (Positioning, 2012).

Um filtro de Kalman estendido é usado para combinar a informação redundante onde o vetor de estado do sensor de inércia é regularmente atualizado com as observações do GPS.

Outro algoritmo importante na solução é o da deteção de tipo de transporte exemplificado na figura 12 que permite então determinar que algoritmos de tratamento serão utilizados.

Para detetar o modo de transporte ou transito, as informações do acelerómetro e giroscópio são obtidas. Para além disto, a solução permite a verificação de um magnetómetro cuja informação é repetidamente verificada a cada 2 segundos e se a mesma registar variações durante esse tempo, então provavelmente estará em modo “a pé”. Para também detetar este tipo de situações, um sensor barométrico pode ser usado e verificado a cada 2 segundos por forma a detetar variações na altura e pressão. O sensor de velocidade determina se o modo de transporte é “no veículo”. Todo este processo ocorre em atualizações de 2 a 4 segundos antes de iniciar a navegação.

Desta forma é determinado o modo de transporte que para além dos resultados “a pé” e “no veículo”, poderia ainda determinar um estado misto (“a pé e no veículo”) ou indeterminado.

De seguida apresenta-se que situações podem ser determinadas:

- A pé: deslocação a pé;
- *No veículo*: um sistema posicionado arbitrariamente num veículo;
- Indeterminado: o algoritmo de posicionamento correu sem apoio do algoritmo de determinação do modo de transporte;
- A pé e no veículo: neste caso estaríamos a falar num sistema colocado num individuo que por sua vez estaria numa plataforma móvel (por exemplo, um comboio);

Um terceiro algoritmo, igualmente importante na solução, é o algoritmo que determina a orientação do dispositivo móvel que vai estimar a orientação necessária para a aplicação das fontes de auxílio e restrições a aplicar ao algoritmo principal de navegação.

A solução oferecida faz uso do *hardware* existente em muitos dispositivos móveis hoje em dia, que geralmente são dispositivos MEMS de baixo custo.

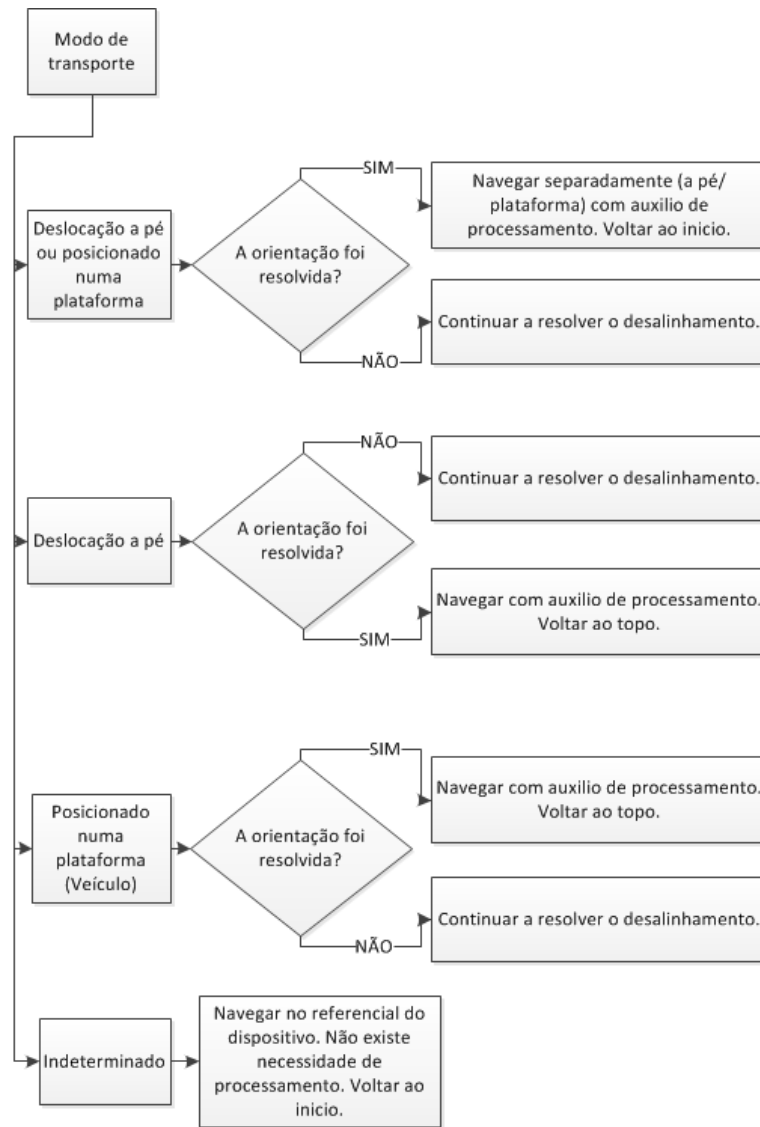


Figura 12 - Algoritmo de deteção do modo de transporte. (Adaptado de Hunter, et al., 2010)

2.6.2 Uninterrupted Portable Car Navigation System Using GPS, Map and Inertial Sensors Data

O sistema de navegação *Uninterrupted Portable Car Navigation System* (Davidson, et al., 2009) é um sistema portátil de navegação cujo objectivo é providenciar navegação ininterrupta e fiável quando o sinal de GPS não está disponível. Nesta abordagem são utilizados mapas digitais, acelerómetros 3D e um giroscópio 3D para medições direccionais de forma a melhorar a disponibilidade e fiabilidade do posicionamento em ambientes com sinal GPS fraco ou indisponível.

O sistema proposto não requer instalação no veículo aumentando assim a portabilidade. A estratégia de integração adoptada é a *Loosely coupled* com a implementação de um filtro de Kalman estendido e o desenvolvimento de um algoritmo de correspondência da posição num mapa digital. Desta forma a solução consegue disponibilizar uma solução de navegação contínua com auto-calibração dos sensores de inércia.

A figura 13 demonstra o conceito delineado para a elaboração do sistema.

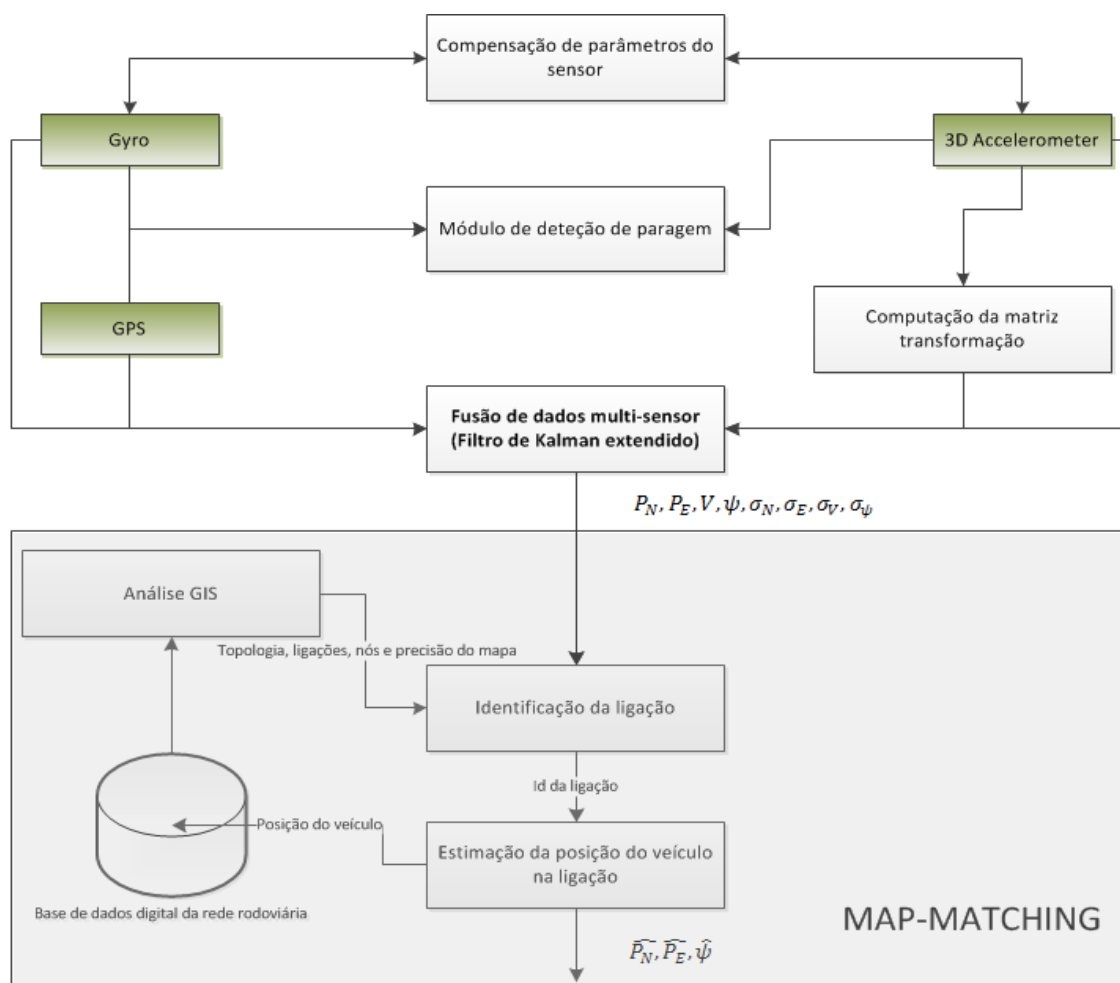


Figura 13 - Descrição conceitual das fontes de informação de navegação e o fluxo de dados do sistema portátil de navegação para automóvel. Os elementos de *hardware* são sombreados a verde. (Davidson, et al., 2009)

Relativamente à configuração do DR, este consiste num giroscópio para medições direcionais e um acelerómetro 3D o que satisfaz os requisitos mínimos para qualquer dispositivo móvel existente no mercado.

Este sistema é principalmente utilizado em telemóveis, periféricos GPS ou dispositivos de navegação GPS portáteis. Permite ainda que os sensores de DR sejam calibrados sempre que exista sinal GPS disponível. Quando o sinal deixa de existir, os sensores DR são usados para determinar a posição atual do veículo. Este sistema foi ainda melhorado com a implementação de um algoritmo de correspondência num mapa digital baseado no filtro de partículas (referido na secção 2.4).

2.6.2.1 Integração de dados

O sistema de navegação proposto inclui diferentes tipos de sensores de navegação e tecnologias. A metodologia seguida para a fusão de dados compreende diferentes etapas no processamento de dados:

- Detecção de paragem com os dados dos sensores de inercia;
- Processamento do posicionamento, velocidade e orientação com os dados dos sensores de inercia;
- Calibração dos sensores de inercia quando o veículo está parado;
- Integração GPS/INS;
- Correspondência de trajetos num mapa (*map-matching*).

A integração ou fusão dos dados provenientes do GPS e dos sensores da configuração INS é efetuada recorrendo a um EKF o qual vai combinar o posicionamento, velocidade e orientação processados pelo GPS com as medidas providenciadas pelo acelerómetro e giroscópio. O EKF usa os valores dos estados do filtro para prever os estados futuros.

2.6.2.2 Calibração e alinhamento

A fim de auxiliar o GPS através dos dados do giroscópio e do acelerómetro, procedimentos de inicialização têm de ser executados: alinhamento inicial e calibração.

O procedimento de alinhamento inicial inclui alinhamento horizontal com base nas saídas dos acelerómetros, estimativa do ângulo de *yaw* (guinada) e alinhamento azimute²².

A calibração inicial inclui a estimativa da tendência de erro do acelerómetro e desalinhamento e também a calibração do erro de desvio do giroscópio. O alinhamento horizontal e calibração são implementados em tempo real e podem ser realizados em qualquer momento quando o veículo está parado. Os ângulos de *yaw* (guinada) entre o enquadramento do sensor e o enquadramento do veículo podem ser estimados quando o veículo está em aceleração com direção constante. Este processo leva aproximadamente 20 segundos quando em modo completamente automático. Não existe necessidade de o utilizador efetuar ações especiais. Durante e após o alinhamento, o dispositivo deverá estar fixo.

2.6.2.3 Algoritmo de correspondência em mapa (*map-matching*)

O mapa digital da rede rodoviária é uma parte importante deste sistema de navegação. A base de dados do mapa é uma fonte de informação valiosa usada para melhorar a precisão do posicionamento gerado pela integração GPS/INS e na calibração dos sensores INS.

²² Azimute é uma direção definida em graus, variando de 0º a 360º. A direção de 0º graus corresponde ao Norte, e aumenta no sentido direto dos ponteiros do relógio.

Os algoritmos de correspondência em mapa geralmente consistem em duas etapas distintas: identificação da ligação da rua onde o veículo muito provavelmente estará a viajar e uma estimativa da posição do veículo na rua selecionada. Este algoritmo é baseado no teorema de Bayes, decorrência imediata do teorema da probabilidade total de forma a selecionar o segmento de rua onde o veículo está a passar. É também usado um filtro de partículas para uma aproximação recursiva de distribuições de probabilidade de interesse usando amostragens e pesos.

2.6.3 Vbox 3i

A Vbox 3i é uma solução de uma empresa inglesa, Racelogic, que desenvolve e fabrica sistemas eletrónicos para medir, armazenar, orientar, analisar e simular dados de veículos em movimento.

A Vbox 3i é um equipamento de registo de parâmetros nomeadamente a velocidade, posição e aceleração. Conectando a VBOX 3i à unidade de medição inercial Racelogic IMU (parte da família de periféricos existentes na marca), um filtro de Kalman em tempo real, incluído no *firmware* da solução, corrige os dados do GPS em áreas onde a receção de GPS é fraca e fornece valores de *Pitch*, *Roll* e *Yaw* a uma taxa de 100Hz. (Racelogic, 2011) Desta forma é capaz de produzir um *output* mais suave e preciso da velocidade devido às correções mútuas das duas origens de dados (GPS / INS). Pode ainda lidar com *blackouts* do GPS mantendo ainda alguma precisão. A figura 14 mostra uma Vbox 3i interligada ao módulo IMU da Racelogic.



Figura 14 – Vbox 3i + IMU.

A solução Vbox é constituída por um receptor GPS do tipo *Survey Grade* (Harch, 2006) capaz de providenciar uma taxa de actualização na ordem dos 100 Hz para todos os parâmetros (incluindo posição, velocidade e orientação). A velocidade e orientação são calculadas com o auxílio do efeito Doppler²³ (Lehtinen, 2001).

A Vbox 3i oferece também como característica distinta das unidades anteriormente fabricadas, um CPU a 400 Mhz com co-processador matemático ou unidade de vírgula flutuante capaz de efectuar 760 milhões de instruções por segundo. Desta forma, o dispositivo oferece a possibilidade de integração com o dispositivo IMU (*Racelogic IMU*) em tempo-real. A Vbox 3i

²³ Desvio da frequência da portadora devido ao deslocamento relativo entre a fonte transmissora - satélite e a fonte receptora - antena do receptor.

armazena a informação num cartão compact flash para facilitar a transferência dos dados para um computador.

Este dispositivo inclui quatro canais analógicos de alta definição de entrada para registo de informações provenientes de sensores externos, dois interfaces de barramento CAN²⁴ que permitem a interligação a dispositivos ou módulos específicos da Racelogic. A figura 15 apresenta um esquema das ligações existentes.

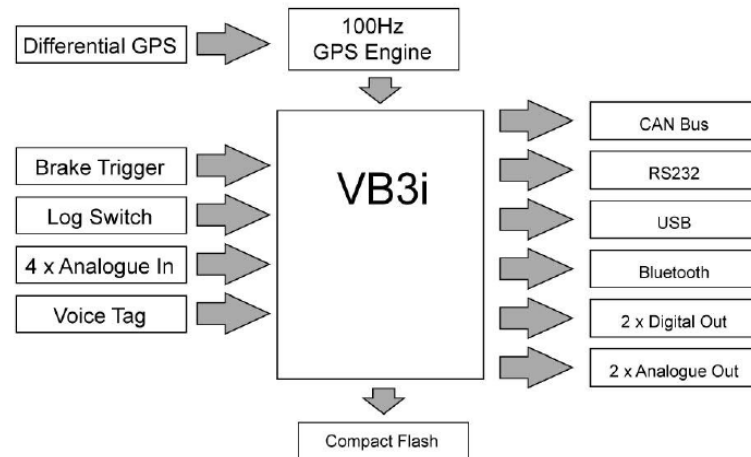


Figura 15 – Esquema I/O da Vbox. (Racelogic, 2011)

A solução Vbox 3i é uma solução modular onde os componentes podem ser acrescentados, melhorando ou adicionando funcionalidades.

A *Racelogic IMU 03* é um periférico que pode ser ligado à Vbox 3i ou ser usado como dispositivo Stand-alone. Quando ligado ao Vbox 3i pelo barramento CAN, este providencia informação sobre *Pitch*, *Roll* e *Yaw* (ver figura 10) assim como detalhes sobre a velocidade de forma integrada e transparente. Isto assegura que a informação exista mesmo quando a recepção de satélite é interrompida visto que a IMU disponibiliza dados continuamente apesar da desvantagem de acumular erros de integração do acelerómetro e giroscópio.

A IMU usa amostragens sincronizadas de 24 bits para cada um dos sensores de inércia e assegura um alto grau de precisão e taxas de resolução de *Yaw* tipicamente de 0.01 graus por segundo e resolução da aceleração até 0.002g.

Todos os dados são processados a 100Hz, melhorando assim os níveis de precisão e ruído. Os parâmetros suavizados por este processo de integração usando um filtro de Kalman incluem a velocidade, posição, altura, aceleração lateral e longitudinal e velocidade vertical.

²⁴ Barramento série assíncrono *Controller Area Network* (ou *Car Area Network*), permite a difusão de informação sobre uma rede, onde o suporte físico normalmente consiste num par trançado semelhante ao usado nas redes *ethernet*. Foi inicialmente concebido nos anos 80 pela Bosch, para conectar unidades de controlo de sistemas embebidos de um veículo automóvel, como por exemplo: unidade de controlo do motor; da transmissão; das portas; da climatização; etc.. Hoje em dia o seu uso vulgarizou-se, sendo aplicado também em sistemas de controlo industriais embebidos (Jorge & Sancho, 2009).

Na análise dos dados registados pelo *software* que acompanha a solução, pode-se distinguir quatro tipos de observações tratadas neste tipo de integração.

O parâmetro velocidade é analisado recorrendo ao gráfico demonstrado na figura 16 onde são ilustradas situações de falhas ou presença de ruído na leitura de sinal GPS num local extremamente arborizado (assinalado com linha vermelha). Com a integração de uma unidade IMU, o ruído é diminuído e é mantida uma precisão sobre as observações de velocidade durante as falhas de GPS (linha azul).

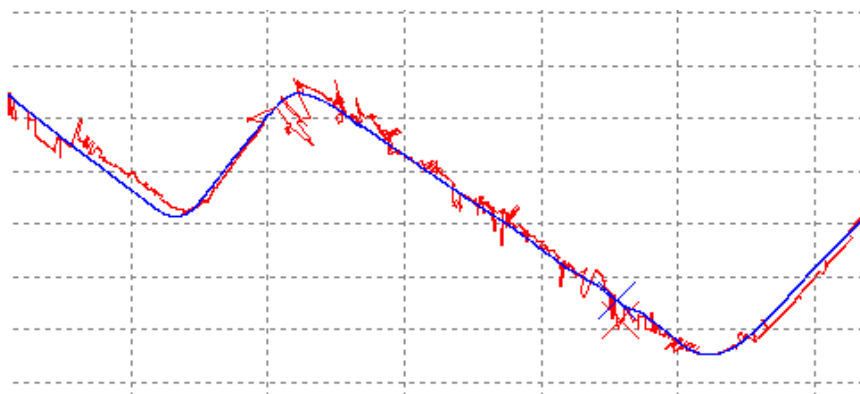


Figura 16 - Mapeamento do posicionamento do veículo (GPS - Vermelho; GPS+INS - AZUL). (Racelogic, 2011)

Relativamente ao filtro de Kalman providenciado pelo *firmware* do dispositivo, existem duas formas distintas de uso:

- *Real-time*: O filtro é usado em condições onde a recepção do sinal do satélite GPS varia muito, tais como zonas muito arborizadas ou edificadas. O filtro de Kalman suaviza a posição e velocidade em tempo real e provisiona informações durante curtos espaços de tempo onde não exista cobertura de sinal GPS, por exemplo túneis.
- *Post-processing*: Este modo providencia uma melhoria significativa no ruído do que o filtro de Kalman embutido na Vbox. É superior porque é aplicado no pós-processamento o que permite que os dados sejam analisados posteriormente. Este filtro é usado no *software* que acompanha a solução e para ser usado, a funcionalidade *Kalman Filter* no *firmware* da Vbox deverá estar desabilitada.

2.6.4 Ctrack

O sistema Ctrack (Thiagarajan, et al., 2011) visa providenciar uma solução de mapeamento de trajetória de forma eficiente em termos energéticos para o dispositivo móvel, visto que não fará uso do GPS, com o máximo de precisão usando medições brutas de posicionamento provenientes de ERB²⁵ (Estações radio base).

A solução foi concebida para ser utilizada em dispositivos móveis que reúnam o mínimo de requisitos para a operacionalidade. Oferece uma solução alternativa ao uso do GPS como

²⁵ Estação Rádio Base (ERB) ou “Cell site” é a denominação dada a um sistema de telefonia móvel para a Estação Fixa com que os dispositivos móveis se comunicam.

sistema de localização. É referida pela particularidade de também esta oferecer uma solução integrada em dispositivos móveis.

O mapeamento da trajetória, que envolve uma sequência de observações de posicionamento e definição do caminho mais provavelmente utilizado pelo utilizador do equipamento é um componente importante em serviços de localização e sistemas de navegação assim como de sistemas turísticos.

Usando apenas o mecanismo *Cellular GSM fingerprint* em vez do GPS o sistema permite poupança de energia, visto que o GPS ainda é um dispositivo que consome bastante energia.

Um sistema semelhante a este é o Vtrack (Thiagarajan, et al., 2009) que tem como principal fornecedor de observações de posicionamento, o sistema WiFi.

2.6.4.1 Integração de observações de diferentes sensores

Obviamente que usando apenas o mecanismo *Cellular GSM fingerprint* proposto pelo sistema Ctrack, iriam ser apenas processados posicionamentos com uma precisão média de 175 metros produzindo uma trajetória muito irregular. A Ctrack enfrenta este desafio usando o modelo oculto de Markov em dois passos que trata as medições brutas de posicionamento provenientes de ERB sem convertê-las em coordenadas geográficas e integra ou funde essa informação com outros sensores presentes nos dispositivos móveis tais como acelerómetros (detecção de movimento), compassos magnéticos (detecção de mudanças de direção).

Para dispositivos móveis, esta estratégia é de grande importância pois permite a um sistema de navegação um baixo consumo energético.

A figura 17 mostra um gráfico onde é possível comparar o consumo energético ao longo do tempo dos diferentes mecanismos existentes no dispositivo, nomeadamente o GPS (ativo a cada 1 ou 120 segundos), o *Wifi* a cada 1 segundo e o conjunto magnetómetro mais acelerómetro.

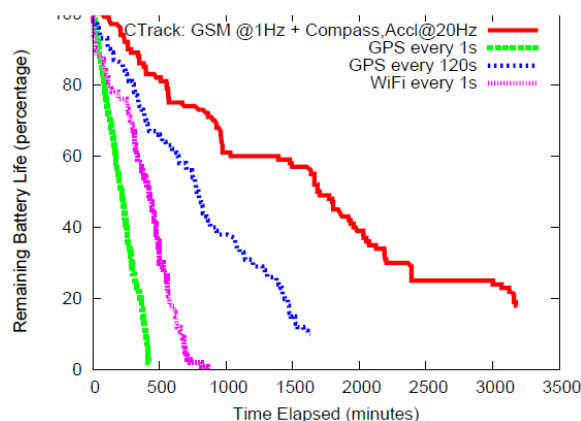


Figura 17 - Consumo energético Ctrack vs GPS vs Wifi. (Thiagarajan, et al., 2011)

2.6.4.2 Arquitetura do sistema Ctrack

Importa aqui descrever a arquitetura do sistema Ctrack que consiste em dois componentes de *software*, o *Ctrack Phone Library* e o *Ctrack Web Service*. O primeiro é responsável por recolher, filtrar e examinar os dados provenientes do GSM e dos sensores dos dispositivos móveis e transmiti-los via qualquer rede sem fios disponível no momento (3G, Wi-Fi ou outra) para o serviço Web (*Ctrack Web Service*) que corre o algoritmo de mapeamento da trajetória. Este algoritmo corre num servidor de forma a evitar armazenamentos no dispositivo móvel de dados do mapa e providenciar um acesso centralizado às informações através do dispositivo móvel ou página *web* e assim visualizar e analisar os trajetos efetuados. A arquitetura está esquematizada na figura 18.

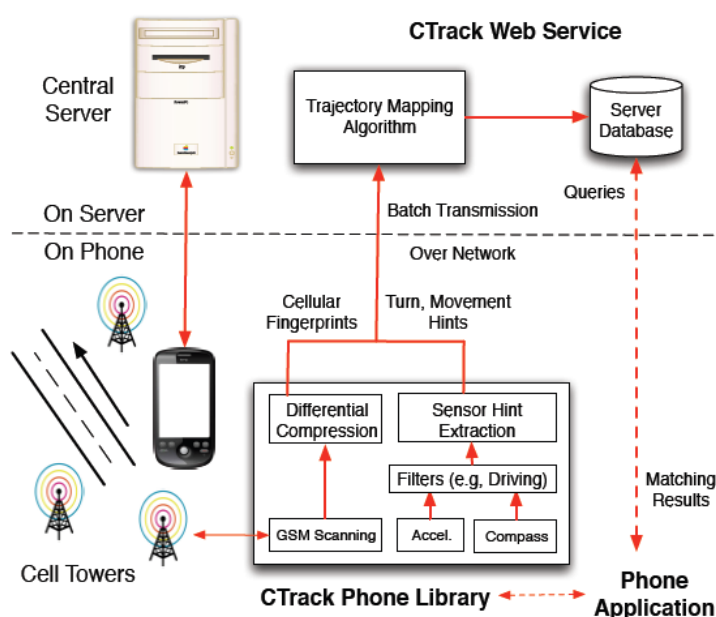


Figura 18 - Arquitetura do sistema Ctrack. (Thiagarajan, et al., 2011)

2.6.5 GNSS/INS Integration in Vehicular Urban Navigation

O estudo “solução combinada GPS/GLONASS com uma IMU de tecnologia MEMS de baixo custo para navegação urbana” (Angrisano, et al., 2010) pretende analisar formas de expansão na utilização de sistemas de navegação integrados GPS/INS dando um novo interesse ao sistema de navegação GLONASS e explicar as vantagens existentes na integração GPS/GLONASS/INS usando as arquiteturas ou estratégias de integração *loosely coupled* e *tightly coupled*.

Neste estudo é colocado o problema conhecido sobre a disponibilidade de sinal GNSS em localizações urbanas, caracterizado pela existência de estruturas de bloqueio de sinal (*e.g.* túneis e edifícios altos) provocando degradação e/ou bloqueio de sinal. Segundo (Angrisano, et al., 2010), no conjunto de investigações efetuadas, normalmente duas aproximações são debatidas para tentar resolver ou minimizar o problema:

- Aumentar o número de satélites visíveis, e isto normalmente significa adicionar sistemas GNSS à solução;

- Integrar GNSS com sensores externos, normalmente INS.

O autor pretende tirar partido das duas soluções efetuando uma integração de dois sistemas GNSS (GPS e GLONASS) com um sistema de navegação por inercia (INS) e desta forma avaliar o papel que terá a adição do GLONASS num sistema integrado GPS/INS no aumento da precisão do posicionamento através do aumento da disponibilidade de satélites. Para além disto é efetuada uma avaliação sobre a performance dos algoritmos numa estratégia de integração *loosely coupled* e *tightly coupled*, com ou sem GLONASS. Por último é efetuada uma avaliação sobre o benefício no uso do GLONASS relacionado com algumas restrições, tais como a altura.

O trabalho detalha algumas diferenças já referidas no capítulo 2.1 entre o GLONASS e o GPS mas que importa detalhar:

- Os sistemas de referência de coordenadas do GLONASS e GPS não coincidem. Enquanto as efemérides dos satélites GPS referem-se ao *datum*²⁶ WGS-84 (*World Geodetic System* 1984), as do GLONASS referem-se a um outro, o PZ90 (*Parametry Zemli* 1990). De forma a resolver este problema, uma transformação matemática será usada para uniformizar os valores.
- Os padrões de tempo adotados pelo GPS e pelo GLONASS. O padrão de tempo representa a espinha dorsal de qualquer dos dois sistemas, cujas observações proporcionam o cálculo da distância. O tempo GLONASS é referenciado ao UTC²⁷ (SU) gerido pela Rússia, enquanto o GPS refere-se ao UTC mantido pelo *US Naval Observatory*, ou simplesmente UTC (USNO). Para resolver, as observações têm de ser reduzidas a um mesmo sistema de tempo para o cálculo de posição. Em termos práticos, a consequência é que, num recetor combinado, a cada observação é necessário determinar-se o *offset* entre o relógio do recetor e o tempo GLONASS, além do correspondente *offset* em relação ao tempo GPS. Isto implica na necessidade de cinco satélites visíveis para o cálculo de posição (tem-se agora cinco incógnitas: as três coordenadas do recetor e os dois offsets), ao contrário de apenas quatro (as três coordenadas e um offset) no caso do GPS sozinho.

O estudo envolve uma IMU baseada em tecnologia MEMS evidenciando as vantagens do mesmo em termos de preço, peso, dimensão e baixo consumo energético. Obviamente as desvantagens têm a ver com a precisão afetada sobretudo pelos erros existentes nas observações dos acelerómetros e giroscópios. O desvio ou tendência do sensor (*sensor bias*) é definido pela média das observações de saída (*output*), obtida durante um período específico, com condições operacionais específicas e com entradas (*input*) nulas. Este erro de sensor está dividido em duas partes: O erro constante ao longo do intervalo de tempo desde que foi ligado (*turn-on bias*) até ser desligado, e um erro de origem aleatória (*in-turn bias*) enquanto o sensor está em funcionamento. O primeiro é de natureza determinística e pode ser determinado na fase de calibração inicial do sistema. O segundo deverá ser modelado como um processo estocástico. Outro erro referido é o erro de escala que é de natureza determinística e também modelado como um processo aleatório.

²⁶ Em cartografia refere-se ao modelo matemático teórico da representação da superfície da Terra ao nível do mar utilizado pelos cartógrafos

²⁷ UTC (*Coordinated Universal Time*) ou Tempo universal coordenado sendo o fuso horário a partir do qual se calculam todas as outras zonas horárias do mundo.

Os erros dos sensores de inercia são expressos em termos de ARW (*angular random walk*) para o caso dos giroscópios e VRW (*velocity random walk*) para o caso dos acelerómetros.

2.6.5.1 Integração dos sistemas de navegação

Na modelação do sistema GNSS, as variáveis de estado são a posição, a velocidade e os erros de relógio. Se apenas um sistema de GNSS estiver a ser considerado (*i.e.* GLONASS apenas ou GPS apenas) o erro de relógio é modelado usando dois estados representativos de erros: *Bias* e *Drift*. Se dois GNSS são combinados para a solução, uma outra variável de estado representativa da diferença de tempo entre os dois sistemas deverá ser usada.

Na integração GNSS/INS, usando a estratégia *Loosely coupled*, um filtro de Kalman é usado para combinar os parâmetros de GNSS com os do INS. Apesar desta estratégia ser de menor complexidade, o maior problema advém da possibilidade de ser impossível processar o posicionamento pelo GNSS devido à insuficiência de satélites e logo não conseguir atualizar o INS. Isto resulta em erros de posicionamento maiores relativamente à estratégia *Tightly coupled*.

Para calcular o posicionamento com o GNSS, o método dos mínimos quadrados é o preferido para simplificar uma comparação entre *Loosely coupled* e *Tightly coupled*, isto porque usando o método dos mínimos quadrados na integração *Loosely coupled*, os resultados serão equivalentes numa integração *Tightly coupled* enquanto existirem satélites suficientes para processar uma solução. Assim sendo, a diferença da posição e velocidade provenientes da solução INS e da solução GNSS (usando o método dos mínimos quadrados ponderados) são usados como observações de entrada no filtro de Kalman.

A estratégia *Tightly coupled* é também referida como centralizada, pois existe apenas um filtro de Kalman central responsável pelo processamento das observações GNSS e INS. Segundo o autor, este processo é mais complicado mas parece ter melhor performance porque a fase de atualização pode ser realizada independentemente de existirem satélites suficientes ou não para uma solução GNSS. O vetor de estados estabelecido nesta estratégia contempla já os erros de relógio do recetor GNSS. Se as observações do GLONASS forem incluídas, a diferença do tempo entre o sistema GLONASS e GPS também deve ser considerada e modelada como uma constante aleatória.

Esta solução também faz referência a métodos auxiliares para melhorar o resultado final. Os métodos baseiam-se na utilização da direção obtida do GNSS (*Heading*) assim como das observações da velocidade e altura do veículo. Para incorporar o primeiro método no modelo de observações GNSS/INS, é utilizada uma equação que relaciona os erros de direção com os erros dos estados de sistema. O segundo método permite reduzir a acumulação de erros de INS aplicando restrições baseadas no movimento do veículo. Assim sendo, o autor assumiu que o veículo não desliza lateralmente e não sofre qualquer tipo de salto de tal forma que o movimento é na direção longitudinal do veículo. As suposições em cima são chamadas de restrições não-holonómicas²⁸.

²⁸ Uma restrição não-holonómica é uma equação não integrável envolvendo os parâmetros de configuração e suas derivadas. São limitações de um determinado objecto.

Para validar os benefícios desta arquitetura, um recetor GNSS capaz de receber sinais do GPS e GLONASS deverá ser usado. Com este projeto, foi demonstrado que um dos principais benefícios no uso de observações GLONASS está no aumento da disponibilidade de satélites para a obtenção de observações GNSS. No entanto, este aumento é pouco notório em zonas onde a degradação e bloqueio do sinal GNSS é regra (*e.g.*, zonas urbanas com edifícios altos).

Num teste efetuado (Angrisano, et al., 2012), foi possível constatar o seguinte:

- Em ambientes onde a visibilidade de satélites é insuficiente para uma solução de posicionamento baseado em GNSS, em aproximadamente 50% do tempo, os benefícios na inclusão das observações do GLONASS são mínimos, numa integração do tipo *Loosely coupled*. No entanto, numa integração *Tightly coupled*, o GLONASS providencia melhorias consideráveis;
- Em ambientes onde o posicionamento por satélite é possível em 70% por tempo, a inclusão do GLONASS numa estratégia *loosely coupled* é similar em termos de performance a uma solução com apenas GPS empregando uma estratégia *tightly coupled*. Isto sugere que um sistema simples integrando GLONASS é possível sem sacrificar a performance;
- Em ambientes onde a localização por satélite é possível a maior parte das vezes, a inclusão do GLONASS traz poucos benefícios;
- A inclusão dos métodos auxiliares produz uma melhoria significativa de performance, independentemente da estratégia utilizada.

2.6.6 Improved Vehicle Positioning in Urban Environment through Integration of GPS and Low-Cost Inertial Sensors

Um trabalho de referência na área de integração de sensores de baixo custo (normalmente presentes nos dispositivos móveis) (Davidson, et al., 2010) mostra como a navegação GPS pode ser melhorada e como pode ser melhorada também a performance em centros urbanos com sensores INS usando um filtro estendido de Kalman. Neste trabalho de investigação com o título “*Improved Vehicle Positioning in Urban Environment through Integration of GPS and Low-Cost Inertial Sensors*”, os sensores INS são calibrados quando existe sinal GPS disponível. Na ausência do mesmo, por motivos demonstrados na secção 2.3, os sensores previamente calibrados são usados para determinar a posição do veículo.

Neste projeto, a configuração do INS consiste num giroscópio para medições angulares, e um acelerómetro de 3 eixos. Ambos os dispositivos são baseados em tecnologia MEMS.

O sistema proposto inclui diferentes tipos de sensores e tecnologias para navegação. O conceito pode ser demonstrado recorrendo à figura 19.

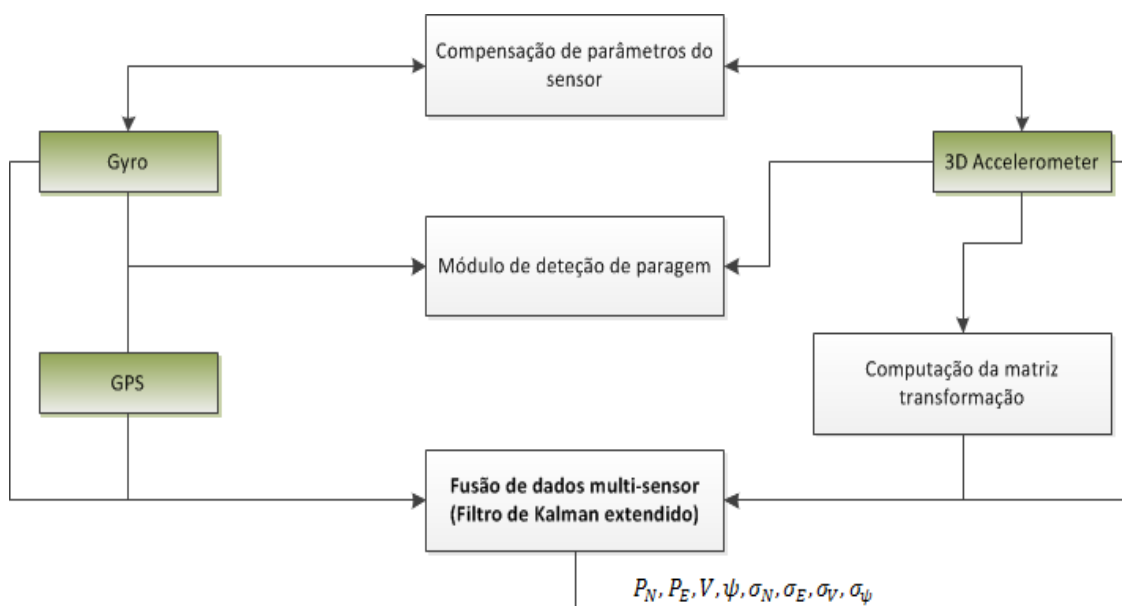


Figura 19 - Descrição conceptual das fontes de informação para navegação e fluxo de dados do sistema portátil de navegação automóvel. (Davidson, et al., 2009)

Esta metodologia obriga a diferentes etapas e tratamento de *inputs* no processamento de dados:

- Dados dos sensores de inercia para deteção de paragens;
- Dados dos sensores de inercia para determinação da posição, velocidade e direcção;
- Calibração dos sensores de inercia quando o veículo está parado;
- Integração do GPS com INS para solução de navegação final;

2.6.6.1 Integração de dados GPS/INS

O algoritmo de fusão de dados em tempo real utiliza um filtro de Kalman estendido para combinar dados processados do GPS, tais como velocidade, posição e direcção, com os dados provenientes do acelerómetro e do giroscópio, nomeadamente a aceleração e direcção. Desta forma, o EKF usa os valores dos estados do filtro para prever os futuros estados através de um modelo dinâmico baseado nas equações INS e assim estimar a posição, velocidade e direcção. Importa agora explicar a construção do modelo dinâmico descrito segundo a equação (10):

$$\dot{x} = \begin{bmatrix} \dot{P}_N \\ \dot{P}_E \\ \dot{V} \\ \dot{\psi} \end{bmatrix} = f(x) = \begin{bmatrix} V \cos \psi \\ V \sin \psi \\ a_L \\ \omega \end{bmatrix} \quad (10)$$

onde P_N e P_E representam o posicionamento a Norte e Este do veículo, ψ representa a direcção do veículo, ω é a taxa de medição da direcção, V é a velocidade, a_L é a medição da aceleração do veículo na direcção longitudinal. A aceleração longitudinal do veículo é calculada pela transformação do vetor de aceleração tridimensional medido pelos acelerómetros no plano

dos sensores para um plano do veículo e calcular a projeção do vetor transformado nos eixos longitudinais do veículo. Também inclui o efeito das forças gravitacionais devido ao tipo de trajeto. Assim sendo, este é definido na equação (11).

$$a_L = a_T + b_L + g \cdot \theta + n, \quad (11)$$

onde a_T é a aceleração real longitudinal do veículo, g é a constante gravitacional, θ é a categoria do trajeto, b_L é o erro longitudinal de aceleração e n é o ruído do acelerómetro. De forma a modelar corretamente os erros do acelerómetro e do giroscópio, assim como a categoria da trajetória (influencia a força de gravidade), o vetor de estados vai incluir dois estados adicionais: o erro do giroscópio $\delta\omega$ e o erro e desalinhamento do giroscópio δa conforme definido na equação (12).

$$x = [P_N \ P_E \ V \ \psi \ \delta\omega \ \delta a]^T \quad (12)$$

A covariância do filtro de Kalman é dada pela equação (13).

$$P(k+1|k) = \Phi(k)P(k|k)\Phi(k)^T + \Gamma(k), \quad (13)$$

onde Φ é o equivalente discreto da matriz transição contínua F e Γ é o equivalente discreto da matriz densidade do ruído do processo. Uma vez que o modelo dinâmico é não linear, necessita de ser linearizado no que diz respeito às variáveis de estado. A linearização é realizada em torno dos valores estimados anteriormente para que a matriz transição F tome da seguinte forma:

$$F = \begin{bmatrix} 0 & 0 & \cos \psi & -V \sin \psi & 0 & 0 \\ 0 & 0 & \sin \psi & -V \cos \psi & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & -\frac{1}{T_g} & 0 \\ 0 & 0 & 0 & 0 & 0 & -\frac{1}{T_a} \end{bmatrix} \quad (14)$$

Os estados para os erros do giroscópio e acelerómetro foram modelados como um processo Gauss-Markov de 1ª ordem com constantes de tempo T_g e T_a respetivamente e os valores destas constantes de tempo e parâmetros de ruído são usados no filtro de Kalman.

Testes executados pelo autor levaram a concluir que quando o sinal GPS está disponível, o algoritmo de integração para as medições do GPS e INS é robusto o suficiente para suportar variações dos erros dos sensores e desvios rápidos no trajeto efetuado. A taxa de amostragem a 10 Hz é garantia para a não existência de degradação no ambiente urbano regular e esta taxa de propagação permite uma aproximação à 1ª ordem da matriz de transição contínua pelo seu equivalente discreto $\Phi = I + F\Delta t$, onde I é a matriz identidade e Δt é o período da amostra. O vetor de observação é calculado pela diferença das posições e velocidades do GPS e INS.

Neste projeto, a taxa de atualização para o GPS foi determinada em 1Hz para que fosse reduzido ao mínimo a necessidade de processamento. A orientação também pode ser

determinada pelas observações de velocidade. Desta forma, se não existir nenhum desvio rápido do veículo, a orientação pode ser determinada pela equação (15):

$$\psi_{GPS} = \arctan\left(\frac{V_{GPS}^{este}}{V_{GPS}^{norte}}\right), \quad (15)$$

onde V_{GPS}^{este} e V_{GPS}^{norte} são as velocidades medidas do GPS a Este e a Norte respetivamente. O desvio padrão do erro de direção pode ser aproximado determinando a equação (16):

$$\sigma(\psi_{GPS}) = \sigma(\delta V_{GPS})/V_{GPS}. \quad (16)$$

A orientação pelo GPS é calculada apenas quando o veículo está em velocidade suficiente para ser possível a sua determinação.

Para que o acelerómetro e giroscópio sejam usados como meios auxiliares ao GPS, os procedimentos de inicialização, alinhamento inicial e calibração, deverão ser executados. O alinhamento inicial inclui o alinhamento horizontal baseado nos *outputs* do acelerómetro, a estimativa do angulo *yaw* e alinhamento azimute.

A calibração inicial inclui as estimativas do desalinhamento e erros de acelerómetro e calibração do giroscópio. Os procedimentos anteriores são efetuados em tempo real quando o veículo encontra-se parado e demora cerca de 20 segundos até o processo estar finalizado. Como este procedimento tem de ser efetuado várias vezes durante o trajeto efetuado, um módulo de deteção de paragem foi adicionado. Este algoritmo de deteção de paragem baseia-se na análise da aceleração do veículo medida pelos acelerómetros e a sua variância.

O autor especifica algumas conclusões interessantes do projeto. Na primeira experiência efetuada com esta configuração, ao longo do trajeto foi simulado um bloqueio de sinal GPS o que originou o processamento da posição, velocidade e direção continua através dos *inputs* do acelerómetro e giroscópio.

Como ponto de referência do sistema de navegação, foi usado um sistema DGPS preciso, em paralelo, para estudar a variação das trajetórias GPS/INS e INS apenas. A precisão, quando em navegação DR, ou seja, durante o bloqueio GPS, foi boa tendo em conta que 30 segundos de bloqueio originaram apenas 20 metros de erro de posição.

3 Orientação e referenciais de navegação

O tipo de sistema de navegação utilizado no desenvolvimento da ferramenta de análise no âmbito deste trabalho, referido no início deste estudo na subsecção 2.2.2, é conhecido como sistema de navegação do tipo *Strapdown*, pois a IMU elaborada não está isolada das rotações do dispositivo onde está instalada. Para descrever a orientação do veículo é necessário por vezes determinar os referenciais, utilizando fórmulas matemáticas representativas da rotação. Neste capítulo apresentam-se as alternativas para a descrição da orientação e movimento nos diversos referenciais.

3.1 Representação da orientação

A necessidade de orientação e avaliação do vetor aceleração, velocidade e posição no referencial desejado, quando o obtemos no referencial do dispositivo, necessita de uma matriz de transformação ou rotação e esta deve ser o mais rigorosa possível.

Tendo em conta a integração necessária destas medidas ao longo do tempo, mesmo um pequeno desalinhamento pode causar grandes erros na estimativa. A matriz rotação é uma transformação linear que quando multiplicada por um qualquer vetor provoca uma rotação desse vetor segundo um eixo mantendo a sua norma. Trata-se de uma representação matricial ortonormada de dimensão 3x3, utilizada para descrever rotações de vetores e efetuar o mapeamento de coordenadas entre referenciais (figura 20). São necessárias à rotação do dispositivo no referencial do corpo para o referencial de navegação e deverão ser atualizadas continuamente visto que o referencial do corpo encontra-se em constante rotação.

Num sistema de navegação inercial, a representação da orientação do dispositivo relativamente ao referencial global de navegação assim como o processo de atualização na propagação da matriz é obtida através de uma de várias representações conhecidas. As mais comuns incluem os ângulos de Euler, quaterniões e/ou cossenos diretores.

3 Orientação e referenciais de navegação

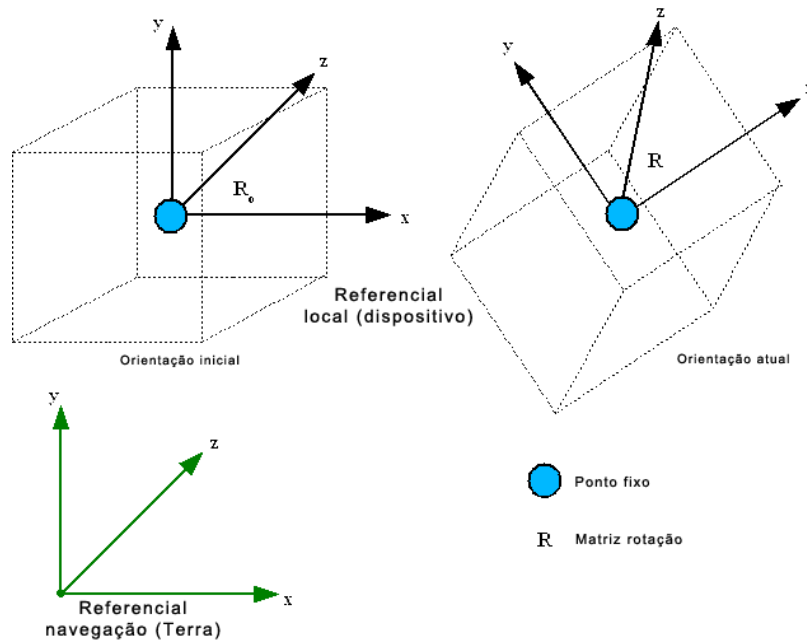


Figura 20 - A matriz rotação R é utilizada para identificar a orientação atual relativamente à inicial.

De forma geral, a transformação de um vetor ρ^b no referencial do corpo para o vetor ρ^n no referencial global de navegação, pode ser representada pela equação (17).

$$\rho^n = C_b^n \cdot \rho^b = \begin{bmatrix} c_{11} & c_{12} & c_{13} \\ c_{21} & c_{22} & c_{23} \\ c_{31} & c_{32} & c_{33} \end{bmatrix} \cdot \rho^b, \quad (17)$$

onde C_b^n é chamada de matriz dos cossenos diretores ou DCM (*Direction Cossine Matrix*).

Da expressão anterior é direta a relação de C_b^n com a transformação inversa C_n^b tendo em conta que

$$C_n^b = (C_b^n)^{-1} \quad (18)$$

Devido à comutatividade do produto interno (de cada um dos elementos da matriz DCM composta pelo produto interno dos vetores ortonormais de cada um dos referenciais) e da relação direta descrita na expressão anterior, é possível obter uma relação direta entre a inversa e a transposta das matrizes ortonormadas.

$$(C_b^n)^{-1} = (C_b^n)^T \quad (19)$$

As abordagens para propagar a matriz de transformação são:

1. Utilização direta da matriz dos cossenos diretores C_b^n , uma matriz 3X3 que contém os cossenos dos ângulos entre o referencial local e o referencial global. Esta matriz é propagada diretamente a partir das velocidades angulares fornecidas pelo giroscópio. As colunas da matriz representam os vetores unitários no referencial b projetados ao longo dos eixos do referencial n .

2. Outra abordagem consiste na utilização dos ângulos de Euler cuja transformação C_b^n de b para n é definida a partir de rotações sucessivas em torno dos eixos do referencial n . Denominam-se *pitch* θ , *roll* ϕ e *Yaw* γ (figura 21) os ângulos representativos da rotação do referencial do corpo relativo ao referencial global nos eixos X, Y e Z²⁹ respectivamente e os mesmos são especificados por três matrizes cossenos diretores conforme as equações (20),(21) e (22).

- a. Rotação em torno de Z:

$$C_\gamma = \begin{bmatrix} \cos\gamma & \sin\gamma & 0 \\ \sin\gamma & \cos\gamma & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad (20)$$

- b. Rotação em torno de Y:

$$C_\phi = \begin{bmatrix} \cos\phi & 0 & -\sin\phi \\ 0 & 1 & 0 \\ \sin\phi & 0 & \cos\phi \end{bmatrix}, \quad (21)$$

- c. Rotação em torno de X:

$$C_\theta = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\theta & \sin\theta \\ 0 & -\sin\theta & \cos\theta \end{bmatrix}, \quad (22)$$

onde a matriz DCM pode ser escrita conforme a equação (26.

$$C_b^n = (C_\gamma)^T \cdot (C_\phi)^T \cdot (C_\theta)^T, \quad (23)$$

$$C_b^n = \begin{bmatrix} \cos\gamma \cdot \cos\phi & \cos\gamma \cdot \sin\phi \cdot \sin\theta - \sin\gamma \cdot \cos\theta & \cos\gamma \cdot \sin\phi \cdot \cos\theta + \sin\gamma \cdot \sin\theta \\ \sin\gamma \cdot \cos\phi & \cos\gamma \cdot \sin\phi \cdot \sin\theta + \cos\gamma \cdot \cos\theta & \sin\gamma \cdot \sin\phi \cdot \cos\theta - \sin\theta \cdot \cos\gamma \\ -\sin\phi & -\cos\phi \cdot \sin\theta & \cos\phi \cdot \cos\theta \end{bmatrix}$$

3. A terceira e última abordagem refere-se ao uso de quatérniões que podem ser usados para parametrizar a orientação no referencial b relativamente ao referencial n por meio de uma única rotação de um ângulo α em torno de um vetor ρ unitário definido em n . O quatérnião Q_b^n que permite a transformação dos sistemas de coordenadas do referencial b para n é um vetor de 4 componentes

$$Q_b^n = \begin{bmatrix} q_0 \\ q_1 \\ q_2 \\ q_3 \end{bmatrix} = \begin{bmatrix} \cos\frac{\alpha}{2} \\ \sin(\frac{\alpha}{2})\rho_X \\ \sin(\frac{\alpha}{2})\rho_Y \\ \sin(\frac{\alpha}{2})\rho_Z \end{bmatrix}, \quad \|\rho\| = 1, \quad (24)$$

onde ρ_X, ρ_Y e ρ_Z são os componentes do vetor ρ descritos em n .

²⁹ Desenvolvendo para Android, o eixo do X define-se como sendo o eixo do Pitch e o Y como sendo o eixo Roll ou num sistema de coordenadas ENU, X=East, Y=North e Z=up.

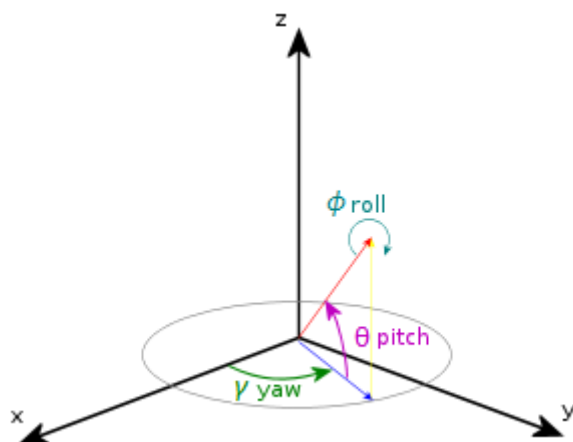


Figura 21 - Ângulos *pitch*, *roll* e *yaw*.

De todas as formas apresentadas, os ângulos de Euler são mais fáceis de entender ou de menor complexidade uma vez que os ângulos θ , ϕ e γ correspondem directamente a ângulos de rotação do referencial b para n .

3.2 Sistemas de coordenadas ou referenciais

Os sistemas de coordenadas são usados para referenciar pontos no espaço. Os sistemas de navegação inercial necessitam de transformação de pontos entre diferentes sistemas de referência ou referenciais.

Os acelerómetros e giroscópios medem a aceleração e velocidade de rotação relativamente a um referencial inercial e esta representação dá pelo nome de 6 graus de liberdade (6DOF). Os sinais do giroscópio são usados para determinar a orientação do dispositivo e compensar a aceleração da gravidade. No desenvolvimento, será usado a fusão de vários sensores para a estimativa da orientação.

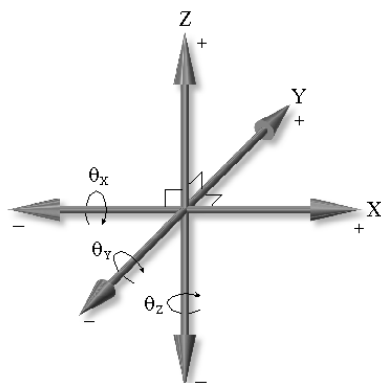


Figura 22 - 6 graus de liberdade (6DOF) num sistema de navegação inercial do tipo *Strapdown*.

Outros sensores tais como o GPS reportam a informação num referencial global conhecido como *Earth Centered Earth Fixed* (ECEF). O sistema de navegação necessita de transformar esta informação para um sistema de coordenadas comum.

De seguida iremos introduzir os sistemas de coordenadas mais conhecidos e que serão suficientes para a navegação local e logo usados neste tipo de aplicações:

- Sistema de coordenadas ECI (*Earth Centered Inertial*) – Este sistema de coordenadas tem origem no centro de massa da Terra sendo que o seu eixo X aponta para o sol passando pela linha do equador, o eixo Z passa pelo Pólo norte e o eixo Y é orientado segundo a regra da mão direita.
- Sistema de coordenada ECEF (*Earth Centered Earth Fixed*) – Tem origem no centro de massa da Terra onde o eixo X passa pelo cruzamento entre a linha do equador e o meridiano de Greenwich, o eixo Z passa pelo Pólo norte e o eixo Y é orientado segundo a regra da mão direita.
- Sistema de coordenadas de Navegação *East-North-Up* (ENU) – A sua origem pode ser estabelecida em qualquer ponto do globo terrestre onde o eixo X aponta para o norte geográfico, o eixo Z aponta para cima com origem no centro da terra e o eixo Y é referenciado com a regra da mão direita. No desenvolvimento para Android, os eixos X e Y são trocados de modo a facilitar o cálculo tendo em conta que o eixo Y do dispositivo móvel Android é o eixo longitudinal e que queremos que referencie o norte.
- Sistema de coordenadas fixo ao veículo (corpo) – É o sistema conhecido como RPY (do inglês *Roll, Pitch e Yaw*). A origem é fixa ao veículo, preferencialmente no centro de gravidade do mesmo. O eixo Y, eixo *roll* no Android, aponta para a direção frontal do veículo, o eixo X, eixo *pitch*, deverá ser ortogonal a Y e apontar para a esquerda e o eixo Z aponta para cima.

É possível assim efetuar a transformação entre os referenciais local, referidos doravante como “b” (e.g., x^b) e global de navegação (e.g., x^n), referido doravante como “n”. A constatação dos referenciais pode ser visualizada na figura 23.

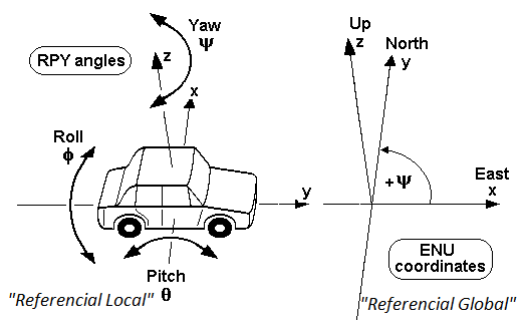


Figura 23 - Referencial local e global.

3.2.1 Cálculo da orientação inicial

O cálculo da direção inicial, neste caso o Azimute, *i.e.*, a direção definida em graus de 0° a 360° medido a partir do norte magnético deverá ser obtida por meio dos sensores Magnetômetro, Gravidade (acelerômetro) e giroscópio.

O uso dos três sensores torna-se essencial se entendermos que para o cálculo da direção usamos o magnetômetro, calculando o ângulo α em relação ao polo norte magnético da Terra dado pela seguinte equação (25):

$$\alpha = \arctan\left(\frac{m_x}{m_y}\right), \quad (25)$$

onde m_x e m_y correspondem à intensidade do campo magnético nos eixos X e Y respectivamente, mas se os sensores não estiverem alinhados com o plano da terra a solução será aplicar o sensor que forneça a rotação do plano do dispositivo relativamente à terra e obter os novos valores já no referencial global. Ainda que se use medidas baseadas em coordenadas geográficas, este valor terá de ser retificado porque os polos magnéticos da Terra não coincidem com os polos Geográficos da Terra que são determinados do eixo de rotação da Terra. Para isso usa-se o valor da declinação do α encontrado correspondente ao local.

3.2.2 Determinação da aceleração e velocidade

A aceleração medida pelo sensor acelerômetro necessita de ser compensada, pois este sensor mede não só a aceleração do veículo, ou seja a aceleração linear, mas também a aceleração centrípeta, a coriolis e a gravitacional.

A terra, por possuir uma massa grande gera uma força de atração de massa vertical à sua superfície e em direção ao centro. Logo o sensor acelerômetro irá sempre medir esta força mesmo estando estacionário.

Um modelo possível para as medidas de um acelerômetro em movimento pode ser dado pela equação (26).

$$f = \frac{dv^b}{dt} + \omega \times v^b - g^b \quad (26)$$

onde $\frac{dv^b}{dt}$ é a aceleração linear, $\omega \times v^b$ é a aceleração centrípeta e g^b é o vetor gravidade no referencial do dispositivo. Assim sendo, a força centrípeta, a força necessária para manter um movimento circular (Figura 24) poderá ser removida para que os resultados estimados sejam válidos para qualquer movimento retilíneo ou de rotação.

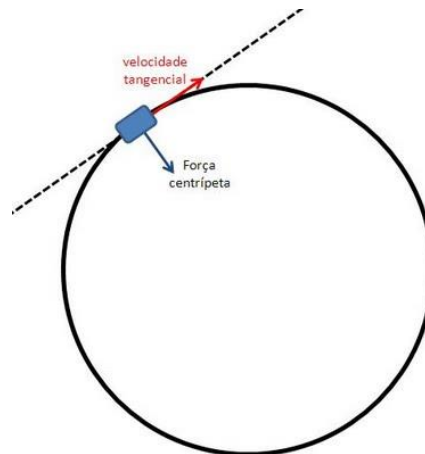


Figura 24 - Efeito da força centrípeta na direção do vetor velocidade.

Se um vetor a_b é representativo no referencial do dispositivo (local ou corpo) da aceleração e quisermos obter a velocidade e posição no referencial da terra, então teremos de usar a matriz de rotação para efetuar os cálculos. Visto que o acelerómetro mede a aceleração física do sensor assim como a força gravítica, a força da gravidade deverá ser primeiro retirada. O vetor aceleração pode ser modelado conforme a equação (27).

$$a_m = a_b - C_n^b \cdot \begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix}, \quad (27)$$

onde a_b é a aceleração medida pelo sensor no referencial local, g é a aceleração da gravidade no referencial da terra e C_n^b é a matriz de rotação que permite transformar um vetor no referencial da terra para o referencial local. Este modelo obviamente assume que não existem erros de *offset* ou escala. Para que seja possível estimar a velocidade e posição no referencial da Terra, necessitamos de remover a força gravítica. Desta forma, resolvendo para o referencial do dispositivo, temos a equação (28).

$$a_b = a_m + C_n^b \cdot \begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix} \quad (28)$$

a qual de seguida deverá ser transformada para o referencial da Terra resolvendo a equação (29) para que seja possível integrar para se obter a estimativa da velocidade e da posição.

$$a_n = C_n^{b^{-1}} \cdot a_m \quad (29)$$

Visto que C_n^b é ortogonal, a sua inversa é a transposta. Assim se consegue obter a aceleração linear, ou seja, sem a força da gravidade.

3 Orientação e referenciais de navegação

Assim que a aceleração no referencial global é obtido, esta é integrada para se obter a velocidade e integrada novamente para se obter a posição conforme se pode constatar na equação (30).

$$V_n = \int a_n \text{ e } P_n = \iint a_n \quad (30)$$

onde V_n é o equivalente ao vetor velocidade e P_n o equivalente ao vetor posição. Na prática, os dados são obtidos em intervalos de tempo discretos pelo que a velocidade e a posição são estimadas usando as equações (31) e (32).

$$V_n(k+1) = V_n(k) + a_n \cdot T(k) \text{ e} \quad (31)$$

$$P_n(k+1) = P_n(k) + V_n \cdot T(k) \quad (32)$$

onde T é o período da amostra (intervalo de tempo). Este período pode não ser constante e logo deve ser também medido para o cálculo das estimativas.

Se o INS for usado durante longos períodos de tempo sem o auxílio de um sensor de posicionamento absoluto, então teríamos ainda que ter em conta a aceleração Coriolis, causada pela rotação da Terra, pois o veículo está a deslocar-se sobre um referencial em rotação. Neste projeto não se teve em conta esta aceleração que teria de ser calculada com rotações provenientes de sistemas de referência da Terra, pois existirão correções em curtos espaços de tempo sobre a velocidade e direção com o auxílio do GPS.

Também o cálculo da força da gravidade será objeto de análise mais detalhado, visto que o mesmo oferece dificuldades na sua obtenção com o dispositivo em movimento, derivado sobretudo à baixa qualidade dos sensores e também à elevada dependência do sensor acelerómetro.

4 Desenvolvimento de aplicações para dispositivos móveis

No desenvolvimento de aplicações para dispositivos móveis, algumas características técnicas dos dispositivos e plataformas terão de ser consideradas para auxiliar e guiar o programador na obtenção do objetivo final. Essas características são o tamanho e resolução do ecrã, capacidades de memória e processamento, métodos de entrada de dados, conectividade e limitações de autonomia de bateria, esta última vista mais detalhadamente no sistema de navegação Ctrack referido na subsecção 2.6.4 deste documento.

Atualmente as plataformas ou *frameworks* mais utilizadas no desenvolvimento para dispositivos móveis são:

- Windows Phone 7 (sucessor do Windows Mobile);
- Android;
- Apple iOS;
- Nokia Symbian OS;
- RIM (BlackBerry OS);

A plataforma Windows Mobile foi desenvolvida pela Microsoft para dispositivos móveis. O seu desenvolvimento inicial foi baseado no *Kernel*³⁰ do Windows CE e as linguagens de programação subjacentes ao desenvolvimento de aplicações para esta plataforma é o *.NET Compact Framework C#* e *VB.Net*. Foi bastante difundida por entre aplicações corporativas pela sua produtividade e capacidade de integração com outros produtos Microsoft. O seu sucessor é o *Windows Phone 7* mais focado no mercado do consumidor final. O *Windows Phone 7* suporta duas tecnologias, o *Silverlight* e o *XNA* cobrindo as necessidades dos utilizadores, visto que a primeira oferece um ambiente aplicacional rico integrado e a segunda

³⁰ *Kernel* é referido como o componente principal dos sistemas operativos de computadores. Serve de ponte entre aplicações e o processamento real dos dados efetuados ao nível do *hardware*.

um ambiente de desenvolvimento para jogos. O lançamento do *Windows Phone 7* no início de 2011 deu um novo fôlego nas vendas deste tipo de plataforma para dispositivos móveis. No segundo trimestre de 2013, as vendas situavam-se em 3.3% de fatia de mercado (Gartner, 2013).

A Apple também possui o seu próprio sistema operativo para dispositivos móveis, o *iOS*, usado no *iPhone*, *iPod Touch*, *iPad* e *Apple Tv*. A Apple não permite que o sistema operativo seja executado em *hardware* de terceiros. A sua arquitetura é bastante similar à arquitetura do *Mac OS X*. Ao nível mais alto, o *iOS* atua como intermediário entre o *hardware* e as aplicações. As aplicações criadas, raramente comunicam diretamente com o *hardware*. Esta abstração facilita o desenvolvimento de aplicações para poderem funcionar de forma consistente em *hardwares* com capacidades diferentes. A linguagem adotada para desenvolvimento de aplicações para esta plataforma é o *Objective-C*, uma extensão da linguagem C. No segundo trimestre de 2013, a fatia de mercado do *iOS* era de 14.2%, apresentando um recuo face ao período homólogo de 2012 (Gartner, 2013).

O sistema operativo Symbian surgiu a partir da evolução de um sistema operativo desenvolvido pela empresa *Psion*. Durante a sua evolução, a equipa de desenvolvimento baseou-se no paradigma de programação orientada a objetos e assim nasceu o *EPOC* para dispositivos móveis. Este sistema operativo tornou-se na base para vários dispositivos móveis de diversas empresas do mundo, de onde surgiu a união de grandes nomes da indústria tais como Nokia, Motorola e Sony-Ericsson. Mais tarde, formou-se um grupo chamado *Symbian* e após a extinção do *EPOC*, surgiu o *Symbian OS* a dar continuidade ao projeto. O desenvolvimento de aplicações *Symbian* pode ser efetuado recorrendo à linguagem de programação C++ ou Java. O *Symbian*, no segundo trimestre de 2013, ainda detinha uma posição de 0.3% de fatia de mercado, continuando a ser a plataforma que mais perdeu relativamente aos períodos homólogos de 2010, 2011 e 2012 (Gartner, 2013).

O sistema operativo *BlackBerry* é um sistema proprietário desenvolvido pela *RIM (Research In Motion)* para a linha de produtos *Blackberry*. O sistema foi desenvolvido em C++ e suporta aplicações desenvolvidas em Java. O sistema ganhou fama no suporte que dava com a ferramenta de receção automática de e-mail em 1999. No segundo trimestre de 2013 a sua fatia de mercado era de 2.7% (Gartner, 2013).

A plataforma *Android* foi inicialmente desenvolvida pela *Android Inc.*, financeiramente apoiada pela *Google* e foi revelada em 2007 juntamente com a criação da *OHA (Open Handset Alliance)* de forma a padronizar uma plataforma de código aberto e livre. Tem como base o *Kernel Linux 2.6* onde várias alterações foram efetuadas para que o sistema se adaptasse à realidade dos dispositivos móveis. O *SDK (Software Development Kit)* disponibiliza ferramentas de desenvolvimento em Java e um emulador que permite correr e testar as aplicações ainda em ambiente de desenvolvimento. É uma das plataformas que maior crescimento apresentou no segundo trimestre de 2013, com uma fatia de mercado na ordem dos 79% face aos 64.2% em período homólogo de 2012 (Gartner, 2013).

Para o desenvolvimento da aplicação, objeto de estudo desta tese, foi escolhida a plataforma *Android*. A razão da escolha prende-se sobretudo pela abrangência apresentada no mercado, visto que apresentava a maior fatia de mercado dos sistemas operativos usados em dispositivos móveis no segundo trimestre de 2013 conforme pode ser constatado na figura 25. Apesar disto, esta abrangência pode ser inconveniente no desenvolvimento da solução, tendo em conta a grande variedade de dispositivos móveis com sensores de diferentes gamas,

tornando a modelação do sistema mais genérica. Outra razão fundamental para a escolha da plataforma Android, tem a ver com a particularidade desta ser de código aberto, ou seja, o seu desenvolvimento é facilitado pela grande disponibilidade de informação e partilha de conhecimento pela comunidade e para além disto, os custos com o desenvolvimento são bastante diminutos, visto que não estão a ser tidas em conta questões de licenciamento.

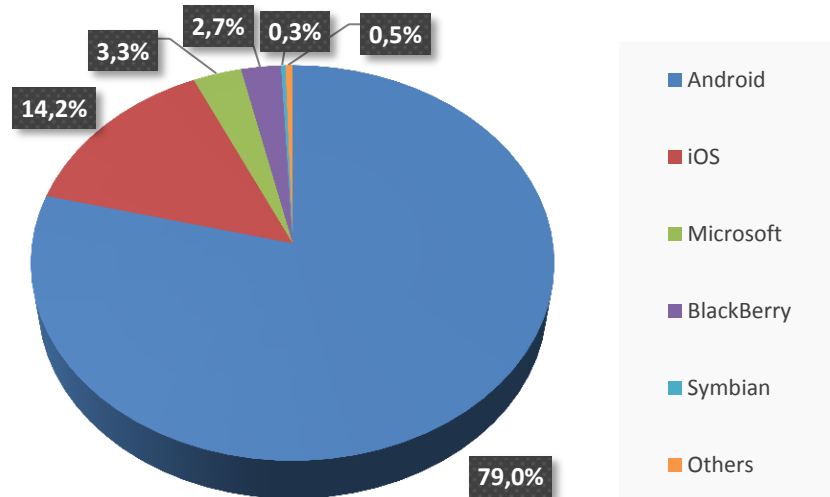


Figura 25 - Venda de Smartphones por sistema operativo no segundo trimestre de 2013. (Gartner, 2013)

4.1 Plataforma Android

A OHA foi criada de forma a padronizar uma plataforma de código aberto e livre para dispositivos móveis, tendo como principal objetivo atender a todas as expectativas e tendências do mercado. O Android foi o resultado da união de diversas empresas de tecnologia e mobilidade entre as quais a Google que se posicionava à frente desta aliança na altura do lançamento do primeiro SDK em 2007.

Os utilizadores dos dispositivos móveis buscam diversas funcionalidades para além da realização de chamadas tais como câmara fotográfica, gravação de voz e vídeo, jogos, acesso à Internet entre outras. Para além dos utilizadores comuns, também as empresas buscam agilidade e cada vez mais concentram-se no desenvolvimento de aplicações para as suas operações diárias.

4.1.1 Arquitetura

A arquitetura da plataforma Android é dividida em várias camadas (*software stack*). Cada camada agrupa vários programas que suportam funções específicas do sistema operativo conforme se pode constatar na figura 26. A camada mais baixa da arquitetura, chamada de

4 Desenvolvimento de aplicações para dispositivos móveis

Linux Kernel, inclui os programas de gestão de memória, configurações de segurança, gestão de processos e *threads*³¹, *software* de gestão de energia e *drivers* de *hardware*.

Na camada imediatamente a seguir são incluídas as bibliotecas³² do Android. Estas bibliotecas estão escritas em C/C++ e são responsáveis pela estabilidade e performance dos diversos componentes. Ao mesmo nível desta camada existe a camada *Android Runtime* que inclui um conjunto de bibliotecas Java. A base de desenvolvimento para Android é Java, porém não existe uma JVM (*Java Virtual Machine*), mas uma outra máquina virtual, a Dalvik³³, otimizada para interpretar código em dispositivos móveis. Todas as aplicações Android correm no seu próprio processo com a sua própria instância de máquina virtual Dalvik. Ao compilar o *bytecode* (.class), este é convertido para o formato específico da máquina virtual Dalvik, o *.dex* (*Dalvik Executable*), que representa a aplicação compilada. Este é um aspeto interessante visto que nenhuma aplicação é dependente de outra e torna o processo de gestão de memória simplificado.

A camada seguinte, chamada de *Application Framework*, inclui as aplicações que gerem as funções básicas do dispositivo móvel, tais como alocação de recursos e aplicações do dispositivo. Os programadores têm acesso total ao *Application Framework* de forma a tirar vantagem de todas as capacidades de processamento do Android.

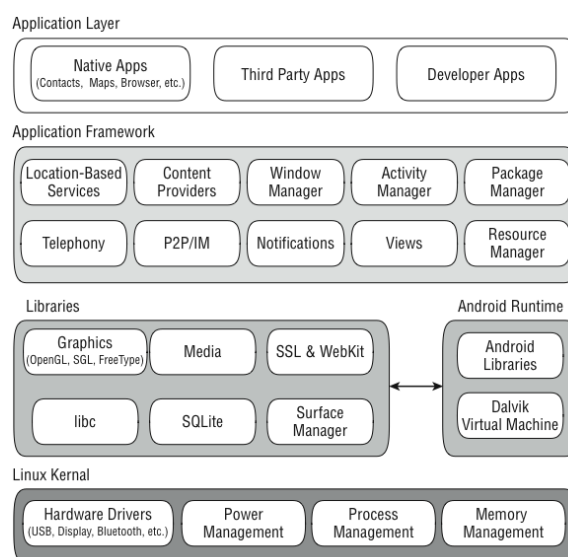


Figura 26 - Arquitetura Android. (Perez, s.d.)

³¹ Um *thread* é uma sequência de instruções que vão ser executadas num programa. No ambiente UNIX, os *threads* encontram-se dentro de um processo, utilizando os recursos desse processo. Um processo pode ter vários *threads*. Pode-se dizer que um *thread* é um procedimento que é executado dentro de um processo de uma forma independente. (Sousa, 2006)

³² As bibliotecas podem ser definidas como um conjunto de instruções que indicam a forma como o dispositivo deve lidar com os diferentes tipos de dados.

³³ Dalvik é a máquina virtual de Java que corre sobre o Kernel de GNU/Linux nos dispositivos com o sistema operativo Android. O nome Dalvik provém do seu criador original, Dan Bornstein, que o nomeou de acordo com uma vila da Islândia de nome Dalvík.

Por fim, na camada de topo, encontram-se as aplicações em si. São aplicações responsáveis pelas funções básicas do dispositivo tais como efetuar chamadas telefónicas, *browser*, lista de contactos e outras.

4.1.2 Desenvolvimento para Android

Como inicialmente referido, o desenvolvimento da aplicação vai ser efetuada recorrendo ao Java como linguagem de programação e ao SDK disponibilizado a programadores para desenvolvimento para Android. O SDK dá ao programador acesso à interface de programação do Android (API) e ainda disponibiliza um conjunto de ferramentas de desenvolvimento, código fonte, um emulador e as bibliotecas necessárias para compilar aplicações para Android.

Uma aplicação Android não segue o formato conhecido de uma aplicação Java comum. Uma aplicação Android é dividida em 4 blocos de desenvolvimento:

- *Activity*: De uma forma geral, cada atividade refere-se a um ecrã apresentado ao utilizador e é implementada como uma única classe que deriva da classe base *Activity*. Esta classe exibe uma interface com o utilizador e trata os eventos a ela associados;
- *Services*: É um componente aplicacional que pode correr tarefas morosas em *background* ou executar processos remotos. Um *service* não possui interface de utilizador;
- *Content Provider*: Permite que as aplicações compartilhem informações entre si;
- *Broadcast Receiver*: Um *broadcast receiver* (recetor de notificações) é um componente que responde a notificações ao nível do sistema. Embora estes *broadcast receivers* não definam uma interface com o utilizador, estes podem gerar notificações na barra de estado para alertar o utilizador acerca de um evento.

A classe *Activity* é a mais importante no ciclo de vida de uma aplicação e a forma como as atividades são lançadas e interligadas é parte fundamental do modelo da aplicação. As atividades no sistema são geridas como uma pilha de atividades. Quando uma atividade é iniciada, ela é colocada no topo da pilha e torna-se a atividade atual, já a atividade anterior, permanece abaixo na pilha e não é mostrada enquanto a atividade atual não terminar.

Os métodos de uma atividade, conforme demonstrado na figura 27, introduzem implicitamente os estados que uma atividade pode assumir ao longo da sua vida:

- Se uma atividade está a ser executada e mostrada no ecrã (topo da pilha), então essa atividade está em modo *Active* ou *Running*;
- Se uma atividade perder o foco mas continuar visível (outra atividade está a correr e a ser mostrada no ecrã mas não a ocupar totalmente) ela encontra-se num estado *Paused*. Apesar de estar em *Paused*, a atividade continua viva mas pode ser encerrada pelo sistema em caso de memória baixa por exemplo;
- Se uma atividade ficar completamente invisível porque entretanto outra atividade foi lançada, a atividade fica em modo *Stopped*. O seu estado e informação (do objecto

Activity) ficam retidos na memória, mas pode ser encerrada quando for necessário libertar memória.

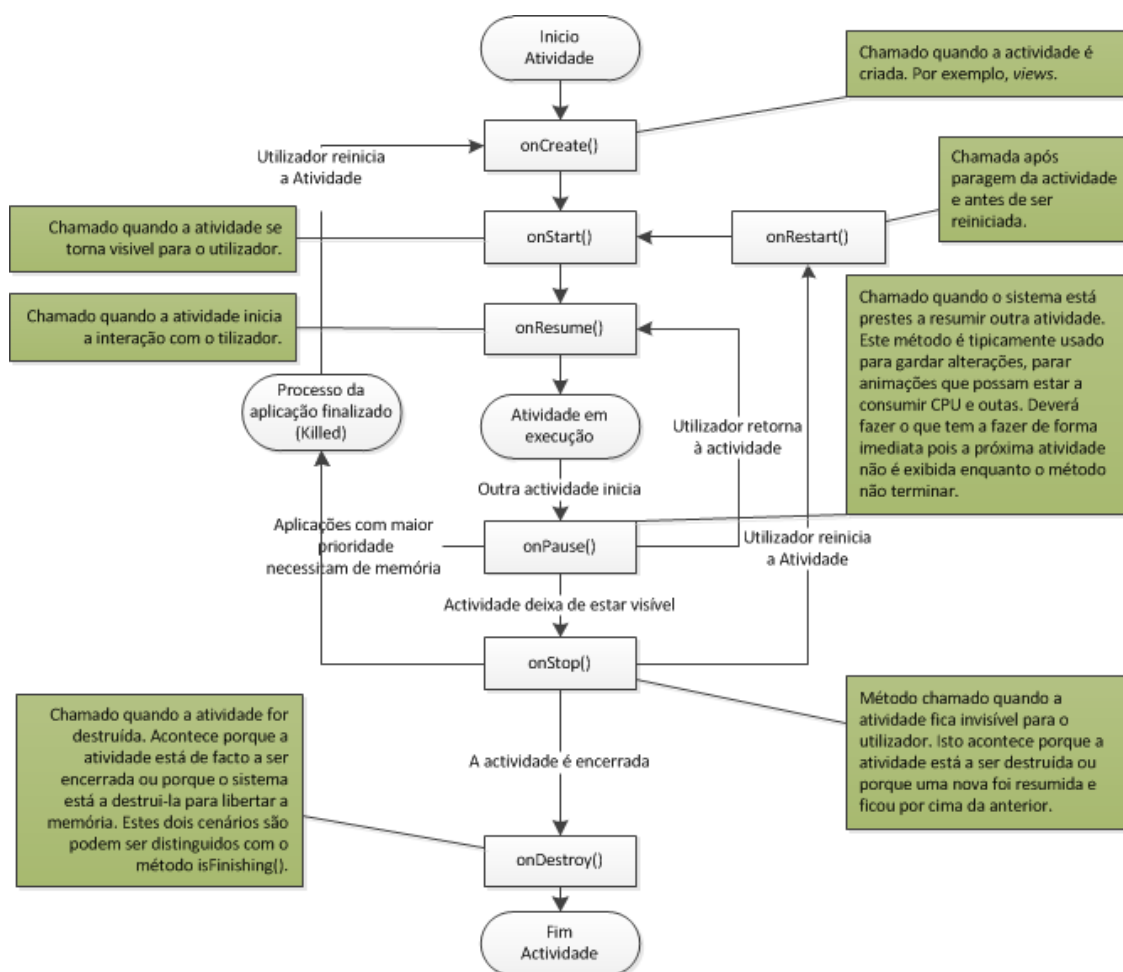


Figura 27 - Ciclo de vida de uma atividade. (Adaptado de Open Source Project, 2012)

4.1.3 Sensores nos dispositivos móveis

É importante entender a forma como podemos interagir com os sensores presentes na maioria dos dispositivos móveis visto que neste trabalho iremos fazer uso de um giroscópio, de um acelerómetro e de um magnetómetro para tentar alcançar os objetivos propostos. Um programador deverá ir para além da típica “caixa negra” onde os dados dos sensores são digeridos por uma aplicação sem compreender o que os dados representam. O Android fornece um grande número de classes e métodos na sua API tornando possível aos programadores acederem à vasta gama de sensores no desenvolvimento de aplicações.

Os sensores referenciados pela classe *Sensor* podem ser de dois tipos: Sensores reais ou sensores virtuais (compostos). Sensores reais fornecem dados reais ou brutos de um sensor dentro do dispositivo Android. Os sensores virtuais (e.g., *Sensor.TYPE_LINEAR_ACCELERATION*) providenciam uma camada de abstração entre o código da aplicação e os componentes do dispositivo combinando dados reais de múltiplos sensores reais ou modificando os dados reais de forma a tornar o seu consumo mais eficiente. Estes

podem reportar uma quantidade física referindo-se a dois ou a três sensores reais (*e.g.*, sensor de orientação que se refere à fusão dos dados do magnetómetro e acelerómetro).

A plataforma Android suporta diversos tipos de sensores e na tabela 3 especifica-se apenas os que irão ser usados de forma direta ou indireta na análise e desenvolvimento da aplicação.

Tabela 3 - Sensores utilizados no desenvolvimento da aplicação.

Sensor	Tipo	Descrição
TYPE_ACCELEROMETER	<i>Hardware</i>	Acelerómetro mede a força de aceleração em m/s^2 que é aplicada no dispositivo em todos os três eixos (X,Y e Z), incluindo a força da gravidade.
TYPE_GRAVITY	<i>Virtual</i>	Derivado do acelerómetro por um filtro passa baixo ou em conjunto com o giroscópio. O Android disponibiliza um método de fusão sensorial cuja gravidade é retirada da matriz rotação proveniente da fusão do acelerómetro com o giroscópio.
TYPE_GYROSCOPE	<i>Hardware</i>	Mede a velocidade de rotação em rad/s em torno de cada um dos três eixos.
TYPE_LINEAR_ACCELERATION	<i>Virtual</i>	Derivado do acelerómetro e mede a força de aceleração em m/s^2 que é aplicada em cada um dos três eixos, excluindo a força da gravidade.
TYPE_MAGNETIC_FIELD	<i>Hardware</i>	Mede o campo magnético de cada um dos três eixos em μT (Micro Teslas).
TYPE_ROTATION_VECTOR	<i>Virtual</i>	Indica a orientação do dispositivo ao providenciar os três elementos do vetor de rotação do dispositivo.

A API do Android, no caso de processamento com sensores, consiste em classes e interfaces para solicitar e processar dados do *hardware* de um dispositivo. O ponto de entrada da API para os sensores é a classe *SensorManager* que permite uma aplicação registar o sensor e requisitar atualizações da informação. Quando registado, os valores dos sensores são enviados a um *SensorEventListener* (interface) na forma de um evento do tipo *SensorEvent* que contém a informação produzida por um sensor. O *SensorManager* é um serviço de sistema Android que como outros serviços de sistema, deixa uma aplicação registar e cancelar registos de eventos relacionados com os sensores. Uma vez registado, a aplicação recebe eventos do *hardware*. O *SensorManager* contém algumas constantes representativas de diversos aspetos do sistema de sensores no Android, tais como:

- Tipo de sensor: Orientação, acelerómetro, iluminação, magnetómetro, proximidade, temperatura, giroscópio, etc;
- Taxa de amostragem ou frequência: A frequência de obtenção de valores é pré-definida de acordo com a escolha de uma das quatro frequências disponíveis (rápida, jogo, normal e interface de utilizador) ou definida manualmente. Quando definimos

uma delas, estamos apenas a sugerir a frequência ao subsistema de sensores visto que o sistema apenas envia os dados assim que tiver disponibilidade;

- Precisão (*Accuracy*): Alta, baixa, média e incerto.

A interface *SensorEventListener* é implementada por uma classe que recebe atualizações dos sensores (*SensorManager*) à medida que os valores se alteram em tempo real. Uma aplicação implementa esta interface para monitorizar um ou mais sensores disponíveis no *hardware*.

Uma instância do *SensorManager* pode ser obtida com o método *getSystemService()* conforme o *snippet* abaixo representado:

```
String service_name = Context.SENSOR_SERVICE;  
SensorManager sensorManager =  
(SensorManager) getSystemService(service_name);
```

A interface *SensorEventListener* é implementada como base para as aplicações que usam sensores. É usada para receber notificações do *SensorManager* quando os valores dos sensores são alterados. São incluídos dois métodos necessários aquando da instanciação:

- *onAccuracyChanged()*: Este método é invocado sempre que a precisão de um sensor é alterada. Como argumentos temos a instância do sensor (representativa do tipo de sensor) e um inteiro que representa a nova precisão do sensor.
- *onSensorChanged()*: É invocado sempre que um valor do sensor é alterado. Tem como único argumento o objeto *SensorEvent* que contém informação sobre o tipo de sensor, um *time-stamp*, a precisão e os dados do sensor.

Para ser possível interagir com o sensor, a aplicação necessita de implementar o método *RegisterListener()* da classe *SensorManager* para a qual devemos especificar o *delay* ou taxa de amostragem (*sensor rate*). Na versão Android 4.0.3, estes *delays* são *hard-coded*³⁴ como 0 milissegundos (*SENSOR_DELAY_FASTEST*), 20 milissegundos (*SENSOR_DELAY_GAME*), 67 milissegundos (*SENSOR_DELAY_UI*) e 200 milissegundos (*SENSOR_DELAY_NORMAL*). Contudo é também possível especificarmos o nosso próprio *delay* em milissegundos. De toda a forma, estes *delays* apenas têm a intenção de sugerir ao sistema, pois os eventos poderão ser despoletados mais ou menos rápido do que o especificado conforme disponibilidade do sistema.

Um aspeto relativo aos *delays* é a importância do campo *SensorEvent.timestamp*, pois os valores não são obtidos em espaços de tempo iguais, permitindo ao programador ter um maior controlo sobre os mesmos.

4.1.4 Deteção do movimento e orientação do dispositivo com Android

Parte integrante deste projeto, é a deteção do movimento e orientação do dispositivo. Conforme referido no capítulo anterior, existem processos de transformação de coordenadas

³⁴ É uma referência à técnica de criação de variáveis com dados pré-definidos ao invés de obter pelo processamento de fontes externas.

que teremos de ter em conta para obtermos a velocidade no sistema de coordenadas correto. A plataforma Android auxilia neste aspeto com a definição de classes e interfaces que contêm métodos específicos na obtenção de matrizes rotação, obtenção da orientação e outros aspetos relevantes.

No uso de sensores para determinação do movimento e orientação no Android, dois sistemas de coordenadas são definidos como é o caso do sistema de coordenadas global (ENU) ou de navegação (x_n, y_n, z_n) e o sistema de coordenadas do dispositivo (x_b, y_b, z_b) . Ambos os sistemas estão ilustrados na figura 28 que mostra o dispositivo móvel posicionado no Equador com alguma inclinação respeitante à Terra.

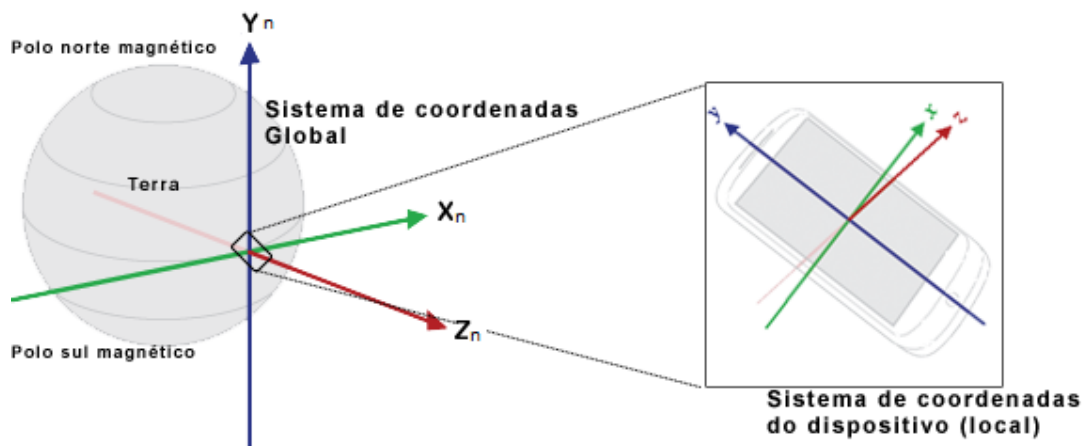


Figura 28 - Sistemas de coordenadas do Android. (Adaptado de Open Source Project, 2012)

Todos os sensores que se referem à orientação absoluta em respeito à Terra (exceto o sensor de orientação), usam o sistema de coordenadas global. Os seguintes sensores e métodos explicam essa referência:

- O Sensor *TYPE_ROTATION_VECTOR* ou vetor de rotação, sensor virtual que faz uso dos dados de um acelerómetro, magnetómetro e de um giroscópio (quando presente), efetuando cálculos de forma a providenciar informação combinada mais precisa;
- Os métodos *getRotationMatrix()*, *getRotationMatrixFromVector()* e *getQuaternionFromVector()*, que permitem obter a matriz rotação ou o quaternião de forma a efetuar o mapeamento do sistema de coordenadas do dispositivo para o sistema de coordenadas global ou a operação inversa usando a transposta da matriz;
- O método *getOrientation()*, que com a matriz de rotação como argumento, devolve um vetor de orientação (*Roll*, *pitch* e *Yaw*);
- O método *getInclination()*, que com o vetor orientação gerado pelo método *getOrientation()*, retorna a inclinação magnética³⁵ (podemos comparar este método ao uso de um pêndulo na determinação dos ângulos *pitch* e *roll* e conjugação com o magnetómetro para determinação do ângulo *yaw*).

³⁵ A inclinação magnética diz respeito à variação de uma agulha de um compasso e seu desvio relativo ao plano horizontal da superfície da Terra.

4 Desenvolvimento de aplicações para dispositivos móveis

No sistema de coordenadas global, o y_n aponta na direção do norte magnético, aproximadamente o verdadeiro norte (polo geográfico norte). O x_n aponta aproximadamente para Este, paralelo à superfície da Terra, mas a 90 graus de y_n . O z_n aponta para fora do centro da Terra.

Os sensores acelerómetro, magnetómetro e giroscópio reportam valores correspondentes ao sistema de coordenadas do dispositivo, muitas vezes referido como sistema de coordenadas local. O sistema de coordenadas local é parcialmente definido pela orientação por defeito que difere conforme o tipo de dispositivo (*portrait ou landscape orientation*). Na sua orientação por defeito, (ver figura 29), os eixos são direcionados para:

- O eixo X é horizontal com valores positivos para a sua direita;
- O eixo Y é vertical com valores positivos para a frente;
- O eixo Z com valores positivos para fora do ecrã.

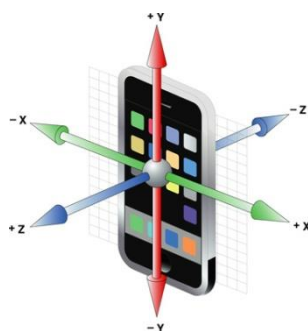


Figura 29 - Definição dos eixos no Android.

Este sistema de coordenadas (sistema de coordenadas local) está fixo ao dispositivo.

A partir da API 9 do Android, foram introduzidas alterações relativamente aos sinais dos sensores. A maior alteração foi a introdução de métodos de fusão de sensores com o uso de sensores virtuais tais como *TYPE_GRAVITY*, *TYPE_ROTATION_VECTOR* (apenas em dispositivos com giroscópio) e obtenção de orientação com o método *getOrientation()*. Inclusive para dispositivos com giroscópio, foi introduzido um novo sensor virtual (*Corrected Gyroscope Sensor*) que proporciona leituras corrigidas suprimindo muito do *drift* que lhe é habitual fazendo uso de um filtro de Kalman quando correlacionado com os valores do acelerómetro.

5 Aquisição e análise de dados dos sensores de um dispositivo móvel

O sistema móvel de telemetria para automóveis pretende ser um auxiliar ao sistema GNSS existente de forma a fazer uso das principais características positivas de uma IMU, que é a elevada taxa de obtenção de dados e o facto de não depender de fatores externos para obter dados. No entanto e visto que esses dados são provenientes de sensores MEMS de baixo custo, os valores podem não ser perfeitos.

O dispositivo utilizado na análise dos registos dos sensores foi um *smartphone* LG 2X (ver figura 30). Este dispositivo tem todos os sensores necessários para a realização deste trabalho, nomeadamente os sensores giroscópio, acelerómetro e magnetómetro. A versão do Android deste dispositivo é a 4.0.4 conhecido pelo nome de código “*Ice Cream Sandwich*”. É uma versão *upgrade* da versão original do dispositivo (2.2 – *Froyo*) e é relevante frisar esta característica pois o tratamento de alguns sinais provenientes dos sensores foi tido em conta na evolução do sistema operativo no que toca ao pré-processamento dos *outputs*. A versão 4.0.4, já apresenta algoritmos para fusão de alguns sensores. Convém referir que apesar de as especificações dos dispositivos variarem bastantes, nomeadamente os sensores MEMS no que toca à resolução, precisão, exatidão e níveis de ruído, o sistema deverá automaticamente conciliar-se existindo uma fase de adaptação (calibração e alinhamento) inicial que deverá ser sempre efetuada. Esta fase de calibração irá fazer com que os sensores virtuais, tais como o *TYPE_ROTATION_VECTOR*, não sejam usados, pois os mesmos fazem uso de uma fusão baseada em valores não calibrados.



Características principais	
S. Operativo	Android OS 4.0.3 (Ice Cream Sandwich)
Chipset	Nvidia Tegra 2 AP20H
CPU	Dual-core 1 GHz Cortex-A9
GPU	ULP GeForce
Sensores	Acelerómetro, giroscópio, proximidade e magnetómetro.

Figura 30 - Smartphone LG 2X P990 e suas características principais.

5.1 Processamento de sinais e erros dos sensores

Os sensores produzem registos por vezes incorretos devido ao ruído ou degradação que ocorre ao longo do tempo. Para abordar estes erros (descritos na secção 2.3.), algoritmos e técnicas são implementadas. A forma de abordagem varia consoante a precisão, velocidade de receção e destino que queremos dar aos nossos dados. É possível filtrar o *output* dos sensores ou fundi-los com outros sensores para melhorar a precisão. A plataforma Android faz uso destas técnicas na disponibilização de sensores virtuais como no caso do Acelerómetro Linear (*TYPE_LINEAR_ACCELERATION*) ou o mais recente processo de obtenção dos dados pré-processados do giroscópio com a fusão do acelerómetro e magnetómetro através de um filtro de Kalman (*TYPE_ROTATION_VECTOR*).

Visto que no desenvolvimento irão ser usadas técnicas de tratamento de erros, convém descrever resumidamente os mesmos no âmbito da plataforma Android.

Os erros sistemáticos são erros que afetam a exatidão de uma medida pois são constantes deslocamentos (*offsets ou bias*) do valor verdadeiro. Em alguns casos estes podem ser previsíveis ou removidos através da calibração.

Já os erros aleatórios tal como o ruído no qual resulta imprecisão na medida, não pode ser removido por esta técnica. O ruído é uma flutuação aleatória de um valor medido e uma das técnicas existentes para mitigar estes ruídos é usar filtros do tipo passa baixo (*low-pass filter*).

Outro tipo de erro que será encarado é a deriva de registos (*Drift Error*) que descreve a forma errante dos dados a longo prazo afastados dos valores reais. Geralmente acontece pois o output do sensor degrada-se ao longo do tempo e os valores integrados causam um pequeno *offset* que é adicionado em cada iteração da integração.

Um erro relatado já neste trabalho é o erro *Zero Offset* ou *Bias* que define um sinal de output diferente de zero quando a propriedade medida é zero. Quando isto acontece, diz-se que o sensor tem um *offset* ou *bias* (desvio tendencioso). Uma boa forma de se obter este valor é colocar o dispositivo deitado numa superfície paralela ao chão e medir a média das observações do acelerómetro do dispositivo e caso os valores não sejam aproximadamente $(0,0,-9,80665 \text{ m/s}^2)$ nos eixos X,Y e Z respetivamente, então o acelerómetro tem um *offset*. No caso de um giroscópio, este deverá obter valores de $(0,0,0)$ com o dispositivo completamente estabilizado pois apenas um pequeno *offset* na integração será suficiente para se obter um erro, visto que para se obter o ângulo, temos de integrar os valores devolvidos pelo sensor.

Um outro aspeto a ter em conta no uso dos sensores de um dispositivo Android é o facto de o próprio Android não ser um RTOS³⁶ (*Real-time Operating System*) o que poderá levar a que os dados provenientes das medições sofram um atraso, resultando em *timestamps* irregulares.

Para a análise dos sensores foi desenvolvida uma aplicação que guarda os registos no cartão de dados do smartphone e uma outra aplicação em MATLAB para proceder à análise dos mesmos. No caso do estudo da aceleração e velocidade foi ainda elaborada uma aplicação que enviasse os dados em tempo real, via rede sem fios, para a aplicação MATLAB.

³⁶ Um RTOS é um sistema operativo destinado à execução de múltiplas tarefas onde o tempo de resposta a um evento (interno ou externo) é pré-definido.

5.2 Análise do acelerómetro

O acelerómetro mede a aceleração nos três eixos em m/s^2 . É o output da aceleração aplicada ao dispositivo, medindo as forças aplicadas no sensor. Esta aceleração é sempre influenciada pela força da gravidade (g) da Terra ($9,80665 m/s^2$). Como resultado, quando o dispositivo está em queda livre em aceleração em direção ao chão, o seu output ideal seria $0 m/s^2$ em todos os eixos. Quando o dispositivo está estabilizado numa plataforma paralela ao chão, com o ecrã virado para cima e estacionário, o output ideal seria $(0,0,g)$ para os eixos X, Y e Z respetivamente. Usa-se esta suposição para uma calibração no processo. Na figura 32, é visível o efeito de ruído que torna a calibração do sensor mais difícil em todos os eixos, visto que dificilmente se consegue estabelecer um *offset* ou definir um coeficiente para o fator de escala ou sensibilidade.

A leitura foi efetuada pelo sensor acelerómetro a uma frequência aproximada de 50 Hz durante aproximadamente 38 segundos recolhendo um total de 1921 registos. O dispositivo foi colocado numa mesa, com o ecrã voltado para cima numa tentativa de estabilizar, o mais possível, os dados obtidos dos eixos.

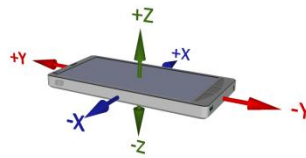


Figura 31 - Constatação do sinal dos eixos do *smartphone* colocado em cima de uma superfície plana para o primeiro cenário.

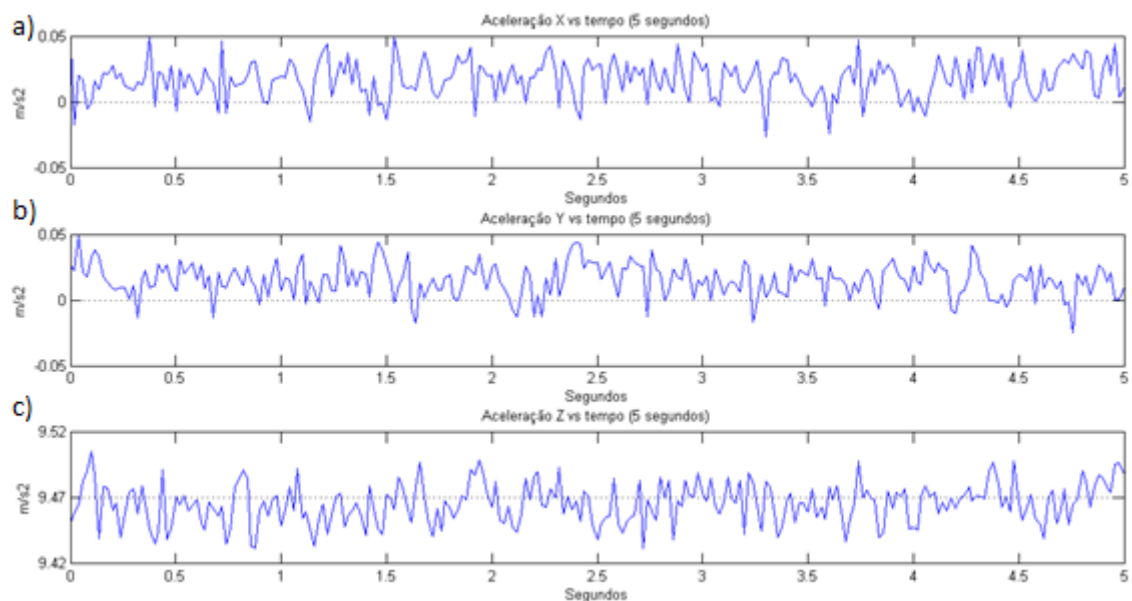


Figura 32 - Output da aceleração para os eixos assinalados como a) x, b) y e c) z do acelerómetro sem tratamento (análise temporal de 5 segundos).

Analisando os dados retiramos algumas conclusões que podemos visualizar na tabela 4.

Tabela 4 - Estatística do output do acelerómetro nos diferentes eixos.

	Ax	Ay	Az	Magnitude
Mínimo	-0.026345	-0.027243	-9.527837	9.527912
Máximo	0.065564	0.064216	-9.417068	9.417515
Média	0.019954	0.015623	-9.471068	9.471102
Desvio	0.013787	0.013019	0.015528	0.015530

Este teste evidencia que a aceleração total medida (proveniente do cálculo da magnitude ou norma das médias dos três eixos) foi de 9,47 m/s², ainda aquém do valor da aceleração da gravidade na Terra (9,81 m/s²). O desvio padrão da magnitude (aceleração total) de todos os registos é de aproximadamente 0,015 m/s² ou 15 mm/s² o que poderá representar, com uma frequência aproximada de 50 Hz (50 registos por segundo), um erro de 0,75 metros em cada segundo. Constatou-se também que o eixo Z apresenta um offset demasiado elevado de aproximadamente 0.37 m/s², valor demasiado elevado mesmo para um dispositivo MEMS de baixo custo e precisão.

5.2.1 Calibração do acelerómetro

No exemplo anterior verificou-se que a aceleração total rondava os 9,47 m/s². Este valor apresentado deveria ficar mais próximo do valor da gravidade de 9,81 m/s² com o dispositivo fixo e estacionário. Existem vários processos mais ou menos precisos para a calibração do acelerómetro, no entanto aqui é apenas referido aquele que menos rigor necessita, ou seja, sem a necessidade de uma plataforma mecânica de precisão para proceder à calibração. Esta calibração não tem em conta o possível desalinhamento do sensor. A aplicação a ser desenvolvida é de uso genérico para utilizadores comuns e pretende-se que o processo de calibração seja o menos complexo e mais automático possível. O modelo de erro para o acelerómetro está representado na equação (33).

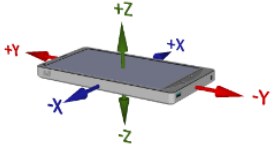
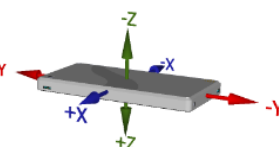
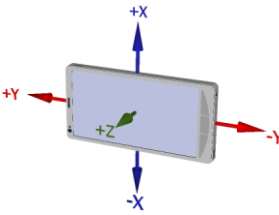
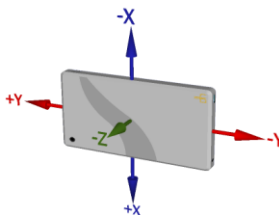
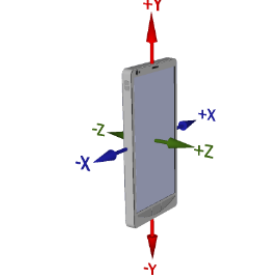
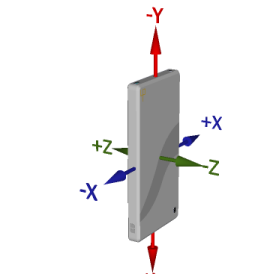
$$\begin{bmatrix} X_{acc} \\ Y_{acc} \\ Z_{acc} \end{bmatrix} = \begin{bmatrix} Sx & 0 & 0 \\ 0 & Sy & 0 \\ 0 & 0 & Sz \end{bmatrix} * \begin{bmatrix} \dot{X}_{acc} \\ \dot{Y}_{acc} \\ \dot{Z}_{acc} \end{bmatrix} + \begin{bmatrix} bx \\ by \\ bz \end{bmatrix} \quad (33)$$

onde X_{acc} , Y_{acc} e Z_{acc} são os valores corrigidos da aceleração dos respetivos eixos, Sx , Sy e Sz são os coeficientes de escala para o ganho e bx , by e bz são os *offsets* ou *bias*. \dot{X}_{acc} , \dot{Y}_{acc} e \dot{Z}_{acc} são os valores de saída do acelerómetro.

Para o sensor acelerómetro usou-se uma calibração de 6 pontos, onde cada eixo do sensor é calibrado separadamente. Para cada eixo, duas médias de observações são efetuadas, uma positiva e outra negativa para a avaliação da aceleração da gravidade.

Colocou-se o dispositivo orientado conforme ilustrado na Tabela 5:

Tabela 5 - Orientações do dispositivo móvel durante o processo de calibração.

	<p>Nivelado, deitado com o ecrã voltado para cima por forma a observar o valor da gravidade no eixo Z (+g)</p>
	<p>Nivelado, deitado com o ecrã voltado para baixo por forma a observar o valor da gravidade no eixo Z (-g)</p>
	<p>Virado para o lado esquerdo por forma a observar o valor da gravidade no eixo X (+g)</p>
	<p>Virado para o lado direito por forma a observar o valor da gravidade no eixo X (-g)</p>
	<p>De pé, apontando a parte superior do dispositivo para cima por forma a observar o valor da gravidade no eixo Y (+g)</p>
	<p>De pé, apontando a parte superior do dispositivo para baixo por forma a observar o valor da gravidade no eixo Y (-g)</p>

Para cada eixo é efetuada uma medição aproximada de 1G e -1G respetivamente tentando de alguma forma orientar o dispositivo com o ângulo mais aproximado possível. Para achar os coeficientes aproximados para uma calibração em cada eixo, recolheram-se amostras de valores de cada eixo respeitando a orientação do dispositivo. Em cada posição, efetuou-se

5 Aquisição e análise de dados dos sensores de um dispositivo móvel

uma média de 500 registos. O processo de calibração implementado na aplicação permite especificar este valor.

As seguintes operações foram codificadas:

mPositivos = A média dos valores mais aproximados de 1G em cada eixo;

mNegativos = A média dos valores mais aproximados de -1G em cada eixo;

G = O valor da gravidade, aproximada de 1G (9,81 m/s²);

Desta forma queremos obter os valores do ganho(S) e *offset*(b) para cada eixo conforme as equações (34) e (35).

$$g = (mPositivos * S) - b \quad (34)$$

$$-g = (mNegativos * S) - b \quad (35)$$

Com $S = 1.0$ e $b = 0$, podemos resolver as equações facilmente para S e b conforme as equações (36) e (37) respetivamente.

$$S = 2 * \frac{g}{mPositivos - mNegativos} \quad (36)$$

$$b = mNegativos * s + g \quad (37)$$

Com os valores obtidos para o fator de escala (S) e *offset* (b) do acelerómetro, efetuaram-se novas medições, desta vez com o acelerómetro calibrado, na expectativa destas observações apresentarem melhores resultados quanto à aceleração total (magnitude) e aproximação do valor da gravidade em cada eixo quando posicionado para o efeito. Foram elaborados 3 cenários diferentes, onde apesar de o dispositivo ter permanecido imobilizado, a orientação do mesmo seguiu a seguinte ordem:

1. Cenário 1: Dispositivo imobilizado com o ecrã virado para cima ($Acc_{xyz} \approx (0,0, g)$);
2. Cenário 2: Dispositivo imobilizado com o ecrã virado para o lado esquerdo ($Acc_{xyz} \approx (g, 0,0)$);
3. Cenário 3: Dispositivo imobilizado num angulo arbitrário;

Estas amostras foram obtidas a uma frequência média de 49 Hz durante aproximadamente 35 segundos. Analisando os dados chegou-se a algumas conclusões com o auxílio da informação da tabela 6.

Tabela 6 - Estatísticas das amostras do acelerómetro nas diferentes posições (1,2 e 3).

Posição	Estatística	Ax	Ay	Az	Magnitude
1	Média	-0.031267	0.010725	9.825786	9.825842
	Desvio	0.014076	0.015216	0.016466	0.016458
2	Média	9.942972	-0.014368	0.077563	9.943285
	Desvio	0.018541	0.014140	0.016926	0.018563

3	Média	2.943895	3.472775	8.672699	9.795019
	Desvio	0.020088	0.018651	0.018917	0.018735

Dos resultados acima evidenciados destacamos uma maior aproximação da aceleração total ao valor real da aceleração da gravidade $9,81 \text{ m/s}^2$, onde aliás pode ser verificado olhando para a figura 33 onde se constata que o eixo Z, para o cenário 1, se aproxima mais do valor real da aceleração da gravidade e os outros eixos X e Y perpendiculares ao Z encontram-se mais próximos de zero. A figura 34 e figura 35 também permitem reconhecer os registos dos diferentes eixos pós calibração para os cenários 2 e 3.

Aliás, tendo em conta os diferentes cenários, a exatidão melhorou bastante e melhor teria ficado se a calibração tivesse decorrido num ambiente controlado com uma plataforma mecânica de calibração.

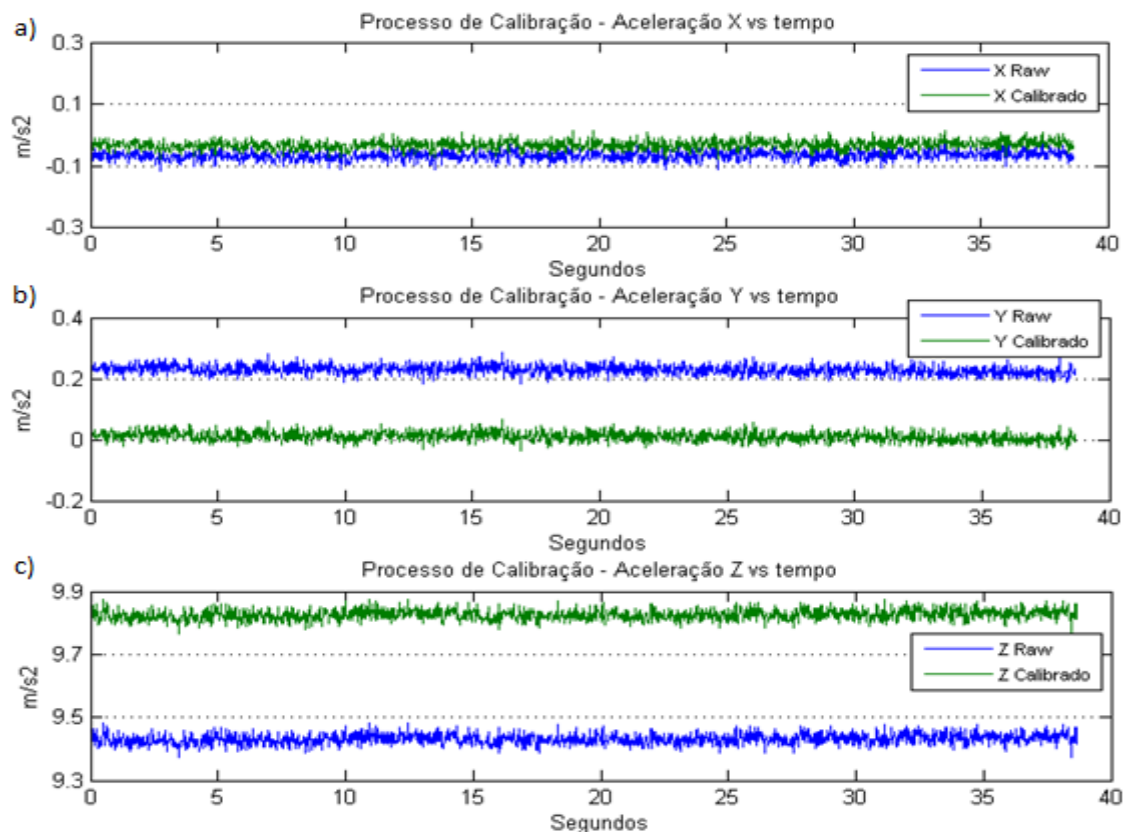


Figura 33 - Comparativo da medição do sensor acelerómetro nos eixos assinalados como a) X , b) Y e c) Z antes da calibração (Raw) e pós calibração (Calibrado) para o cenário 1.

Apesar desta melhoria significativa e importante para o desenvolvimento do projeto, nota-se que o enviesamento da aceleração total proveniente das amostras não melhorou e isto deve-se ao facto de o ruído, causado por diferentes interferências, ainda existir no sinal obtido, isto porque este modelo matemático é apenas uma estimativa do modelo do acelerómetro real pois não tem em conta as não-linearidades, enviesamentos provocados pela temperatura e outros aspetos decorrentes do uso de um sensor do tipo MEMS de baixo custo. Uma forma de atenuar este efeito é recorrer a outro tipo de filtros.

5 Aquisição e análise de dados dos sensores de um dispositivo móvel

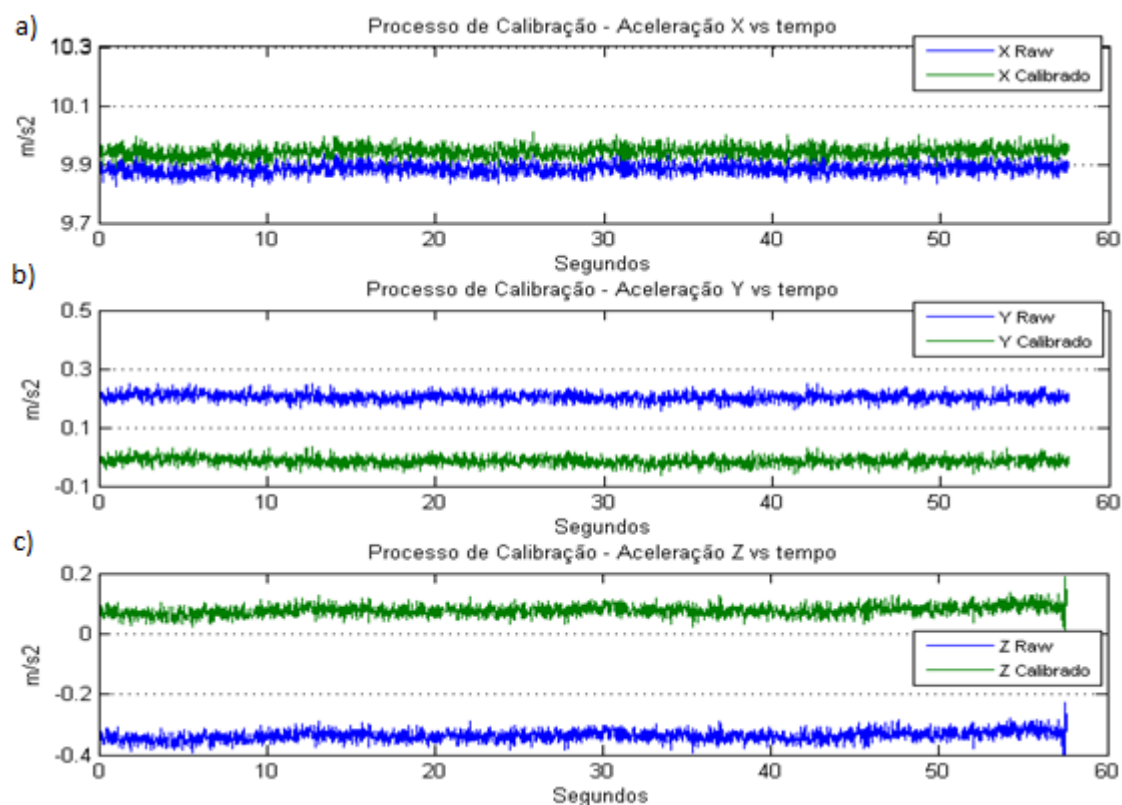


Figura 34 - Comparativo da medição do sensor acelerómetro nos eixos assinalados como a) X, b) Y e c) Z antes da calibração (Raw) e pós calibração (Calibrado) para o cenário 2.

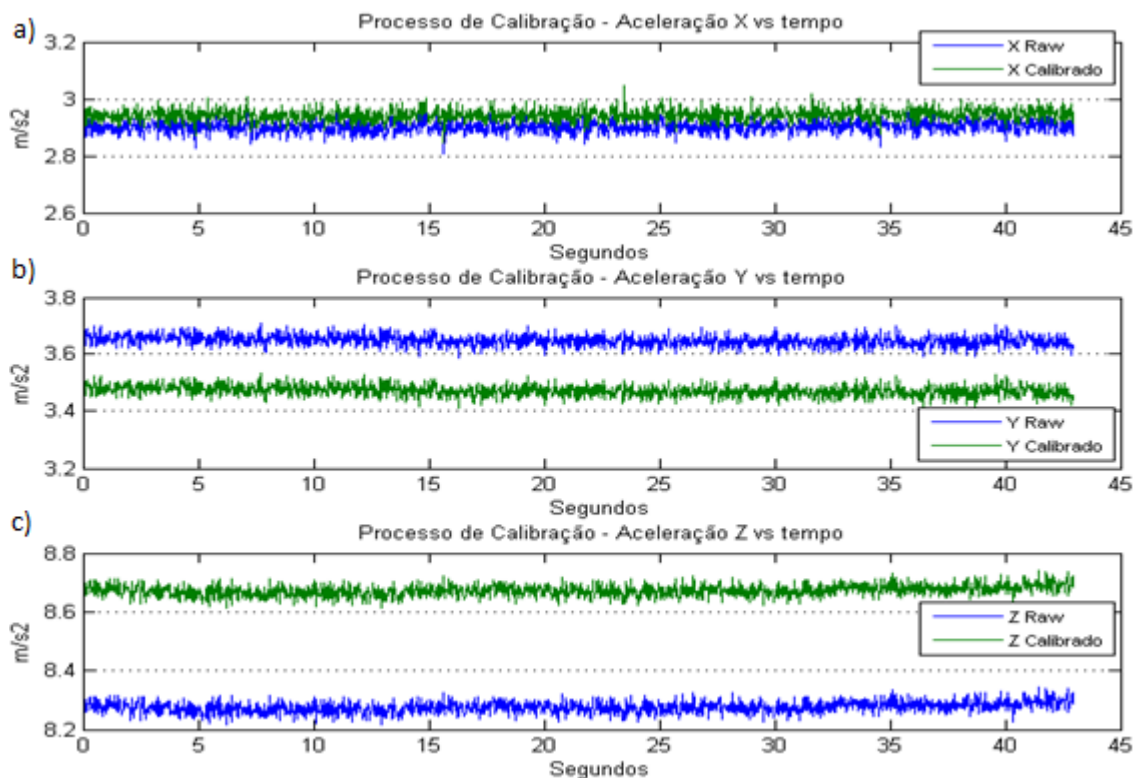


Figura 35 - Comparativo da medição do sensor acelerómetro nos eixos assinalados como a) X, b) Y e c) Z antes da calibração (Raw) e pós calibração (Calibrado) para o cenário 3.

5.2.2 Atenuação do ruído

Os sensores estão sujeitos a ruídos e perturbações aleatórias que podem dificultar os procedimentos da análise e identificação. Para contornar estes problemas tem-se recorrido a utilização de filtros com o fim de se capturar somente os sinais com as dinâmicas e componentes de interesse presentes no sistema. Filtrar um sinal é deixar passar pelo sistema a informação de interesse e bloquear a informação indesejada. Filtros podem ser aplicados não somente para diminuir a influência do ruído do processo, mas também para “suavizar” o resultado da não homogeneidade dos sinais, turbulências ou fluxos não uniformes.

5.2.2.1 Filtro média móvel (*Simple Moving Average*)

Analisaram-se vários filtros no tratamento do ruído nas amostras do acelerómetro. O primeiro filtro aplicado foi o filtro média móvel (*Simple Moving Average*).

Este filtro usa uma janela móvel de n amostras para calcular a média de uma sequência de registos a cada instante de tempo. Faz parte da família de filtros digitais de implementação por convolução³⁷ (resposta ao impulso finita - FIR³⁸) usado no domínio do tempo.

Para a implementação deste filtro consideramos o cálculo de uma média de um número de pontos do sinal de entrada $x[]$, para produzir cada ponto do sinal de saída $y[]$. Ver equação (38) (Smith, 1997).

$$y[i] = \frac{1}{n} \sum_{j=0}^{n-1} x[i + j] \quad (38)$$

onde n é o número de pontos para cálculo da média, i o índice do sinal de saída e j é o índice do somatório que designa o valor inicial e percorre todos os valores até alcançar o limite superior $n - 1$.

Pegando nos cenários anteriores (1,2 e 3) onde o dispositivo se encontra estacionário, aplicou-se o filtro de média móvel com dois argumentos diferentes, aplicando médias a 5 e 10 pontos com o sistema calibrado.

Conforme pode ser constatado na figura 36, notou-se uma significativa suavização dos picos existentes no sinal e quantos mais pontos forem usados para o cálculo da média, mais suavizado fica o sinal. Este filtro apresenta um bom desempenho na diminuição do ruído e consegue preservar o grau de variação do sinal. Poderemos no entanto estar a perder dados para um cálculo minucioso da aceleração total em variação com o tempo, ou seja, em variações de aceleração bruscas, logo terá de ser definido um valor para o número de pontos a serem usados para o cálculo da média que se coadune com a variação do sinal sem se perder informação.

³⁷ Operação matemática de duas funções (f e g) produzindo uma terceira função que é tipicamente vista como uma versão modificada das funções originais.

³⁸ FIR - resposta ao impulso finita é um tipo de filtro digital caracterizado por uma resposta ao impulso que se torna nula após um tempo finito.

5 Aquisição e análise de dados dos sensores de um dispositivo móvel

Além destes pressupostos, convém salientar que este estudo dos sinais está a ser efetuado sobre dados obtidos de um dispositivo estacionário. Veja-se a tabela 7 que apresenta os dados relevantes a esta análise.

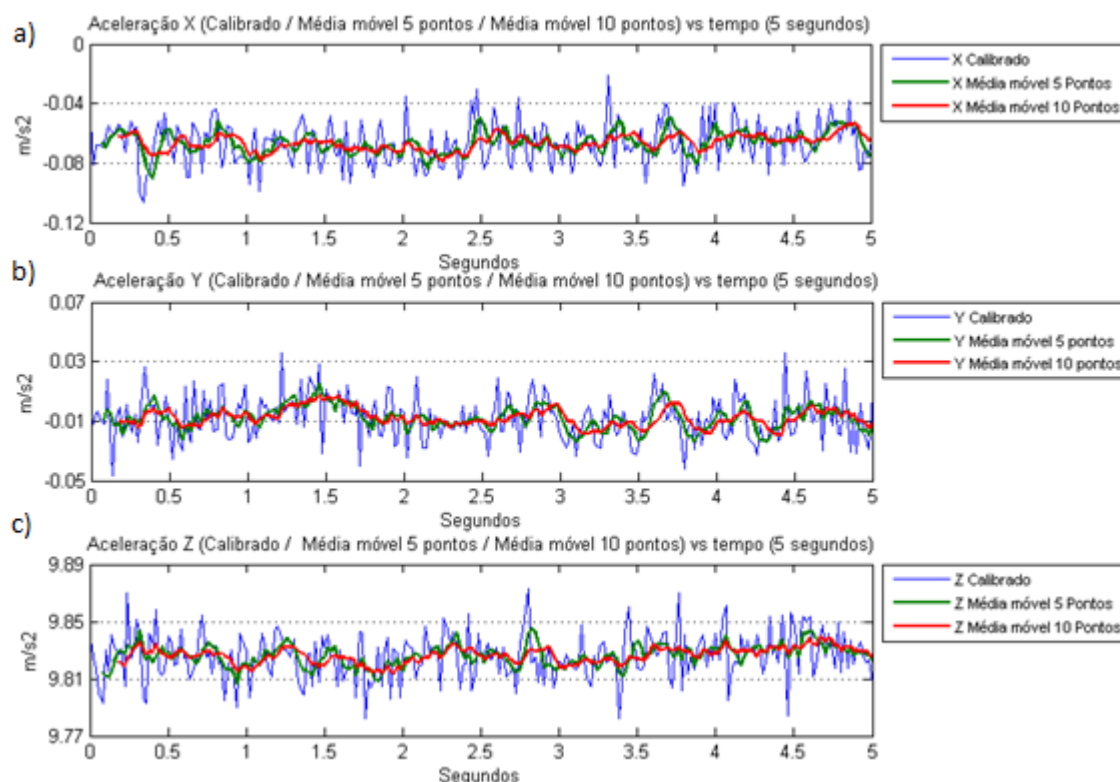


Figura 36 - Comportamento do sinal após aplicação da média móvel com 5 (verde) e 10 (vermelho) pontos nos eixos assinalados como a) X, b) Y e c) Z para o cenário 1.

Os resultados apresentados na tabela 7 levam a concluir que o bom comportamento na suavização dos registos obtidos do sensor é maior, quanto maior o número de pontos usados no filtro. O número de pontos está diretamente relacionado com o número de picos ou variações que o registo sofre, conforme se pode constatar olhando para o valor dos desvios dos registos dos eixos e desvios da aceleração total comparativamente à tabela anterior (tabela 6).

Tabela 7 - Estatísticas dos valores obtidos do acelerómetro pós filtragem com o filtro média móvel nas diferentes posições (1,2 e 3).

Cenário	Estatística	Ax	Ay	Az	Magnitude
1	Média	-0.064513	-0.006080	9.827100	9.827312
	Desvio (5 pontos)	0.008041	0.008894	0.009397	0.009085
	Desvio (10 pontos)	0.006471	0.007234	0.007037	0.007022
2	Média	9.942965	-0.014380	0.077510	9.943287
	Desvio (5 pontos)	0.010546	0.007936	0.010910	0.010574

	Desvio (10 pontos)	0.008426	0.006297	0.009402	0.008459
	Média	2.943915	3.472775	8.672646	9.795003
3	Desvio (5 pontos)	0.010861	0.010810	0.011765	0.011328
	Desvio (10 pontos)	0.008178	0.008699	0.009963	0.009382

Obviamente não influencia o resultado da aceleração total média, mas permite obter resultados interessantes sobre a aceleração total ao longo do tempo aquando da obtenção da velocidade e posição e permitirá também obter um limite superior e inferior mais adequado aquando da determinação do estado do veículo (parado ou não).

5.2.2.2 Filtro média móvel ponderada exponencial (*Exponentially Weighted Moving Average*)

O filtro média móvel simples olha para cada ponto na janela móvel como sendo igualmente importante no cálculo do valor médio. No entanto, em sistemas dinâmicos, os valores mais atuais são aqueles que refletem melhor o estado do processo. Um filtro que coloca mais ênfase nos dados mais recentes poderia de certa forma ser mais útil.

Este filtro pode ser desenhado partindo do princípio do filtro de média móvel simples e adicionando a média de mais um ponto sendo expresso na equação (39) (Shome, et al., 2012).

$$y_n = y_{n-1}(1 - \alpha) + x_n \quad (39)$$

onde y é o output, x o input e α é uma constante representativa da frequência de corte e vai ditar o grau de filtragem ($0 \leq \alpha \leq 1$). Neste caso, um valor mais baixo de α vai resultar em registos mais suavizados. Este filtro coloca maior importância aos dados mais recentes através do desconto de dados mais antigos de uma forma exponencial. Este filtro é um filtro do tipo IIR (resposta ao impulso de duração infinita) pois é recursivo porque o *output* é usado recursivamente como *input* fazendo com que o filtro pareça infinito.

Este filtro é normalmente conhecido como *low-pass filter* ou filtro passa baixo que permite que frequências mais baixas que a frequência de corte passem pelo filtro.

Olhando para as características do resultado ilustrado na figura 37, podemos chegar a algumas conclusões.

A utilização do filtro permite obter um bom resultado no que toca à harmonização dos registos. Responde de forma quase imediata nas variações mais preponderantes e apresenta ainda assim um bom histórico dos picos do sinal original.

Conforme referido anteriormente, quanto mais baixo o valor de α , maior a supressão dos picos do registo original.

Podemos comparar este registo com o filtro média móvel, resultado que pode ser visto na figura 38 para o eixo X no cenário 1. Para se poderem comparar os registos dos dois filtros, teremos de limitar ainda mais a janela de tempo efetuando um “zoom” para visualizar as diferenças para o valor de alfa usado, que são muito semelhantes e apenas se podem diferenciar quando vemos os pontos mais de perto.

5 Aquisição e análise de dados dos sensores de um dispositivo móvel

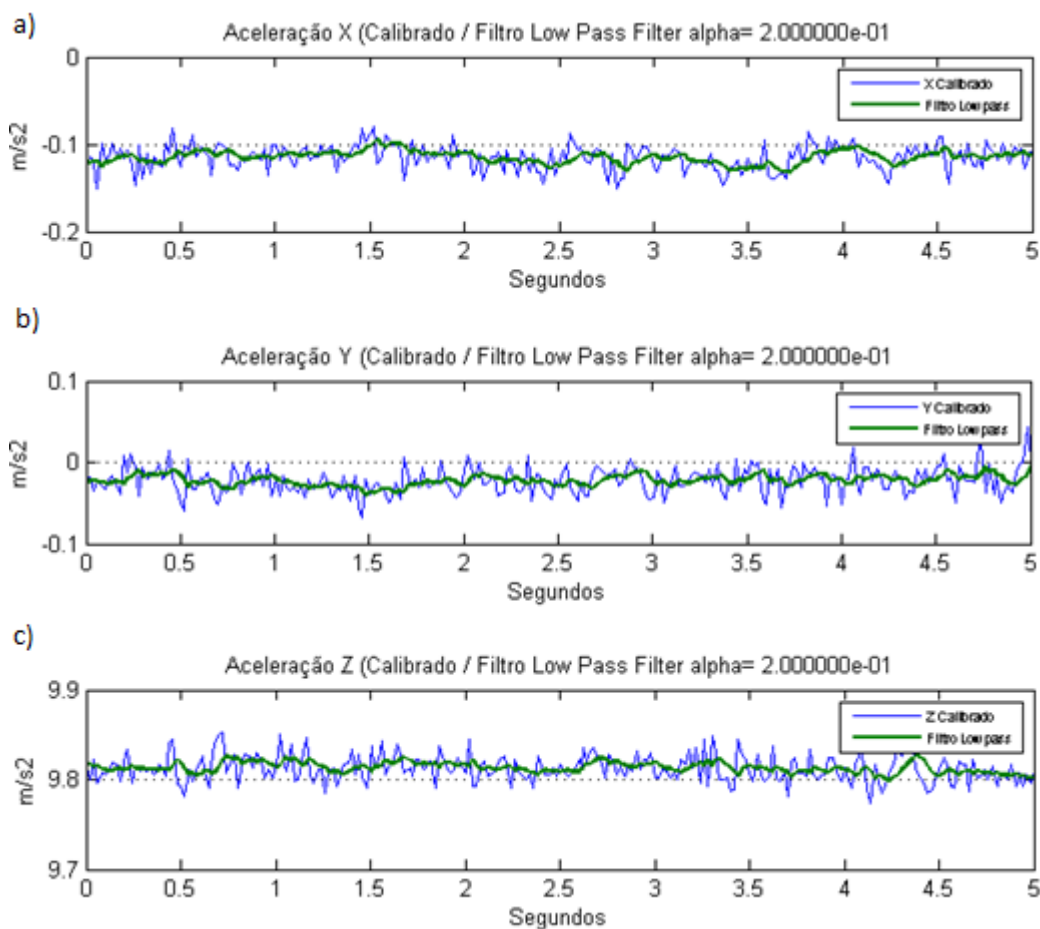


Figura 37 - Registo da aceleração nos eixos assinalados como a) X, b) Y e c) Z e visualização dos mesmos registos com o filtro passa baixo com $\alpha = 0.2$.

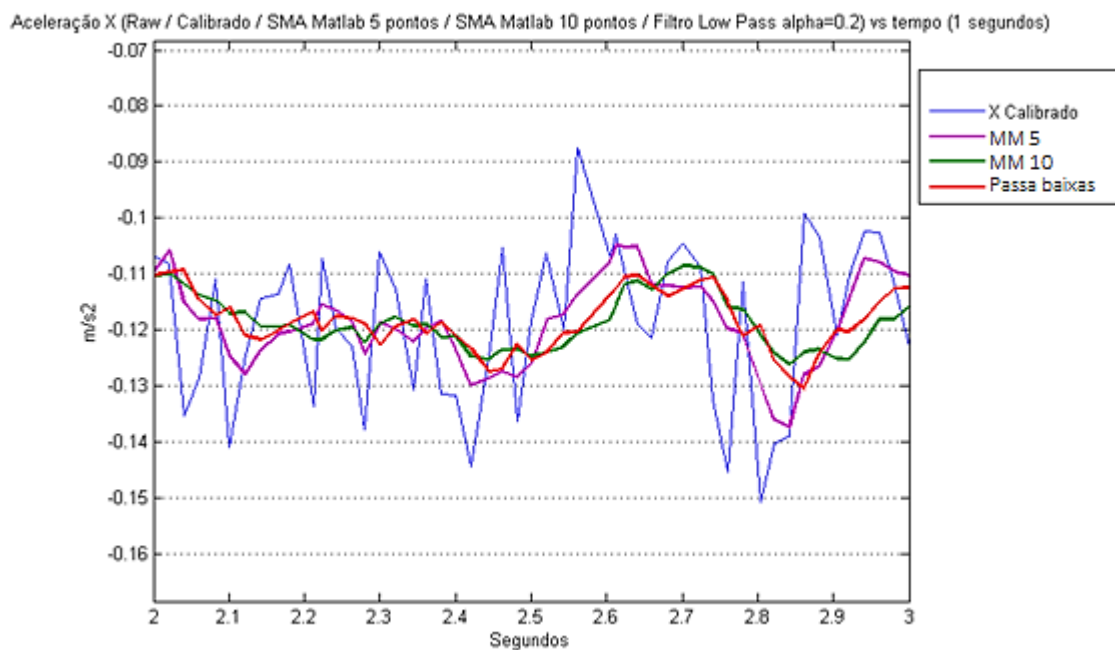


Figura 38 - Comparativo dos registos de aceleração no eixo X para o cenário 1 com o filtros média móvel (MM) a 5 e 10 pontos e o filtro passa baixo com $\alpha=0.2$.

Tendo em conta o valor de alfa, o registo filtrado pelo filtro passa baixo ainda assim consegue acompanhar os picos do registo inicial.

Obviamente, não querendo correr o risco de o filtro retirar os componentes do sinal da aceleração, deveremos tentar encontrar o melhor valor para α de forma a obter o melhor resultado. As duas simulações referentes ao cenário 1 para $\alpha = 0.35$ e $\alpha = 0.55$ estão demonstradas na figura 39 e figura 40, respetivamente.

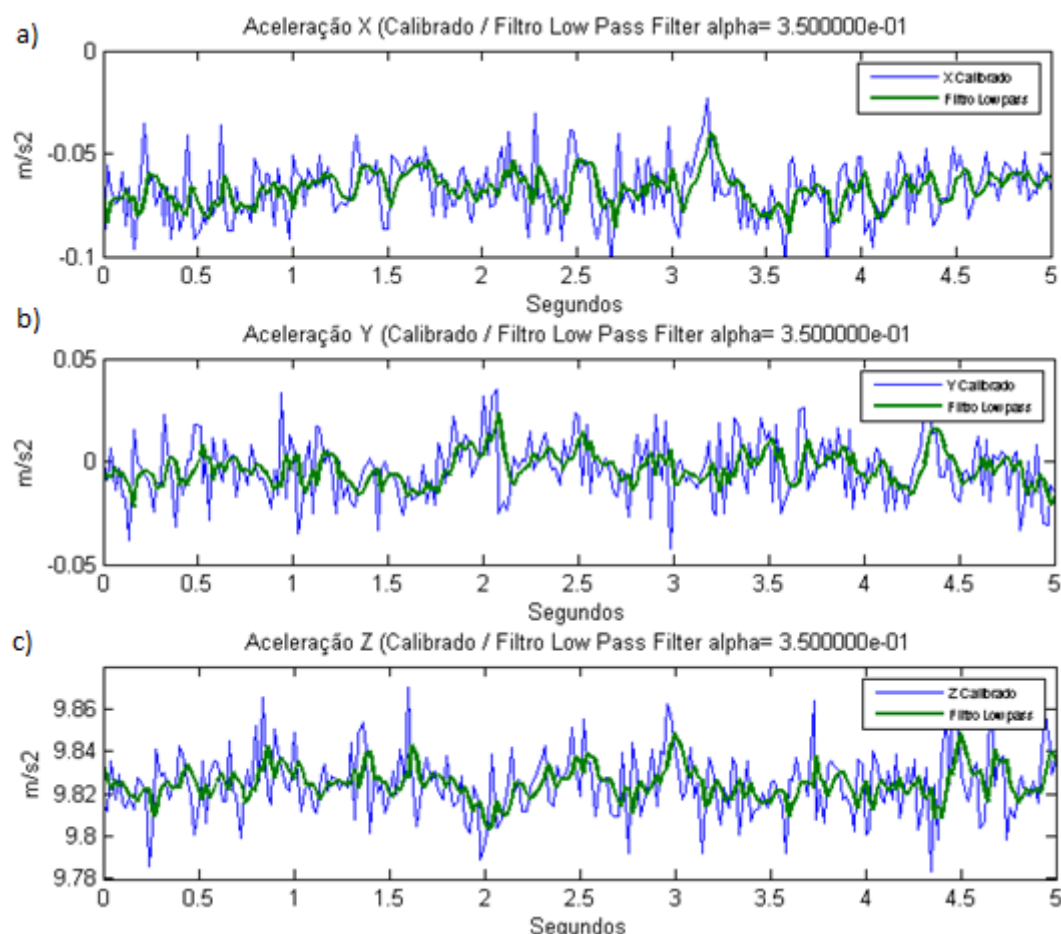


Figura 39 - Registo da aceleração nos eixos assinalados como a) X, b) Y e c) Z e visualização dos mesmos registos com o filtro passa baixo com $\alpha = 0.35$.

Quanto maior o valor de α no filtro, maior acompanhamento faz dos picos do sinal original descrevendo as componentes principais do mesmo mas a suavização do registo e eliminação do ruído é menor. Aumentando o α para 0.55, o acompanhamento é ainda maior conforme verificamos na figura 40.

5 Aquisição e análise de dados dos sensores de um dispositivo móvel

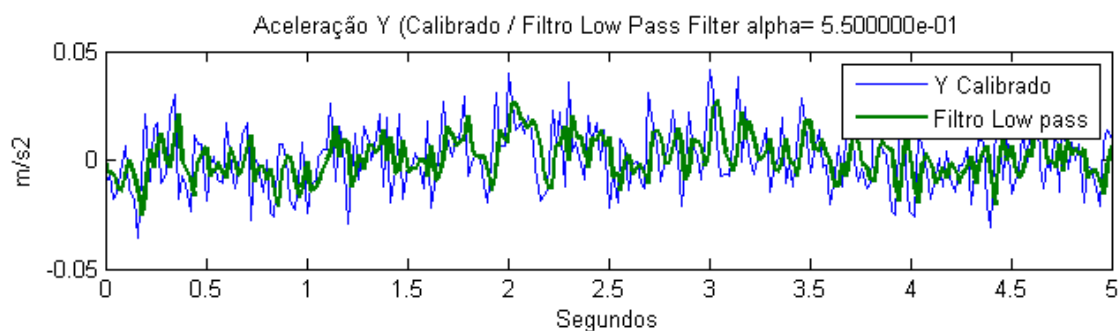


Figura 40 - Registro da aceleração no eixo Y e visualização dos mesmos registos com o filtro passa baixo com $\alpha = 0.55$.

Com o fim de testar o filtro num ambiente mais próximo do objetivo deste trabalho, foi elaborado um novo cenário cujo objetivo foi sujeitar o sensor a movimento. O dispositivo foi colocado num veículo acoplado ao para-brisas numa posição fixa. Visto que queremos testar a veracidade da informação, teremos que ter um ponto de referência no que toca, por exemplo, à velocidade e para isso passou-se a registar também a velocidade obtida do GPS. O objetivo será diminuir o nível de ruído existente no sinal representativo do movimento a que o dispositivo está sujeito.

Tendo em conta que existirá movimento, não faz sentido efetuar uma análise aos desvios padrões pois o valor pode ir muito além do intervalo $[-1g \ 1g]$. Para isso concentramo-nos na análise ao gráfico por forma a verificar o comportamento em movimento. Outro aspeto relevante será verificar até que ponto a utilização do filtro melhora ou não a determinação da velocidade através da integração dos valores da aceleração.

O novo cenário a acrescentar aos três definidos anteriormente na subsecção 5.2.1 representa um trajeto em linha reta com o dispositivo paralelo ao chão e o eixo Y posicionado longitudinalmente relativamente ao veículo. O alinhamento é feito manualmente e o equipamento que apoia o *smartphone* usado para gerar as amostras, por vezes apresenta alguma inclinação alterando os valores do sensor.

A figura 41 representa o resultado final deste cenário com o filtro média móvel simples a 5 e 10 pontos e o filtro passa baixo com $\alpha=0.30$.

A curva da velocidade calculada pela aceleração do eixo Y neste cenário permite obter um resultado muito semelhante relativo à velocidade reportada pelo GPS. Constata-se que nos primeiros segundos, o GPS não apresentou aceleração pois o veículo estava parado. Já o acelerómetro registou alguma aceleração, mesmo parado e provavelmente devido ao facto de o eixo em questão não estar devidamente alinhado.

Relativamente aos filtros usados, verificamos que os mesmos tiveram pouco efeito no cálculo da velocidade, apresentando registos de onda muito semelhantes e pouco expressivos no cálculo final. Estes filtros vão ser necessários no cálculo do alinhamento e orientação pois terá de se acompanhar o ângulo do dispositivo em movimento e para isso os registos filtrados do acelerómetro e giroscópio assim como do magnetómetro, assumem um papel essencial.

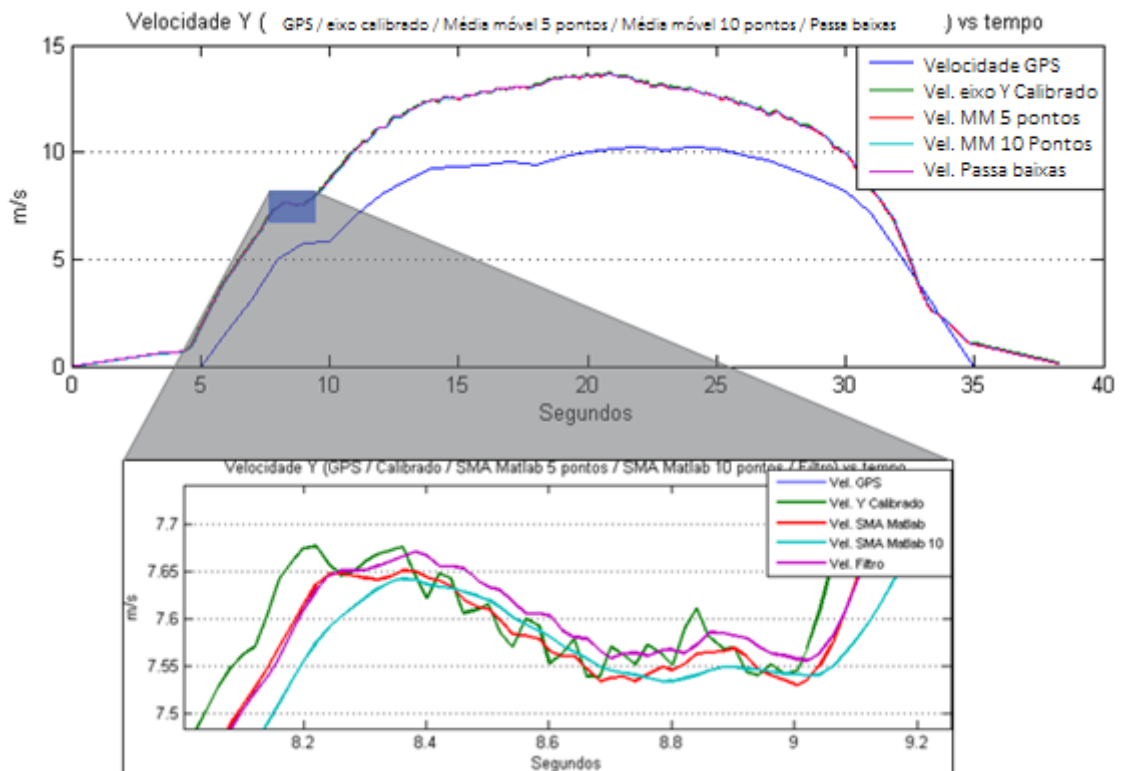


Figura 41 - Cálculo da velocidade com a aceleração do eixo Y previamente calibrado e utilização de filtros Média Móvel e passa baixo.

5.3 Análise do giroscópio

O giroscópio, conforme definido no início deste trabalho, é um dispositivo que permite medir a velocidade de rotação em torno de um centro de rotação ou eixo, ou seja, mede o quão rápido um objeto gira em relação a um sistema de referência inercial, conhecida como velocidade angular. Esta velocidade angular pode ser integrada em relação ao tempo para se obter a posição angular.

Consoante a montagem do sensor, a rotação é medida em relação a um dos três eixos, *Yaw*, *pitch* e *roll*. O princípio que está por trás dos giroscópios MEMS é o efeito de Coriolis que estabelece que uma massa m , movendo-se com uma velocidade linear v num referencial que roda com uma velocidade angular w sofre o efeito de uma força denominada força de Coriolis F_c e como consequência uma aceleração Coriolis a_c ambas representadas pelas equações (40) e (41).

$$F_c = 2mv \times w \quad (40)$$

$$a_c = 2v \times w \quad (41)$$

A medida da velocidade angular é feita utilizando a aceleração de Coriolis a_c .

5 Aquisição e análise de dados dos sensores de um dispositivo móvel

A grandeza dos sinais gerados por estes dispositivos vem tipicamente expressa em rad/s (radianos por segundo). Quando o dispositivo está imobilizado sobre uma superfície os valores gerados pelo giroscópio deverão ter uma magnitude de 0 rad/s.

Geralmente os valores retornados pelo giroscópio estão sujeitos a derivas (*drifts*) causados pelos erros explicados anteriormente. Quando se efetuou uma análise aos valores do giroscópio reparou-se que esses erros estavam de alguma forma suprimidos, ou seja, o sensor apresenta valores pós-processados conforme se pode ver no gráfico da amostra retirada dos sinais dos eixos do giroscópio. No entanto o giroscópio não apresenta valores nos eixos como zeros absolutos quando imóvel, originando deriva ao longo do tempo quando integrado para se obter o ângulo.

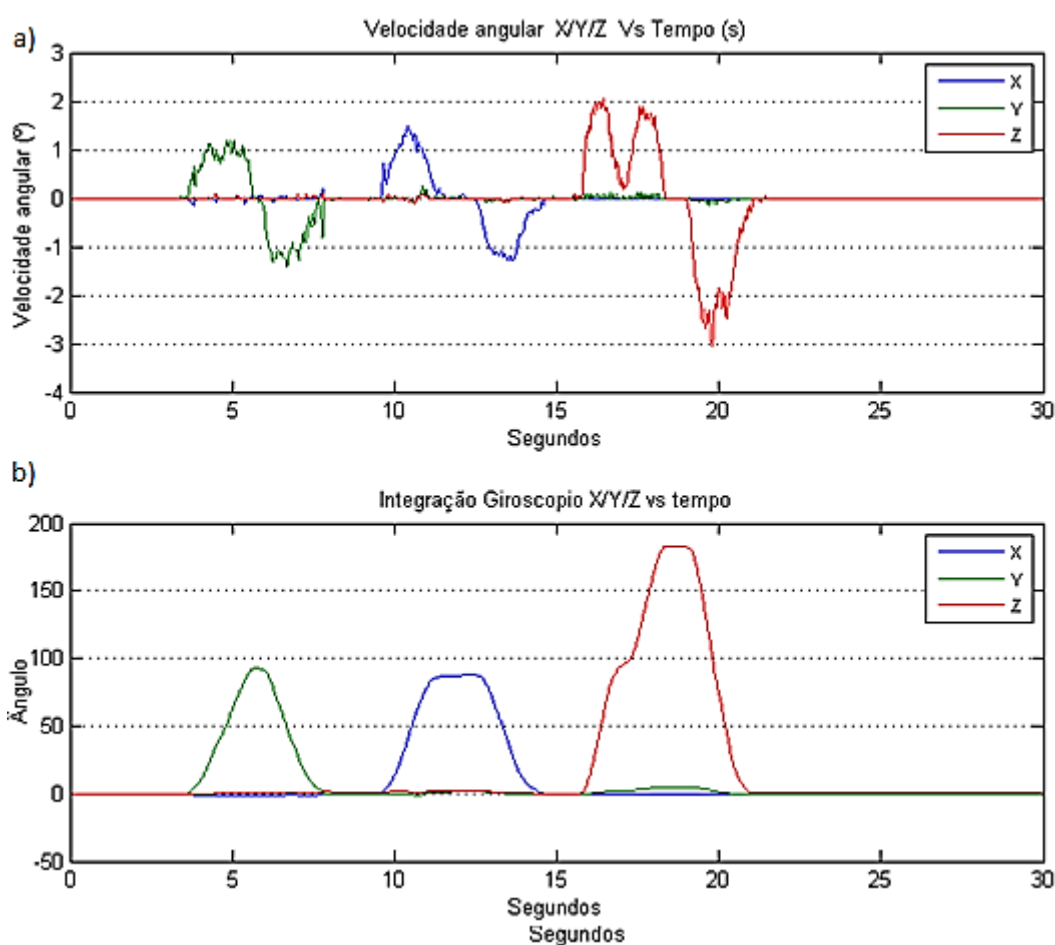


Figura 42 - a) Velocidade angular ($^{\circ}/s$). b) ângulo ($^{\circ}$) após integração.

Note-se na figura 42 b) como os eixos apresentam pouca deriva relativamente ao movimento do giroscópio. Isto é conseguido porque o sensor do dispositivo apresenta sinais provenientes de um processamento, ou seja, o sinal obtido já é pré-processado. Este aspeto relevante pode ser constatado do segundo 16 ao segundo 20 do gráfico na figura 42 onde se nota a deriva (*Drift*) característica do giroscópio no eixo Y a ser corrigida por um filtro. Este processamento geralmente tem como característica a fusão de sensores. Combinando características de outros sensores, torna-se possível com um filtro complementar ou filtro de Kalman, obter valores tratados para compensar o resultado final. No caso aqui retratado, o giroscópio é um sensor da marca *Invensense* MPU-3050 e tem como características um processador digital de

movimento (DMP) e um barramento que serve de interface para acelerómetros de forma a complementar um sistema 6DOF através de processos de fusão.

Para além desta característica, a API do Android de nível 14 (*Ice Cream Sandwich*) do dispositivo utilizado, permite a fusão recorrendo a um filtro de Kalman onde os valores do giroscópio são refinados com os do acelerómetro. O sensor virtual vetor rotação (*TYPE_ROTATION_VECTOR*) também recorre a este filtro derivando os seus valores de uma fusão Acelerómetro/Magnetómetro/Giroscópio.

Apesar dos valores obtidos pelo giroscópio aparentarem bons resultados, de acordo com análises realizadas, o vetor de rotação providenciado pela API, não apresenta os melhores resultados, pois a determinação do vetor rotação necessita da informação do ângulo de inclinação relativo ao referencial global. Este ângulo de inclinação relativo ao referencial global, corresponde aos valores obtidos do acelerómetro que por sua vez está contaminado pelos erros de *Bias* e de escala. Visto que estes valores de *Bias* e escala são determinados de acordo com o sistema implementado e descrito anteriormente, é necessário recorrer a outras técnicas na obtenção dos registos de orientação do dispositivo que não os providenciados pelo Android. Esta técnica está descrita na subsecção 5.5 como filtro de orientação capaz de obter a orientação do dispositivo relativamente ao referencial global usando um filtro complementar com os sensores acelerómetro, magnetómetro e giroscópio.

5.4 Análise do magnetómetro

O magnetómetro é utilizado para determinar a direção que o dispositivo toma relativa aos polos magnéticos terrestres. No entanto a sua precisão é extremamente afetada por diversas interferências magnéticas provocadas pela aproximação de outros campos magnéticos ou materiais ferrosos. Para além desta desvantagem, existe outra que prende-se com o facto do magnetómetro, após um movimento brusco, demorar algum tempo até estabilizar os seus registos novamente.

Este sensor mede a força do campo magnético no meio ambiente em μT (micro teslas)³⁹. O valor varia entre $30 \mu T$ e $60 \mu T$ perto dos polos norte e sul. Veja-se o comportamento dos sinais do magnetómetro na figura 43 segundo o cenário 1 concebido e definido atrás:

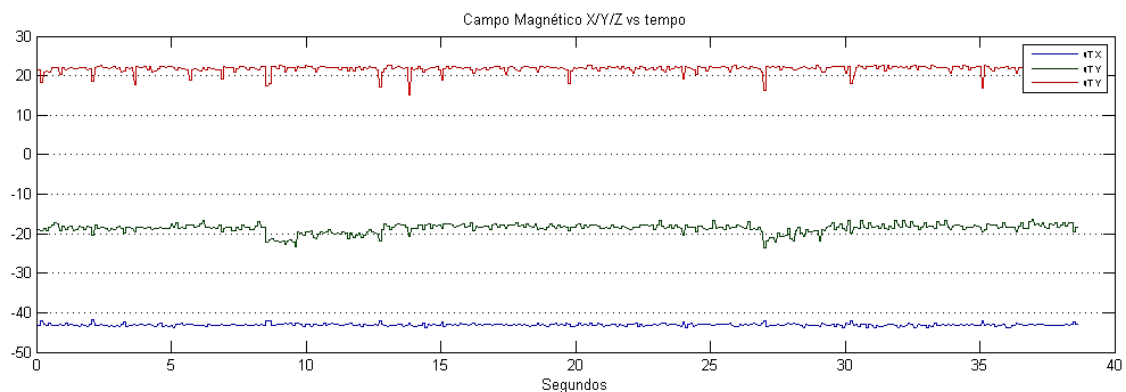


Figura 43 - Output do magnetómetro nos eixos x, y e z.

³⁹ Tesla (T) é a unidade de indução magnética ou campo magnético no sistema internacional.

Esta amostra foi retirada a uma frequência aproximada de 49 Hz durante aproximadamente 38 segundos. A tabela 8 ajuda a entender o valor real destes dados.

Tabela 8 - Estatísticas dos valores obtidos do sensor magnetómetro.

	Mx	My	Mz	Magnitude
Média	-43.080099	-18.616301	21.775060	51.736011
Desvio	0.302366	1.124439	0.885755	0.542481

O primeiro aspeto que se destaca é o desvio padrão do magnetómetro em todos os eixos que é muito elevado. Estes registos foram obtidos tendo em conta a minimização das interferências magnéticas. O desvio padrão da magnitude é de $0.542481 \mu T$, representando entre 1 a 5% do valor dos eixos e pode potencialmente produzir um grande erro.

O magnetómetro vai complementar os registos do acelerómetro e giroscópio para se obter a componente *yaw*, *heading* ou direção dos registos de orientação.

5.5 Filtro complementar de orientação

A componente *yaw* é essencial num sistema de navegação inercial e representa a direção do dispositivo relativamente ao polo norte magnético. No entanto é necessário usar uma combinação de sensores se queremos que esta informação seja minimamente rigorosa e em intervalos de tempo curtos. Conforme especificado anteriormente, o componente de medição que acompanha a orientação do dispositivo usa vários sensores disponibilizados pelos dispositivos e plataforma Android. Os sensores especificados têm as suas forças e fraquezas, *e.g.*, o giroscópio presente em muitos dos dispositivos móveis, apresenta registos que integrados com o tempo, resultam num *drift* muito elevado, típico em soluções de *hardware* de baixo custo, nomeadamente de dispositivos MEMS.

Os ângulos de orientação podem ser extraídos combinando para isso o magnetómetro e o acelerómetro. O acelerómetro providencia um vetor tridimensional que contém a aceleração para cada eixo. Com o vetor a apontar constantemente para o centro da Terra, os graus de inclinação, nomeadamente o *pitch* e *roll*, podem ser determinados. Para a determinação do *yaw*, usa-se a componente do eixo magnético fornecida pelo magnetómetro.

A API do Android providencia funções para determinar estes valores sem ser necessário implementar cálculos adicionais. No entanto, usando esses mesmos métodos, erros são introduzidos no cálculo da orientação onde os dois sensores, acelerómetro e magnetómetro, são responsáveis pela inclusão de graus de imprecisão pela falta de calibração e existência de ruído nos registos.

A esta função de suprimir as fraquezas de cada sensor com as suas forças chama-se de fusão sensorial. A fusão sensorial permite agregar os melhores aspetos de cada um dos sensores para colmatar as falhas de cada um deles. Apesar do acelerómetro e do magnetómetro providenciarem dados de orientação que não derivam ao longo do tempo, estes contêm demasiado ruído. Já os giroscópios, presentes em alguns dispositivos móveis, com API Android inferior a 14 (este aspeto é importante pois as API posteriores usam um filtro de Kalman para

tratar o *drift*), produzem velocidades de rotação com muito pouco ruído mas com pequenos erros provenientes da integração que acumulados ao longo do tempo podem adulterar os dados de orientação. O filtro complementar é uma solução para resolver esta questão e pode ser resumidamente esquematizado de acordo com a figura 44.

Os sensores utilizados nesta solução são o acelerómetro, o magnetómetro e o giroscópio. O filtro passa baixo do sinal do acelerómetro em conjunto com o magnetómetro (orientação acelerómetro + magnetómetro na figura 44) providencia ângulos de orientação.

O filtro passa alto dos registos do giroscópio integrados ao longo do tempo é implementado através da substituição dos componentes de baixa frequência da orientação Acelerómetro/Magnetómetro com a correspondente orientação do giroscópio conforme a equação (42).

$$FusaoOrientacao = (1 - \alpha)valorGiro + (\alpha)valorAccMag \quad (42)$$

onde *FusaoOrientacao* é o valor do filtro complementar que retornará o Azimute (*Yaw*), *Pitch* e *Roll* e α o coeficiente do filtro, *valorGiro* é a medida proveniente do sensor giroscópio e *valorAccMag* é o valor correspondente à orientação proveniente da combinação do acelerómetro e magnetómetro. Se soubermos o intervalo de tempo entre observações e a constante de tempo então podemos encontrar o coeficiente α do filtro usando a equação (43).

$$\alpha = \frac{\tau}{\tau + dt} \quad (43)$$

onde *dt* é o período da amostragem ou o intervalo de tempo entre observações e τ é a duração relativa do sinal sobre o qual o filtro irá atuar.

A componente filtro passa baixo, deixa sinais que são maiores que a constante de tempo inalterados enquanto a componente filtro passa alto deixa passar sinais que são menores que a constante de tempo.

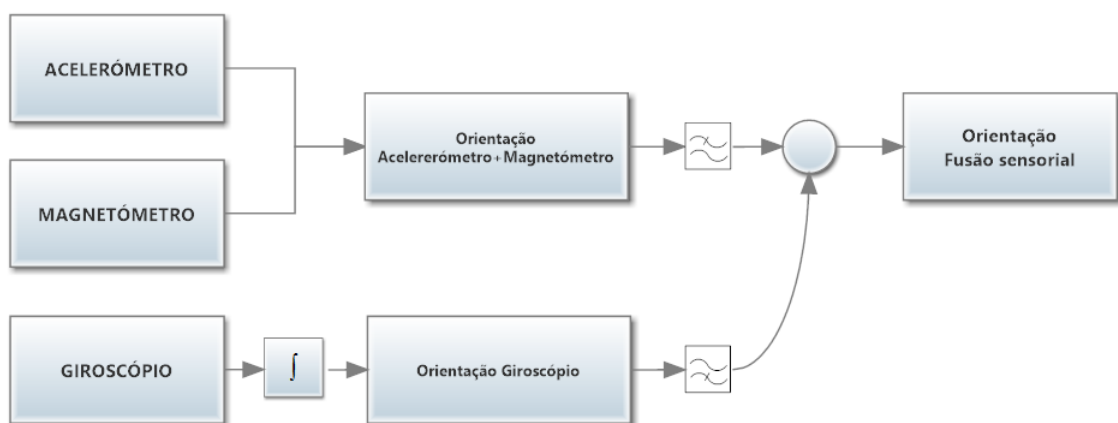


Figura 44 - Fusão sensorial com filtro complementar.

Para se conseguir o resultado final dos sensores usados no filtro, os dados do giroscópio são processados por um filtro passa alto por forma a eliminar o drift característico do giroscópio (pouco relevante no dispositivo usado neste projeto com API superior a 14), substituindo os

dados em falta pela informação dos outros dois sensores, acelerómetro e magnetómetro respetivamente.

Tendo em conta os processos que o Android utiliza para a deteção da orientação, demonstra-se em seguida os resultados da observação da orientação usando a matriz de rotação obtida com recursos aos métodos nativos do Android. O primeiro método é o *getRotationMatrix()* cujos argumentos são o acelerómetro e magnetómetro (legendado como “AccMag” na figura 45, figura 46, figura 47 e figura 48). O segundo método é o *getRotationMatrixFromVector()* cujo argumento é o registo proveniente do sensor virtual *TYPE_ROTATION_VECTOR* (legendado como “RotVet” na figura 48). De seguida, em ambos os métodos, para se obter o azimute, *pitch* e *roll* usa-se o método *getOrientation()*.

Estes métodos nativos do android são comparados com o método desenvolvido *getRotationMatrixFromOrientation(a, p, r)* cujos os argumentos *a, p e r* são respetivamente o azimute, *pitch* e *roll* provenientes da fusão dos registos com recurso ao filtro complementar para suprimir os efeitos do ruído e possibilitar o uso do processo de calibração do acelerómetro (legendado como “AccMagGir” na figura 45, figura 46, figura 47 e figura 48).

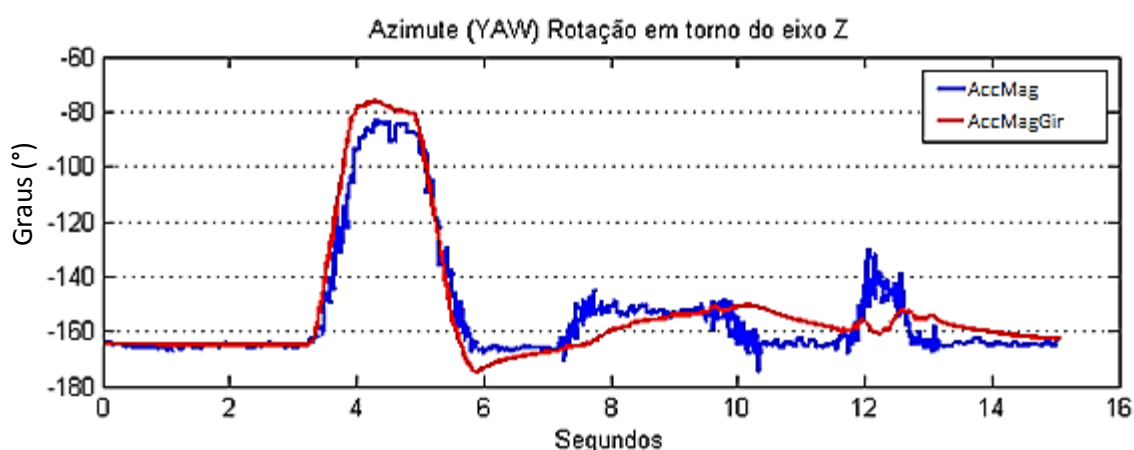


Figura 45 - Comparativo para a obtenção do Yaw com a utilização do método da API Android *getRotationMatrix(acc,mag)* e o método desenvolvido com recurso ao filtro complementar *getRotationMatrixFromOrientation(a,p,r)*.

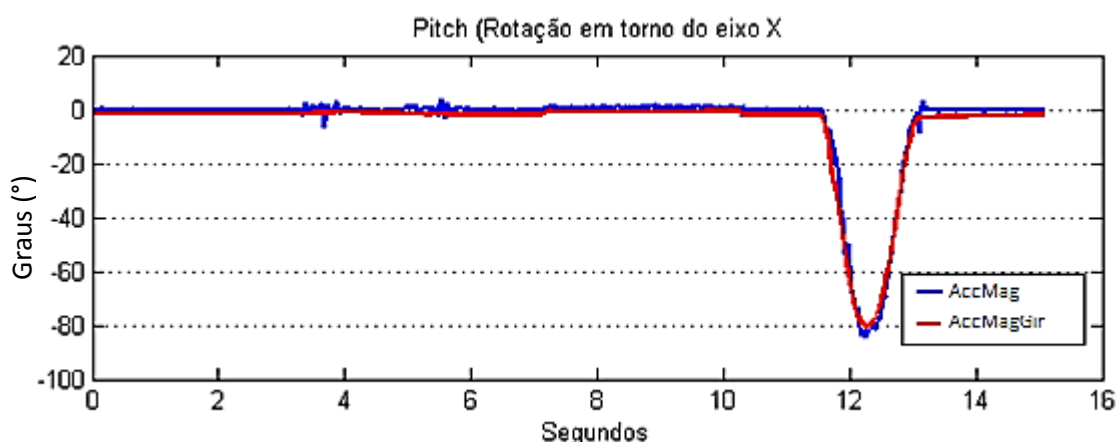


Figura 46 - Comparativo para a obtenção do Pitch com a utilização do método da API Android *getRotationMatrix(acc,mag)* e o método desenvolvido com recurso ao filtro complementar *getRotationMatrixFromOrientation(a,p,r)*.

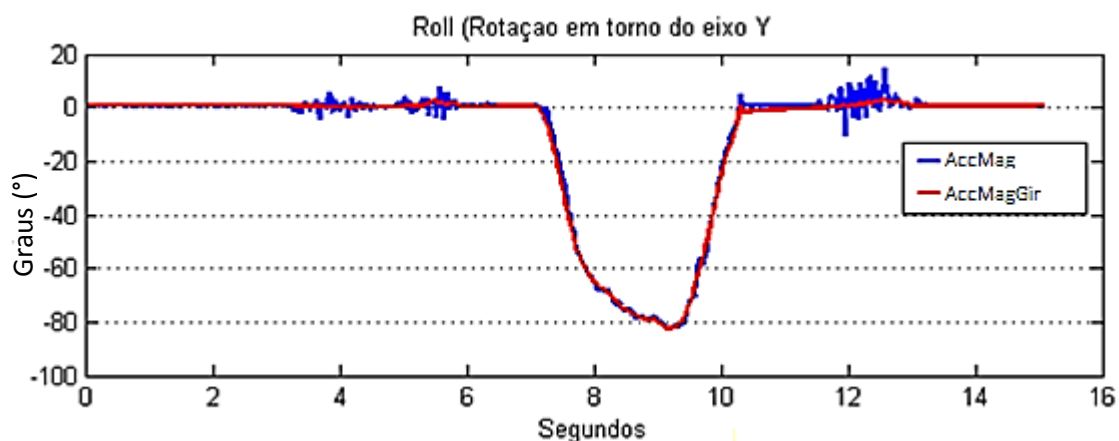


Figura 47 - Comparativo para a obtenção do Roll com a utilização do método da API Android *getRotationMatrix(acc,mag)* e o método desenvolvido com recurso ao filtro complementar *getRotationMatrixFromOrientation(a,p,r)*.

Os resultados finais demonstrados para o *Yaw*, *Pitch* e *Roll* traduzem uma maior suavização dos registos com o filtro de fusão e um tempo de resposta ainda assim semelhante comparativamente aos calculados pela orientação obtida da combinação do acelerómetro e magnetómetro usando o método nativo da API Android *getRotationMatrix()* cujos argumentos são precisamente o vetor do campo magnético e o vetor aceleração da gravidade.

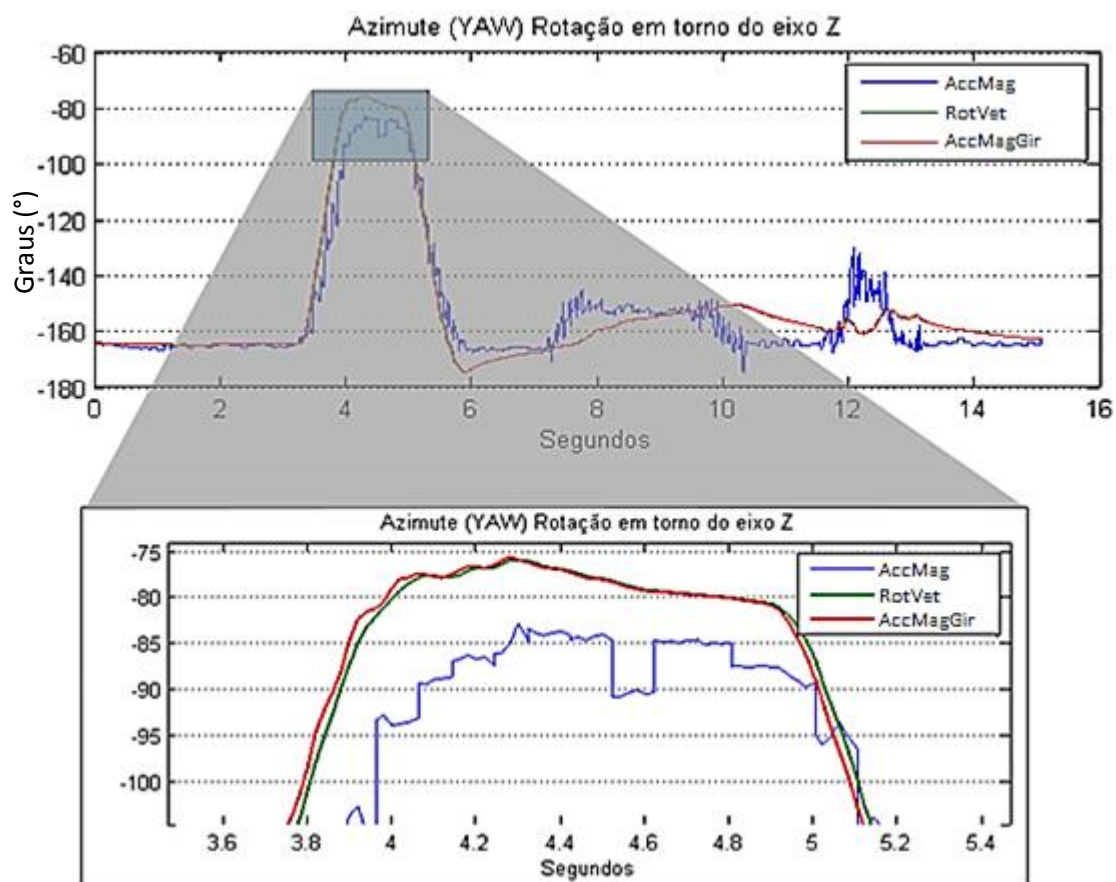


Figura 48 - Comparativo na utilização do três métodos na determinação do YAW.

Tendo em conta que o Android calcula nativamente a gravidade efetuando um filtro passa alto do acelerómetro, este vem viciado com os registos do sensor acelerómetro sem prévia calibração.

A plataforma Android providencia outro método de cálculo da orientação para dispositivos com giroscópio integrado. O método obtém a orientação a partir da matriz rotação calculada através do método *getRotationMatrixFromVector()* cujo argumento são os registos do sensor virtual *TYPE_ROTATION_VECTOR* chamado de *Rotation Vector*. A mesma comparação foi efetuada usando os três métodos, *getRotationMatrix()* combinando os registos do acelerómetro e do magnetómetro, *getRotationMatrixFromOrientation()* cujos registos são provenientes do filtro complementar e *getRotationMatrixFromVector()* com a utilização do sensor virtual *TYPE_ROTATION_VECTOR* na determinação do *Yaw*.

Verificou-se que na determinação da orientação com os métodos *getRotationMatrixFromVector()* nativo e *getRotationMatrixFromOrientation()* os resultados são muito semelhantes onde até por vezes os pontos sobrepõem-se. Este gráfico apenas demonstra que a orientação obtida da matriz de rotação por cada um dos métodos é semelhante no seu resultado final não determinando se os valores que a compõem (matriz de 3x3) são mais precisos ou não. Os valores de orientação *Yaw*, *Pitch* e *Roll* são obtidos com recurso às equações (44), (45) e (46).

$$Yaw = \text{atan2}(Cbn[1], Cbn[4]) \quad (44)$$

$$Pitch = \text{asin}(-Cbn[1]) \quad (45)$$

$$Roll = \text{atan2}(-Cbn[6], Cbn[8]) \quad (46)$$

onde *atan2* representa uma função para o cálculo do arco tangente com dois argumentos cujo segundo argumento determina o quadrante, *asin* o arco seno e *Cbn* é a matriz rotação de mudança de referencial local para global de 3x3.

Tendo em conta que o sensor virtual *TYPE_ROTATION_VECTOR* disponibiliza registos obtidos por um filtro de Kalman usando os registos do acelerómetro, giroscópio e magnetómetro, a razão pela qual o uso do filtro complementar apresentará melhores resultados nesta situação é a calibração e atenuação do ruído preliminar dos sensores, nomeadamente o acelerómetro. O sensor virtual *TYPE_ROTATION_VECTOR* usa os registos dos sensores sem a calibração preliminar necessária no acelerómetro, logo inquinando os resultados finais. Esta observação pode ser facilmente comprovada mediante a observação da aceleração no referencial global e a determinação da aceleração linear, lembrando que para se obter a aceleração no referencial global, teremos de multiplicar a matriz rotação obtida, pelo vetor aceleração no referencial do dispositivo e para determinar a aceleração linear, teremos de retirar o peso da aceleração da gravidade em cada um dos eixos independentemente da orientação. A figura 49 e figura 50 mostram o resultado para os eixos X, Y e Z do acelerómetro no referencial global em dois cenários cuja posição inicial e final do dispositivo é igual. O resultado esperado será obter os eixos X, Y e Z com valores aproximados ao vetor {0,0,g} quando o dispositivo estiver imobilizado, com o ecrã voltado para cima, independentemente da sua orientação.

Os cenários foram equivalentes para as duas observações: uma com o filtro complementar ativo e outra com o sensor virtual *TYPE_ROTATION_VECTOR* ativo. A intenção aqui é demonstrar que utilizando o filtro complementar, o dispositivo marcará o valor mais aproximado do $\{0,0,g\}$. Os movimentos descreveram uma elevação do dispositivo, uma ligeira rotação em torno do eixo X e Z e tornar a realizar o circuito inverso pousando novamente o dispositivo na mesa registrando as acelerações. Analisando a figura 49 nomeadamente os registos nos eixos X e Y, apesar do dispositivo efetuar uma movimentação e orientação aparentemente aleatória, estes mantêm-se fidedignos ao referencial para o qual estão a ser referidos. O processo de calibração aqui é essencial destacando-se os registos dos eixos X e Y que quase se sobrepõem. Na figura 50, representativa dos registos nos eixos X, Y e Z do acelerómetro no referencial global com o auxílio do sensor virtual *TYPE_ROTATION_VECTOR*, isso já não acontece e nota-se perfeitamente o quanto a inexistência de uma calibração inicial pode afetar os registos.

Na figura 49, os eixos X e Y estão notoriamente desfasados registando uma ligeira aceleração constante no eixo Y, quando esta é na realidade inexistente. Este desfasamento vai obviamente contaminar o cálculo da velocidade e posição quando integrados introduzindo erros que poderão diminuir a janela de tempo para a determinação da velocidade minimamente precisa em intervalos de tempo onde os registos GNSS não existem.

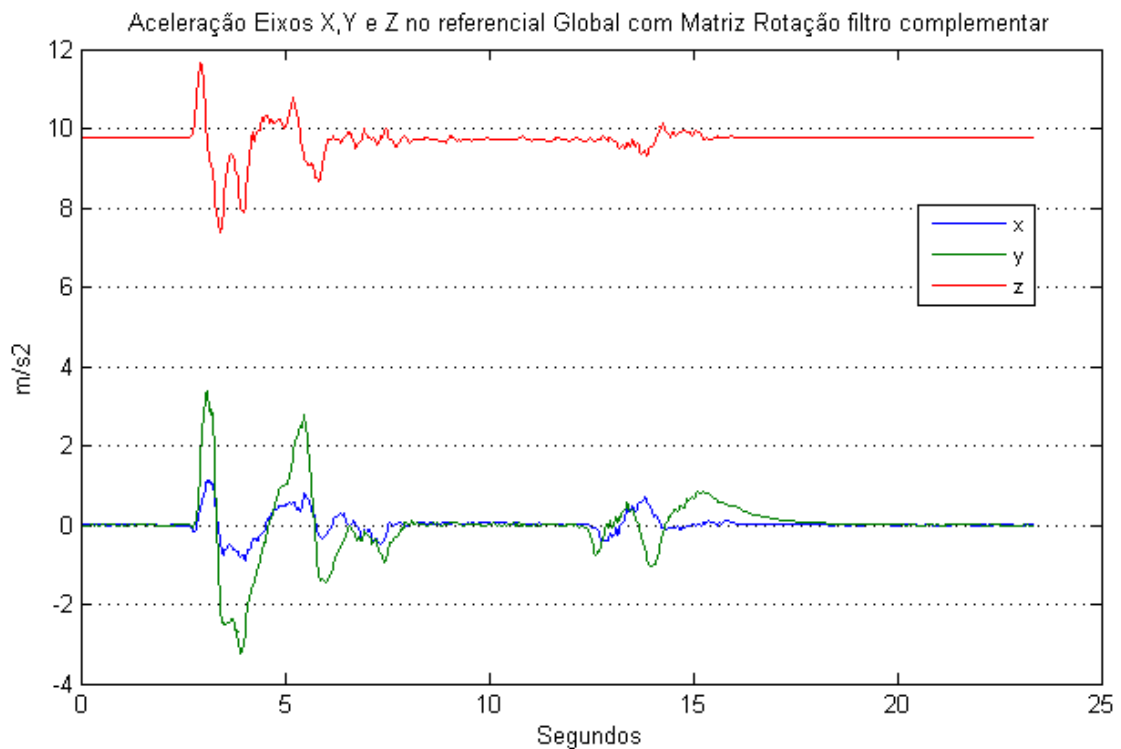


Figura 49 - Obtenção da aceleração dos eixos X, Y e Z no referencial global com o auxílio da matriz de rotação obtida com o filtro complementar.

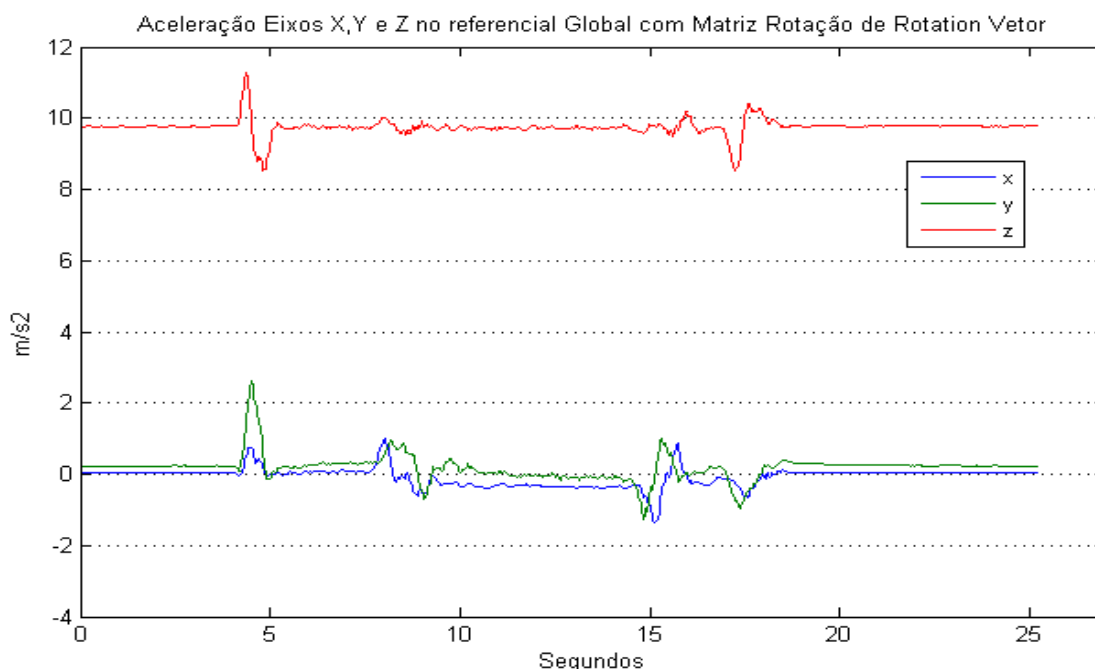
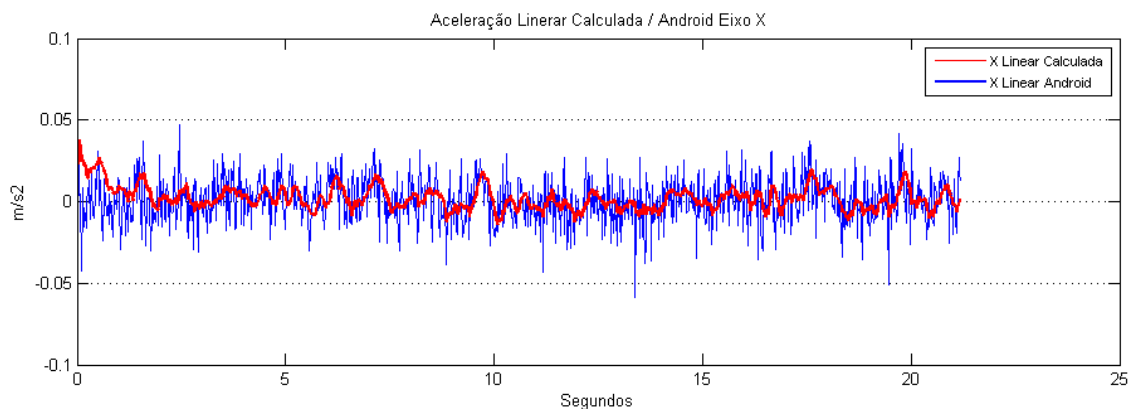


Figura 50 - Obtenção da aceleração dos eixos X, Y e Z no referencial global com o auxílio da matriz de rotação obtida com o auxílio do sensor virtual *TYPE_ROTATION_VECTOR*.

Na observação seguinte, num novo cenário, podemos verificar o resultado do sensor virtual *TYPE_LINEAR_ACCELERATION* comparativamente à obtenção da aceleração linear com o auxílio do filtro complementar. Apesar de neste novo cenário o dispositivo estar deitado numa mesa com o ecrã voltado para cima com uma ligeira inclinação sobre o eixo Y, os resultados deverão ser idênticos independentemente da orientação deste. Conforme dito anteriormente, para se calcular a aceleração linear usa-se a seguinte equação (47).

$$a_m = a_b - C_n^b \cdot \begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix} \quad (47)$$

onde a_m é a aceleração linear, a_b é a aceleração sentida pelo dispositivo, C_n^b a matriz de rotação que permite transformar as coordenadas do referencial global para local e o vetor $\begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix}$ é representativo da aceleração da gravidade no referencial global.



5.6 Cálculo da gravidade com o dispositivo em movimento

Figura 51 - Comparativo dos registos do acelerómetro linear do Android e do calculado com o auxílio do filtro complementar para o eixo X.

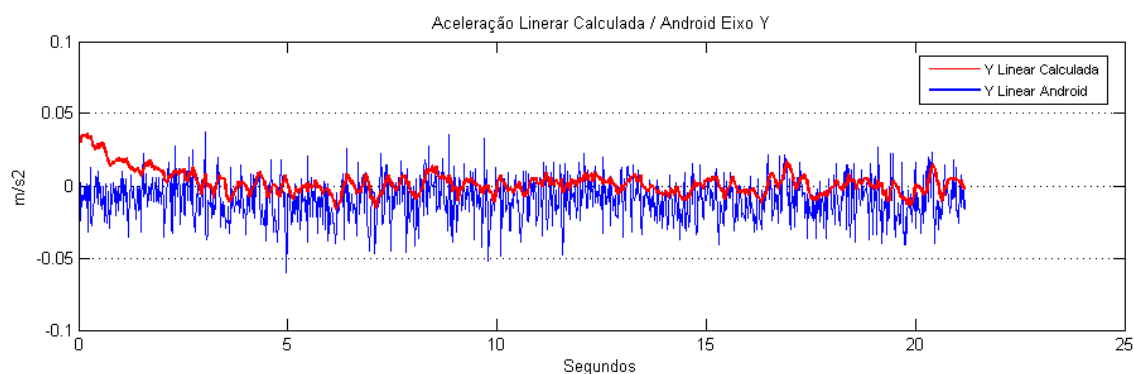


Figura 52 - Comparativo dos registos do acelerómetro linear do Android e do calculado com o auxílio do filtro complementar para o eixo Y.

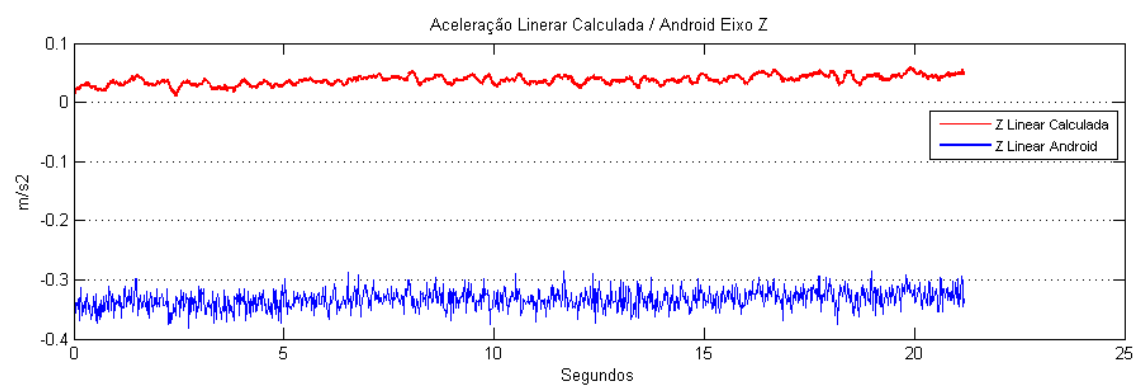


Figura 53 - Comparativo dos registos do acelerómetro linear do Android e dos calculados com o auxílio do filtro complementar para o eixo Z.

Analisando os resultados, repare-se no efeito que o filtro tem na inicialização do mesmo até ficar regular nas figuras 49 e 50 nos primeiros 5 segundos. Esta inicialização será tida em conta no desenvolvimento da aplicação para que os registos fiquem regulares antes da determinação de qualquer observação. Por outro lado, merece destaque o facto da velocidade linear calculada com o filtro, ter um comportamento bastante mais rigoroso que o registo da aceleração linear do sensor virtual do Android, visto os registos aproximarem-se mais do resultado real (0,0,0) num cenário onde o dispositivo se encontra sem qualquer aceleração. Este resultado deve-se sobretudo à calibração e tratamento do ruído do acelerómetro *a priori*, algo que o Android não faz nos métodos nativos que utiliza.

5.6 Cálculo da gravidade com o dispositivo em movimento

A determinação da aceleração do movimento pelo dispositivo móvel utilizado no desenvolvimento deste projeto é extraída sobretudo pelo acelerómetro. O vetor da aceleração contém diversos componentes tais como a aceleração do movimento e a aceleração da gravidade a que o dispositivo é sujeito e por isso mesmo é importante

acompanhar a componente da aceleração da gravidade durante todo o processo de determinação da aceleração linear.

A separação da gravidade de todas as outras forças, com o dispositivo imóvel, torna-se uma tarefa fácil pois com um simples filtro passa alto somos capazes de separar os componentes do sinal. Se por um lado isto é funcional em movimentos de rotação, movimentos longitudinais de curta duração ou em movimentos onde a orientação é desprezível, por outro lado em movimentos determinados por acelerações mais exigentes e contínuas, esta tarefa é mais difícil. Por exemplo, a própria determinação da matriz de rotação pelo método nativo do Android, *getRotationMatrix()*, exige que o movimento detetado nos eixos pelo acelerómetro do dispositivo seja limitado, correndo o risco de os resultados finais serem pouco precisos.

Se o acelerómetro for usado para medir a gravidade e para determinar o grau de inclinação do dispositivo (conforme implementação no filtro de orientação atrás descrito) e a direção exata do vetor gravidade não for precisa, qualquer aceleração de movimento irá corromper ou adulterar a determinação da aceleração linear não sendo possível subtrair a força da gravidade dos registos obtidos da aceleração.

Sendo o acelerómetro um sensor fulcral na determinação das principais componentes de interesse do projeto, um dispositivo móvel fixo num veículo sujeito a alterações de orientação e aceleração provocadas pelo movimento, se não forem cuidadosamente tratadas podem ser interpretadas como alterações na velocidade num cenário de integração.

Apesar de na secção 3.2.2 ter sido discutida o modo de determinação da gravidade, sujeitando o vetor gravidade no referencial de navegação a uma transformação para o referencial do corpo, a verdade é que a mesma está sujeita a um erro de cálculo derivado sobretudo a parâmetros errados nos filtros. Assim sendo, a aplicação desenvolvida no âmbito deste trabalho, disponibiliza ainda outros dois métodos para a determinação da gravidade de modo a se poder analisar de forma isolada cada um dos sensores e entender a mais-valia do uso do filtro complementar.

Os outros dois métodos que a aplicação desenvolvida disponibiliza para a determinação da gravidade passam pela utilização do giroscópio para efetuar a rotação do vetor gravidade inicialmente estimado e o filtro passa alto. Note-se que estes mesmos sensores são usados para a determinação da orientação com o filtro complementar. Todos os métodos obrigam a que o veículo efetue uma calibração esporádica da gravidade e esta é efetuada de forma automática quando o estado do veículo é detetado como parado.

Para comparação destes métodos, foi definido um cenário onde o dispositivo Android foi fixado no para-brisas de um veículo com um ângulo de aproximadamente 45° com o eixo Y do dispositivo apontado para a frente do veículo. O trajeto é uma linha reta sem inclinações relevantes mas com piso irregular.

O primeiro método referido (utilização do giroscópio para rodar o vetor gravidade) consiste na utilização do giroscópio para separar as componentes gravidade e movimento. Se tivermos inicialmente um vetor gravidade fidedigno, podemos usar os ângulos de rotação produzidos pelo giroscópio para rodar este vetor para uma nova posição. Como o giroscópio não está sujeito à aceleração provocada pelo movimento, a atualização do vetor gravidade pelos ângulos de rotação do giroscópio não será afetada pela aceleração do movimento.

O desenvolvimento deste sistema de obtenção da gravidade com auxílio do giroscópio, passa pela caracterização do algoritmo de simulação de gravidade da seguinte forma:

- Se o valor absoluto da aceleração for significativamente maior ou mais pequeno que 1g, então ignora estes valores provenientes do acelerómetro para a determinação da gravidade e passar a usar o vetor simulado que obtém a gravidade pela rotação angular do giroscópio;
- Se o valor absoluto da aceleração for muito próximo de 1g então atualizar o vetor simulado com os registos do acelerómetro.

O segundo método disponibilizado, o filtro passa alto, será capaz de atenuar o sinal durante tempo suficiente para o registo da aceleração relativa ao movimento tendo pouca capacidade de sentir as alterações da inclinação pois para este filtro a situação ideal é um trajeto plano. Tal como o método anterior, este método exige sempre um alinhamento inicial antes do início da obtenção de qualquer registo.

Para podermos comparar os diferentes métodos de obtenção da gravidade, ou seja, o filtro passa alto e a utilização do giroscópio para determinar a rotação do vetor gravidade com o sensor virtual *TYPE_GRAVITY* do Android analisamos inicialmente o registo obtido pelo acelerómetro calibrado e filtrado com o filtro passa baixo. (figura 54).

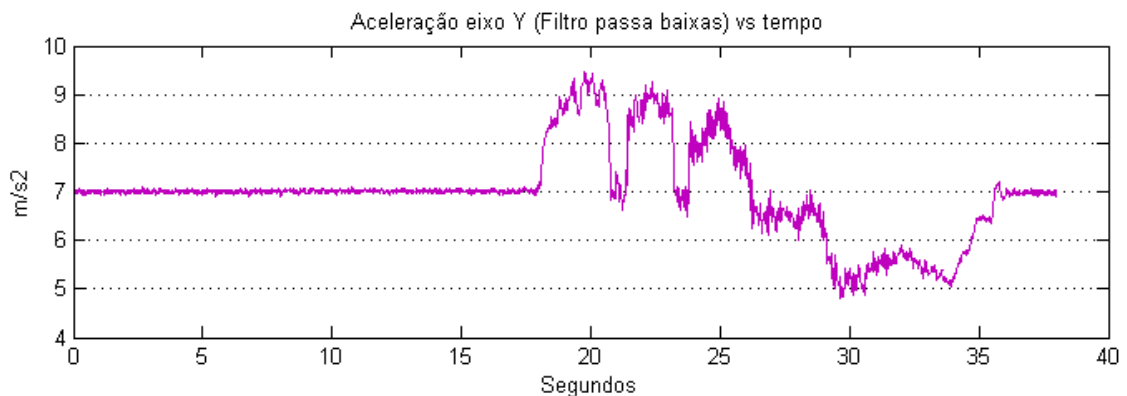


Figura 54 - Aceleração sentida no eixo Y durante o trajeto.

Pode-se constatar, analisando o início e o final do sinal comprovado pela definição do trajeto efetuado e pela estabilização do sinal perto dos 7 m/s², que não existiu alteração significativa na gravidade sentida no eixo Y, mas apenas aceleração responsável pelo movimento.

De seguida registou-se a gravidade proveniente do sensor *TYPE_GRAVITY*, a gravidade determinada pelo método 1 (giroscópio) e a gravidade determinada pelo método 2 (filtro passa alto) cujo resultado pode ser visualizado na figura 55.

5 Aquisição e análise de dados dos sensores de um dispositivo móvel

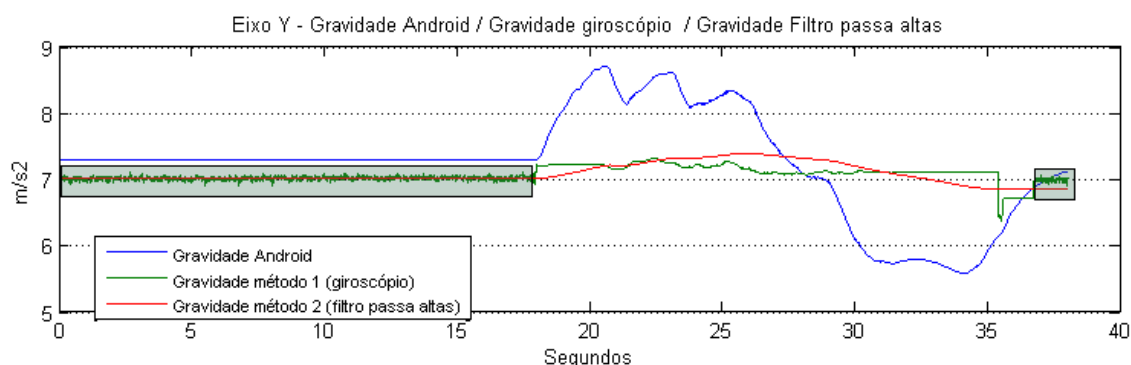


Figura 55 - Registo da gravidade no eixo Y.

Analisando os registos, destaca-se em primeiro lugar o efeito da falta de calibração do sensor acelerómetro no registo da gravidade providenciada pelo sensor `TYPE_GRAVITY`. Podemos também verificar o quanto o movimento afeta a determinação da gravidade apresentada por este sensor virtual cujo resultado final acaba por ser irrealista.

Relativamente ao registo da gravidade determinada pelo primeiro método, os dados indicam a obtenção inicial do vetor gravidade do acelerómetro e a determinação da rotação do mesmo ao longo do trajeto com o uso do giroscópio. O algoritmo infere qual o sensor a ser usado para determinar a gravidade, ou seja, o acelerómetro ou o giroscópio consoante este deteta o dispositivo imóvel ou não. Para destacar esses momentos, a figura está sinalizada com retângulos verdes nos dois momentos em que o acelerómetro se torna responsável pela determinação da gravidade. Durante os restantes momentos, é o giroscópio que efetua a rotação do vetor gravidade. Este algoritmo e técnica são funcionais em movimentos como os registados em sensores para consolas de jogos no registo de movimentos determinando a dinâmica de um jogo. No entanto em acelerações geradas por movimentos persistentes, onde estas são registadas em longos intervalos de tempo (trajetos), com períodos de aceleração longos e seguidos (*e.g.*, alternância nas acelerações do veículo) e alterações de direção repentinas e/ou longas aquando da aceleração em curva, este algoritmo pode apresentar problemas. Um dos problemas é o facto do giroscópio MEMS ainda não ser um sensor fiável e ser perturbado pelo *drift* gerado pela integração, *drift* esse que poderia deturpar o registo da gravidade, não sendo por isso confiável. Para implementação do método 1 (giroscópio), a complexidade do processo de cálculo da gravidade com o auxílio do sensor giroscópio, obriga a definir parâmetros relativos ao tipo de trajeto (sinuoso ou não) e relativos ao dispositivo (sinal do giroscópio tratado ou não consoante a API do Android).

O registo da gravidade determinada recorrendo ao método 2 (filtro passa alto) pode ser visualizado na figura 55 através da linha vermelha. Nota-se que este método não tem em conta as alterações do piso, logo não respondendo de forma imediata a alterações da inclinação do trajeto.

De seguida registou-se a aceleração linear para analisar as diferenças do sinal obtido pelo uso dos dois métodos relativamente ao sensor virtual `TYPE_LINEAR_ACCELERATION`. Para a determinação da aceleração linear com os métodos 1 e 2, subtraiu-se ao vetor aceleração, a gravidade determinada cujos resultados podem ser visualizados na figura 56.

5.6 Cálculo da gravidade com o dispositivo em movimento

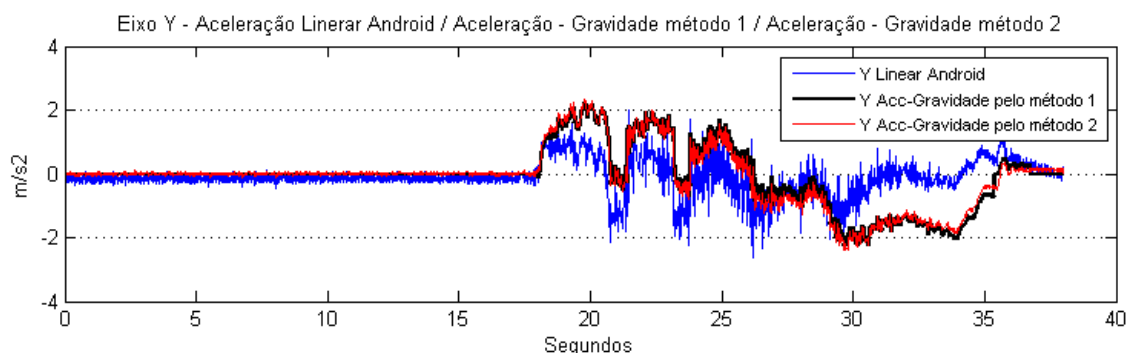


Figura 56 - Aceleração linear (sem efeito da gravidade) determinada pelo sensor virtual *TYPE_LINEAR_ACCELERATION*, aceleração menos a gravidade determinada pelo método 1 (giroscópio) e aceleração menos a gravidade determinada pelo método 2 (filtro passa alto).

Comparando os registos da figura 54 com a figura 56, verificamos uma perturbação causada pela aceleração do movimento na aceleração linear obtida pelo sensor *TYPE_LINEAR_ACCELERATION*. O facto dos registos da aceleração Linear obtida do sensor *TYPE_LINEAR_ACCELERATION* do Android apresentarem tanto ruído tem a ver com o facto de estes não estarem a ser filtrados.

Já a determinação da aceleração linear cujo resultado é determinado pelo registo da aceleração menos o efeito da gravidade determinada pelos métodos 1 (giroscópio) e 2 (filtro passa alto) correspondentes às linhas vermelha e preta da figura 56, retornam um registo mais fidedigno.

Comparando o resultado final da determinação da aceleração total e correspondente velocidade com a velocidade obtida do GPS, podemos verificar uma aproximação que vai ao encontro das necessidades do projeto conforme se pode verificar nos resultados na figura 57.

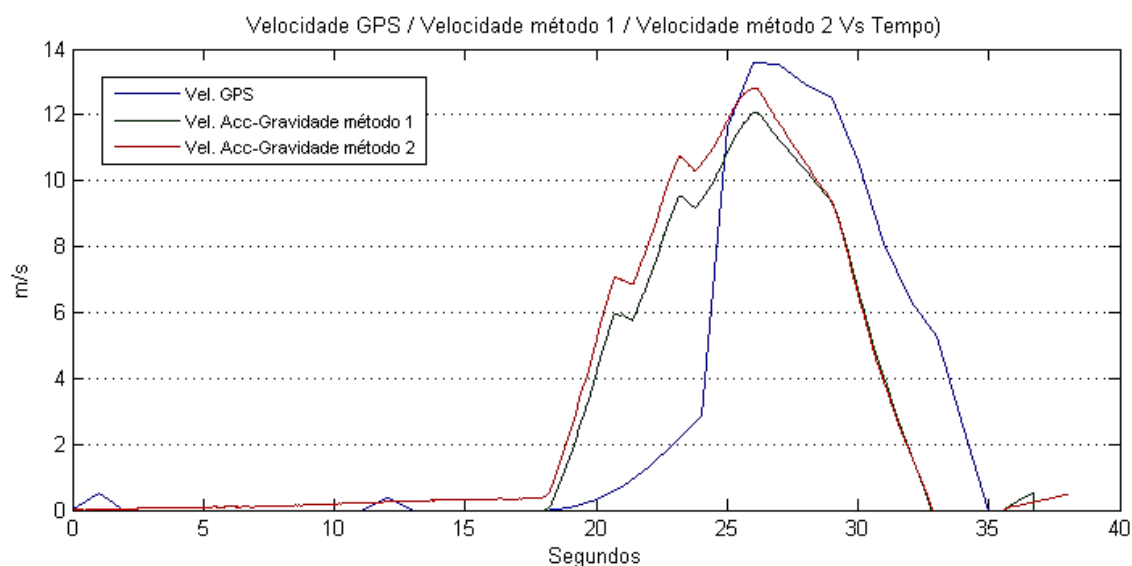


Figura 57 - Registo da velocidade obtida do GPS, velocidade determinada pela aceleração total menos a gravidade calculada pelo método 1 e velocidade determinada pela aceleração total menos a gravidade calculada pelo método 2.

5 Aquisição e análise de dados dos sensores de um dispositivo móvel

Os dois métodos apresentam resultados interessantes num curto espaço de tempo e podem ser selecionados mediante o tipo de trajeto que se venha a percorrer.

Outro aspeto interessante que pode ser visualizado na figura 57, é a rapidez na obtenção da velocidade oferecida pelo INS relativamente ao GPS. Com o sistema GPS conseguimos, em boas condições, obter estimativas da velocidade a uma média de 1 por segundo. Com o INS, é possível obter a estimativa da velocidade 45 vezes por segundo, conseguindo assim diminuir o *delay* existente no GPS.

A utilização do giroscópio para determinar a gravidade pode ser uma opção vantajosa pois existe um maior controlo do vetor gravidade em circuitos sinuosos. O giroscópio permite efetuar a rotação do vetor gravidade consoante o veículo estiver a subir ou a descer por exemplo, obrigando a que o peso da gravidade nos diferentes eixos altere. O filtro passa baixo não consegue acompanhar este processo porque o valor de alfa definido para este filtro funcionar é muito baixo e o tempo de resposta a alterações de gravidade nos eixos em circuitos sinuosos é demasiado longo.

Por estes motivos, o terceiro método (determinação da gravidade com o auxílio da matriz de rotação do filtro complementar discutido na secção 3.2.2) para a obtenção da gravidade, ou seja, a utilização da matriz de rotação obtida pelo filtro complementar, pode ser a melhor opção. Se por um lado o ângulo de inclinação obtido pelos dados do acelerómetro filtrado por um filtro passa baixo tem um tempo de resposta longa, já o ângulo de inclinação obtido pela integração dos dados do giroscópio apresenta uma resposta rápida apesar do *drift* a que está sujeito.

6 Desenvolvimento da solução

Com a análise efetuada sobre os métodos de obtenção e tratamento dos dados de acordo com as necessidades do projeto, importa agora definir em linhas gerais os aspetos relativos às várias funcionalidades da aplicação móvel desenvolvida.

Neste capítulo procede-se à descrição dos aspetos mais importantes da aplicação elaborada, ou seja, as restrições ou limitações internas do projeto, a descrição dos módulos desenvolvidos, as técnicas de obtenção e análise de dados e a apresentação dos resultados finais.

A aplicação tem por finalidade pôr em prática os conceitos definidos e testar a implementação dos filtros necessários para ir ao encontro do objetivo que é assegurar a obtenção da localização geográfica do dispositivo móvel em curtos períodos de tempo onde o sinal do GPS é inexistente. Dependendo de um conjunto de fatores, a obtenção da localização unicamente por INS em caso de falha do GPS é mais ou menos precisa.

Os fatores que podem influenciar o resultado final são:

- A sinuosidade do percurso;
- A calibração inicial do sensor acelerómetro do dispositivo móvel;
- A obtenção de uma localização GPS inicial com boa precisão;
- A definição ótima de parâmetros existentes na aplicação, nomeadamente os parâmetros dos filtros de ruído, os limites para o módulo de controlo de paragem do veículo, a frequência dos sinais dos sensores, a seleção do filtro de orientação e a seleção do método de determinação da gravidade junto com a definição da sensibilidade do giroscópio;
- Qualidade dos sensores existentes no dispositivo móvel.

6.1 Restrições impostas ao desenvolvimento

Tendo em conta a obtenção dos resultados finais, convém definir um conjunto de restrições necessárias ao bom funcionamento da aplicação. Estas restrições são definições de limites e imposições sobre as quais a aplicação ficará pronta a produzir resultados satisfatórios.

A primeira restrição imposta é a colocação do dispositivo móvel fixo ao veículo. Assim sendo ficou definido que o dispositivo deveria obedecer a uma posição longitudinal ao veículo com o eixo Y apontado para a frente do veículo e um ângulo de *pitch* superior a 10° e inferior a 60° conforme pode ser visualizado na figura 58, a qual mostra a disposição do dispositivo móvel utilizado para este projeto acoplado no tablier do veículo, ao centro do mesmo. Inicialmente o dispositivo encontrava-se fixo ao lado esquerdo, mas as interferências magnéticas impediam obter registos minimamente fidedignos do magnetómetro.



Figura 58 - Dispositivo móvel utilizado para análise do sistema, fixo no tablier do automóvel.

Para se determinar a velocidade do veículo é calculada a magnitude do vetor aceleração multiplicado pelo período de tempo decorrido até à obtenção do valor. Para se saber se a velocidade total calculada é para ser adicionada ou subtraída à velocidade anterior temos de definir o vetor de desaceleração que em matemática é a aceleração em sentido contrário. A definição dos sentidos da aceleração e desaceleração é um dos motivos desta restrição. Outro motivo é o facto de a determinação do azimute ser compensada em termos de inclinação do dispositivo, pelo acelerómetro e giroscópio, responsáveis pela obtenção da matriz de rotação.

A segunda restrição imposta prevê a necessidade de calibração do sensor acelerómetro para achar os coeficientes de *bias* e escala a que o sensor está sujeito. Apesar de ser já possível utilizar ROMs⁴⁰ personalizadas que permitem a calibração dos sensores, normalmente as ROMs originais não o permitem. A aplicação proporciona um método de calibração do acelerómetro que permite ao utilizador de uma forma simples seguir os passos para a calibração.

A terceira restrição prevê o uso moderado do sistema. Tendo em conta que o acompanhamento da gravidade ao longo de um trajeto não é rigoroso pois está suscetível a

⁴⁰ ROM, literalmente refere-se a “Read Only Memory” mas neste caso refere-se ao *firmware* ou sistema operativo que o dispositivo móvel corre.

erros de integração, caso o veículo não efetue paragens esporádicas para calibração do vetor gravidade, o mesmo deixa de ser consistente e o INS perde o seu efeito prático de disponibilização de localização com o mínimo de rigor em caso de falha de sinal GPS.

Por fim a última restrição ao uso da aplicação consiste no alinhamento inicial e calibração dos limites superiores e inferiores para o controlo de paragem do veículo. Este é um processo que deve ser efetuado com o veículo ligado e demora o tempo necessário à obtenção da primeira localização geográfica pelo GPS e o tempo de obtenção do vetor gravidade inicial que consiste no processo de cálculo da média da aceleração de 500 registos. O controlo de paragem é essencial não só para determinar a paragem do veículo, como também para achar os valores mais recentes do vetor gravidade.

6.2 Activities

As *Activities* são as componentes da aplicação que permitem a interação com o utilizador. Cada *activity* providencia uma hierarquia de *Views* (objetos), tais como caixas de texto, botões, barras de progresso, etc.

A figura 59 mostra as *activities* desenvolvidas para este projeto.

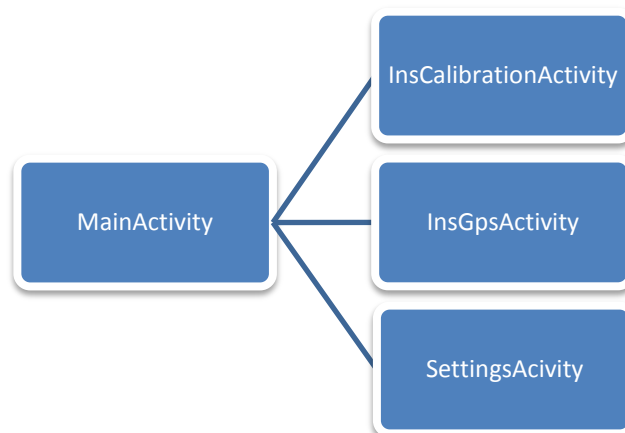


Figura 59 - Principais *activities* da aplicação.

A *MainActivity*, é o ecrã que o utilizador acede quando este inicia a aplicação. Este ecrã apresenta ao utilizador três opções correspondentes às apresentadas na figura 60 e numeradas a vermelho. A primeira opção permite ao utilizador aceder ao ecrã de calibração. A segunda opção permite ao utilizador alterar alguns parâmetros relativos aos processos principais da aplicação. A terceira opção permite ao utilizador aceder às funcionalidades principais, nomeadamente o mapa onde os trajetos são apresentados, e um ecrã informativo onde é possível ler a informação relativa à velocidade e direção.



Figura 60 - Ecrã inicial da aplicação.(1- botão para aceder ao ecrã de calibração; 2- botão para aceder às definições; 3- botão para arrancar o sistema)

Para que a aplicação produza resultados coerentes, na primeira vez que a mesma é iniciada, é necessário proceder à calibração do acelerómetro. O utilizador ao aceder a esse ecrã (figura 61) pode visualizar desde logo, os valores atuais de calibração, caso esta tivesse já sido efetuada.

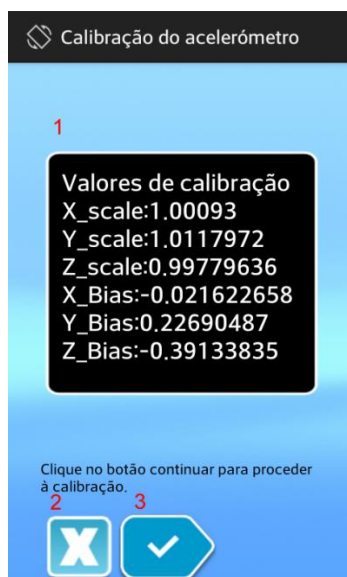


Figura 61 - Ecrã de calibração. (1- Valores atuais de calibração; 2- botão de saída; 3- Botão para continuar)

Ao continuar com o processo (figura 62), surge um novo ecrã onde os valores do sensor acelerómetro são mostrados e refrescados ao ritmo da taxa de frequência definida nos parâmetros da aplicação. Existe outro parâmetro que permite selecionar o sensor que irá apoiar o utilizador no processo de calibração. Este sensor irá apoiar o utilizador mostrando os valores sobre os quais a calibração está a ser efetuada. Os sensores disponíveis para auxílio da

calibração são o *TYPE_ACCELEROMETER* e *TYPE_GRAVITY*. Os dois sensores disponibilizam valores diferentes, visto que um sensor disponibiliza os valores reais da aceleração (*TYPE_ACCELEROMETER*) em todos os eixos e o outro sensor (*TYPE_GRAVITY*) disponibiliza valores obtidos por um processo de fusão, nativo do Android, com o giroscópio.

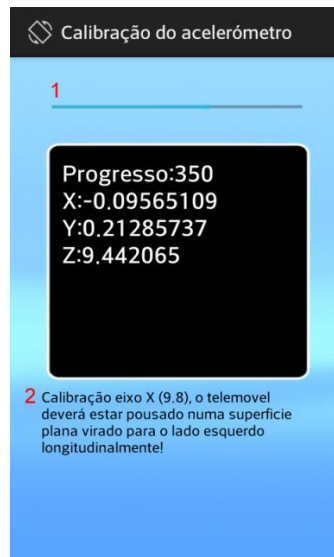


Figura 62 - Processo de calibração. (1- barra de progresso; 2- Texto de apoio)

As etapas de calibração para cada eixo podem ser configuradas, nomeadamente o número de pontos para cálculo da média de acelerações em cada eixo. O dispositivo tem de estar completamente imóvel aquando da calibração para que o acelerómetro meça apenas o valor da gravidade. Os textos de apoio aparecem no ecrã para auxiliar o utilizador sobre a forma como deve colocar o dispositivo para calibração em cada um dos eixos.

Para cálculo do *bias* e fator de escala, a seguinte função é invocada para cada leitura das duas posições necessárias para cada eixo (X,Y e Z):

```
public void calculateCoefficients(float high, float low, String tipo)
{
    float k_temp = 0;
    float b_temp = 0;
    k_temp = (2*SensorManager.GRAVITY_EARTH) / (high-low);
    b_temp = (low * k_temp) + SensorManager.GRAVITY_EARTH;
    savePreferences(k_temp,b_temp,tipo);
}
```

Código 1 - Método de determinação dos coeficientes bias e fator de escala.

As variáveis *k_temp* e *b_temp* tomam os valores determinados pelo método de calibração e equações definidas na subsecção 5.2.1. Estes valores são gravados recorrendo ao método do Android *SharedPreferences()*, que providencia métodos para consultar e guardar dados do tipo “chave-valor”. Estes coeficientes vão ser utilizados na determinação dos valores do acelerómetro, os quais vão gerar o vetor de aceleração calibrado que irá ser usado pela restante aplicação.

A *SettingsActivity* (figura 63), permite definir uma série de parâmetros que irão influenciar os resultados. Estes parâmetros permitem alterar o comportamento da aplicação e assim reconhecer as diferenças entre os diversos métodos de obtenção de resultados durante a análise e desenvolvimento.



Figura 63 - Parâmetros da aplicação.

Os parâmetros que a aplicação permite alterar são os seguintes:

1. Aceleração centrípeta – Este parâmetro permite ao utilizador escolher se pretende determinar a aceleração centrípeta ou não. Este processo é essencial para permitir que a aceleração centrípeta não afete o cálculo da velocidade em trajetos marcadamente sinuosos. O método está detalhado na subsecção 3.2.2;
2. Usa o mapa Google – Permite ao utilizador escolher se pretende que a informação de localização seja apresentada num mapa Google. Os trajetos, quando em modo GPS, são tracejados em azul. Quando em modo INS apenas, então o trajeto é desenhado em vermelho. A aplicação usa a API Google Maps Android v2;
3. Frequência – Permite ao utilizador especificar a frequência dos sensores entre os níveis pré-definidos do Android (*delay UI, delay Normal, delay Game, delay Fastest*);
4. Orientação – O utilizador pode escolher qual o método que pretende utilizar para determinar a orientação. Os métodos possíveis são os especificados na subsecção 5.5, ou seja, o método nativo do Android `getRotationMatrix(TYPE_ACCELEROMETER, TYPE_MAGNETOMETER)`, o `getrotationmatrixfromvector(TYPE_ROTATION_VECTOR)` e o filtro complementar

desenvolvido para o efeito que permite fundir os registos do acelerómetro, giroscópio e magnetómetro;

5. Determinação da gravidade – O utilizador pode escolher o método de determinação da gravidade. Os métodos possíveis são os descritos na subsecção 5.6, ou seja, a utilização de um filtro passa baixo, a utilização do giroscópio para acompanhar o movimento de rotação do vetor gravidade e a utilização da matriz de rotação, método especificado na subsecção 3.2.2;
6. Filtro de ruído – A aplicação permite escolher os filtros utilizados na atenuação do ruído. Os filtros disponíveis são os discutidos na subsecção 5.2.2. Para os filtros, filtro média móvel ou filtro passa baixo, é disponibilizado um campo para definição do número de pontos ou o valor de alfa respetivamente;
7. Controlo de paragem – O utilizador pode indicar se pretende utilizar o controlo de paragem. Este controlo vai permitir ainda definir os limites sobre os quais os valores do acelerómetro serão tidos em conta, ou não, na determinação da velocidade;
8. Calibração de sensores – Permite especificar os valores para a calibração do acelerómetro e magnetómetro. Os valores de *bias* e fator de escala, podem ser especificados manualmente. Para além disto, é possível definir a declinação magnética para o magnetómetro. Esta declinação define o ângulo entre o norte magnético e o norte geográfico. Esta declinação varia consoante a localização no mundo;
9. Log sensores – Quando selecionada, a aplicação irá criar um registo de todos os valores obtidos pelos sensores. Os valores que forem objeto de tratamento (calibração, filtrados, etc) também são registados. O formato dos ficheiros é .csv para serem usado no processo de importação e análise dos dados no MATLAB;
10. Log gps – Permite efetuar o registo das coordenadas, num ficheiro, obtidas do GPS;
11. Log KML – Se esta opção estiver selecionada, a aplicação permite gerar um ficheiro do tipo *Keyhole Markup Language* (KML) cujo formato é compatível com o Google Earth ou o Google Maps.
12. Pacotes UDP – A aplicação desenvolvida permite enviar pacotes UDP com os dados do acelerómetro, giroscópio e magnetómetro para serem visualizados na aplicação desenvolvida no MATLAB. Esta aplicação MATLAB, permite visualizar gráficos sobre os dados em tempo real.

Por fim, a *InsGpsActivity*, cujo interface gráfico apresenta os dados relevantes determinados pela aplicação. A *activity* possui uma *View* do tipo *ViewSwitcher* permitindo organizar a informação em dois ecrãs distintos.

Após o arranque da *activity*, esta vai correr uma série de testes. Em primeiro verifica se o GPS está ativo ou não, e caso não esteja, permite ao utilizador ligá-lo. Após certificar que o GPS está ativo, a aplicação fica em modo de espera até obter sinal GPS (figura 64). Aquando da obtenção da localização do GPS, a *activity* alerta o utilizador que o sistema de INS está a inicializar (figura 65). Posteriormente à inicialização do GPS e INS, o utilizador pode escolher

entre as diferentes *views* que dispõe. Na primeira *view* é possível observar no mapa o trajeto efetuado em tempo real, seja em modo GPS ou INS, distinguindo os dois pela cor do tracejado (figura 66).

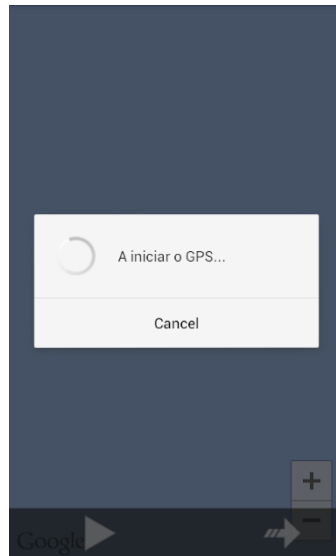


Figura 64 – Caixa de diálogo enquanto a aplicação obtém sinal GPS.



Figura 65 - Caixa de diálogo enquanto o INS inicializa.



Figura 66 - Visualização do percurso efetuado. A vermelho o percurso estimado pelo INS e a azul o percurso obtido pelo GPS. O primeiro botão permite ao utilizador efetuar o log e envio de dados via UDP. O segundo botão permite aceder à segunda *view*.

Com a segunda *view*, é possível observar outras informações em tempo real, tais como a direção determinada pelo GPS, a direção determinada pelo INS, a velocidade GPS e a velocidade INS conforme a figura 67.

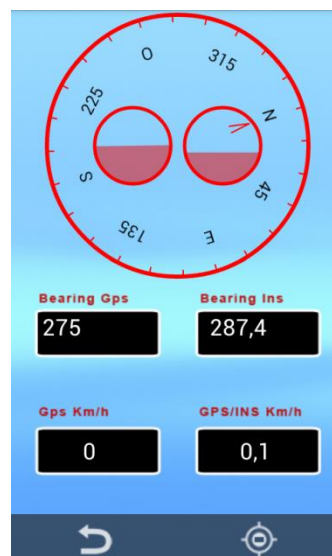


Figura 67 - A aplicação desenvolvida apresenta informações sobre a direção e velocidade do INS e GPS. Com os dois botões em baixo é possível ao utilizador voltar atrás ou simular falha GPS para análise.

A visualização em mapa usa a API do Google Maps Android v2 cujos mapas se encontram encapsulados na classe *MapFragment*, uma extensão da classe *Fragment* do Android.

6.3 Módulo INS

O INS implementado foi simplificado relativamente aos sistemas mais comuns, alguns deles enunciados na secção 2.6. A arquitetura adotada para o desenvolvido da integração INS/GPS foi a *Loosely coupled* pelo facto dos sistemas funcionarem de modo modular e independente.

O INS envolve 4 diferentes fases apresentadas na figura 68.



Figura 68 - 4 fases envolvidas no sistema INS.

A fase de calibração vai providenciar os coeficientes necessários à obtenção de valores calibrados do acelerómetro. Esta é efetuada apenas na primeira vez que a aplicação corre. A partir daqui, os coeficientes para a calibração do acelerómetro serão utilizados na obtenção dos registos pelo acelerómetro.

A fase de alinhamento consiste na obtenção da orientação inicial com vista a interpretar os dados dos sensores. Esta orientação presume a obtenção da matriz de rotação do giroscópio, do magnetómetro e acelerómetro e a obtenção do peso da gravidade inicial em cada eixo. A classe Java desenvolvida para o efeito implementa a interface *SensorEventListener* responsável pela obtenção dos dados dos sensores.

A fase de inicialização consiste na obtenção da velocidade e localização inicial. Estes valores são obtidos do GPS. Foi desenvolvida uma classe que implementa as interfaces *LocationListener*, *GpsStatus.Listener* que implementam os métodos de obtenção das coordenadas e obtenção de informação sobre os satélites, respetivamente.

A fase de avaliação consiste na obtenção da posição relativa ao ponto de referência. O sistema INS vai acompanhar o sistema de GNSS e na falta deste, o INS assume a responsabilidade de determinar a velocidade e orientação relativas. O GNSS enquanto ativo, corrige a velocidade e localização estimada pelo INS.

O processo de controlo de paragem do veículo é essencial nesta fase de avaliação. Ele não só é responsável pela determinação da paragem do veículo, como também propicia o momento para a calibração do vetor gravidade. Sempre que um novo valor de aceleração é obtido, o mecanismo de deteção de paragem é invocado. O que o mecanismo determina é a magnitude do sinal de aceleração e compara-o com o valor obtido aquando da inicialização.

A figura 69 pretende resumir os processos delineados e discutidos anteriormente, assinalando os passos para a obtenção do posicionamento com o INS. Os passos assinalados na figura como 1, 2 e 3, são o processo de obtenção dos dados pelos sensores acelerómetro, giroscópio e magnetómetro. Os dados obtidos pelo acelerómetro são calibrados determinando a equação (33) com os coeficientes estimados durante o processo inicial de calibração do acelerómetro. O ruído do acelerómetro é atenuado pelo filtro escolhido (filtro passa baixo ou média móvel simples). O sinal do magnetómetro também é filtrado por um filtro passa baixo.

Os valores obtidos são usados no filtro complementar (passo 4 na figura 69) para a determinação da orientação, processo discutido na subsecção 5.5.

Para a determinação da velocidade e localização, aos valores do acelerómetro é retirada a aceleração centrípeta (passo 5) e a aceleração da gravidade (passo 6), conforme definido nas equações (26), (27) e (28) referidas na subsecção 3.2.2. A aceleração resultante é integrada uma e duas vezes respetivamente, relativamente ao tempo (passo 7).

O processo passa ainda pela correção da velocidade e inicialização da localização pelo GPS.

O GPS (passo 9) e a determinação da localização (passo 10) é um processo que verifica se o GPS está ativo ou não. Se receber localização do GPS, então essa é usada para obter a Longitude e latitude do dispositivo. Para além disso, a velocidade estimada pelo INS é corrigida pelo GPS. Caso o GPS esteja desativo, então apenas o INS fornece a localização até que o GPS esteja ativo novamente.

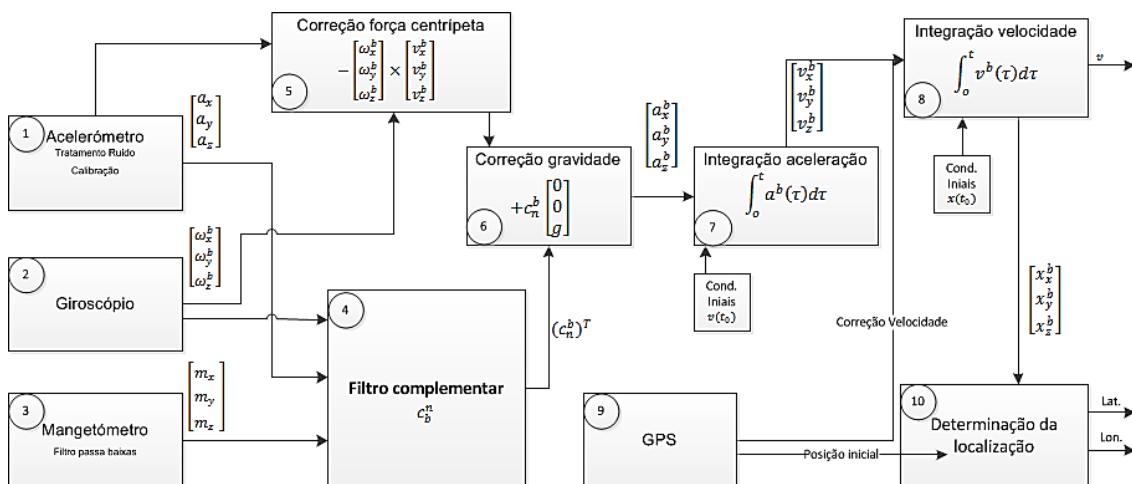


Figura 69 - Resumo dos processos inerentes ao INS implementado.

A determinação da velocidade, caso a mesma esteja a ser obtida apenas pelo INS, é calculada invocando o seguinte método:

```
public void updateVelocityRotation(float[] acc, float dt)
{
    mAcceleration.set(0,acc[0]); // Guarda o valor do eixo x
    mAcceleration.set(1,acc[1]); // Guarda o valor do eixo y
    mAcceleration.set(2,acc[2]); // Guarda o valor do eixo z

    // Força específica (acc=acc+Cbn'*gravity)
    // Dcm em float, tenho de passar para densematrix64f

    for (int i = 0;i<9;i++)
        m_tempdcm.set(i,mCbn[i]);

    // mAcceleration = mAcceleration + (m_tempdcm)T * mGravity
    CommonOps.multAddTransA(m_tempdcm, mGravity, mAcceleration);

    // atualiza vel com força específica (Vel_b=Vel_b+dt*acc)
}
```

```

CommonOps.addEquals(mVelocityBody, dt, mAcceleration);
mVelocityTotal = velocityCorrection +
    operations.getMagnitude(new
        float[] {(float)mVelocityBody.get(1),
            (float)mVelocityBody.get(2)});
updatePosition(dt);
}

```

Código 2 - Método de atualização da velocidade.

O método *updateVelocityRotation* é invocado sempre que é obtido um valor do acelerómetro dentro dos limites de aceleração definidos como sendo aceleração de interesse. O cálculo da gravidade para subtrair ao valor da aceleração, é efetuada segundo o método escolhido nas definições da aplicação. A classe *CommonOps* referida no código permite efectuar a integração da aceleração para a determinação da velocidade. A aplicação também permite determinar a aceleração centrípeta para assim ser removida.

Se nos lembrarmos do cálculo da centrípeta na subsecção 3.2.2, o que o método aplicado na aplicação faz é obter a matriz antissimétrica do vetor velocidade e efetua o produto vetorial entre a matriz antissimétrica e o vetor de valores em radianos obtido do giroscópio multiplicados pelo período de tempo. É efetuada a mesma integração para determinar a velocidade tal como no caso do cálculo da velocidade no código 2.

Com o cálculo da velocidade passamos à determinação da localização. A classe *WSG84* contém métodos que permitem obter a localização atual. O método *findPointAtDistanceFrom* estima a localização atual mediante a localização anterior, o azimute atual (baseado no magnetómetro) e a distância percorrida e devolve a localização atual.

```

public static double[] findPointAtDistanceFrom(double _latitude,
double _longitude, double _azimuth, double _distancia)
{
    double[] retval = new double[2];
    // Distância dividida pelo raio da Terra
    double d = _distancia / EarthRadius;
    // Passagem de graus para radianos
    double lat1 = Math.toRadians(_latitude);
    double lon1 = Math.toRadians(_longitude);
    double lat = Math.asin((Math.sin(lat1) * Math.cos(d))
        + (Math.cos(lat1) * Math.sin(d) *
            Math.cos((_azimuth))));

    double lon = lon1
        + Math.atan2(Math.sin((_azimuth)) * Math.sin(d)
            * Math.cos(lat1),
            Math.cos(d) - Math.sin(lat1) * Math.sin(lat));

    retval[0] = lat;
    retval[1] = lon;
    return retval;
}

```

Código 3 - Função para determinar localização baseado na localização anterior, no valor do azimute atual e distância percorrida.

Tem como argumentos a latitude e longitude anteriores, o azimute atual obtido do magnetómetro que providencia a direção e a distância percorrida calculada pela integração do acelerómetro.

6.4 Resultados

Por forma a validar o sistema e as funcionalidades descritas, foi definido um conjunto de cenários representativos de algumas situações onde o sistema pode ser de grande utilidade. Estes cenários representam percursos efetuados e em todos eles foram alterados parâmetros que a aplicação, desenvolvida no âmbito deste trabalho, permite. Desta forma foi possível analisar e adaptar o sistema.

Efetuuou-se o registo dos dois sistemas (INS e GPS) simulando falhas do GPS. Todas as coordenadas foram guardadas num ficheiro KML gerado pela aplicação ao longo do trajeto.

Os trajetos efetuados também podem ser visualizados posteriormente, num mapa, sob a forma de linhas poligonais. As linhas vermelhas sinalizam a localização obtida do INS, a azul a obtida do GPS e a verde está sinalizado o trajeto real.

Os seguintes cenários foram definidos para testar o sistema:

1. Cenário 1: Um trajeto em linha reta em piso plano, com curva à direita e com inclinação descendente acentuada após a curva. O GPS foi desligado após 54 metros percorridos do total de 215 metros do trajeto;
2. Cenário 2: Um trajeto em linha reta com curva à esquerda e entrada numa rotunda com piso inclinado ascendente. O total do trajeto são 1107 metros e simulou-se a falha do sinal GPS aos 136 metros percorridos;
3. Cenário 3: Um circuito apenas determinado pelo INS com a localização inicial obtida pelo GPS. O circuito apresenta uma linha reta com inclinação ascendente, curva à direita em piso plano mas irregular e novamente curva à direita com um piso ligeiramente inclinado de forma descendente.
4. Cenário 4: Um circuito caracterizado por uma situação comum. Um trajeto com entrada num parque subterrâneo. O veículo faz o circuito em redor do estacionamento num parque caracterizado pelas imensas perturbações magnéticas.

6.4.1 Cenário 1

Para o primeiro cenário, foram aplicadas as configurações pré-definidas da aplicação. O dispositivo móvel foi colocado com um ângulo de *pitch* de aproximadamente 45° sobre o tablier do veículo automóvel. Foi utilizada a frequência máxima dos sensores, o filtro complementar como orientação e o filtro passa baixo com $\alpha=0.4$ para filtrar o ruído do acelerómetro. A gravidade é determinada com auxílio do giroscópio de forma isolada (método 1 descrito na secção 5.6). O resultado pode ser visualizado na figura 70, que mostra um percurso num trajeto com um total de 215 metros. A linha azul do percurso assinalado na

figura 70, representa a determinação da localização e o percurso efectuado com o auxílio do GPS, a linha vermelha o percurso determinado pelo INS e a verde o percurso real. A bola azul pretende referenciar o local onde foi simulada uma falha na obtenção da localização pelo GPS.

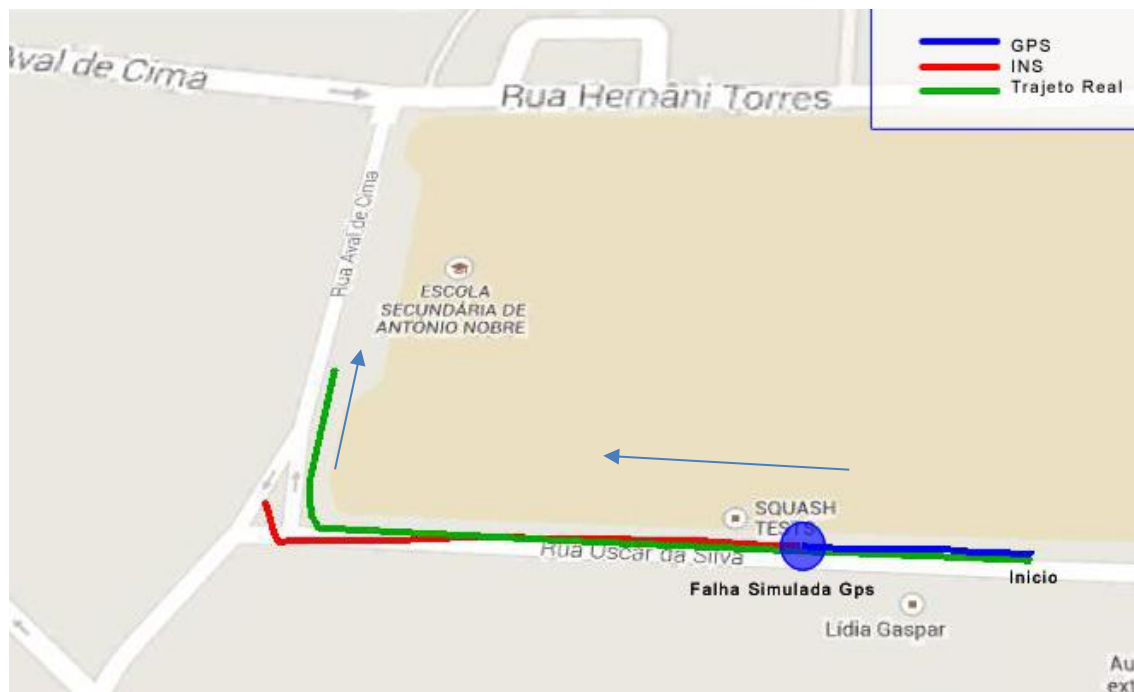


Figura 70 – Trajeto relativo ao cenário 1.

Aos 54 metros de distância do trajeto, foi simulada uma falha do GPS, local sinalizado na figura com uma bola azul. O INS a partir desse momento responsabilizou-se pela determinação da localização. Quase no final do trajeto, realizou-se uma curva à direita a qual não foi bem acompanhada pelo INS apesar de a distância até à curva ter sido bem determinada, contribuindo para isso uma boa resolução da velocidade.

Uma das causas para a curva ter sido mal delineada poderá estar no ângulo de *pitch* do dispositivo móvel, causando desvios na determinação da direção pelo magnetómetro, cujo resultado advém da colaboração dos sensores magnetómetro e acelerómetro na construção da matriz de rotação. Outra razão tem a ver com possíveis interferências magnéticas no magnetómetro.

Já a distância percorrida após a curva foi influenciada por um conjunto de fatores, nomeadamente o *drift* do giroscópio na determinação da gravidade, tendo em conta a inclinação do piso após a curva, o *bias* do acelerómetro e a obtenção inicial dos limites superior e inferior a partir dos quais a aceleração começa a contar para a determinação da velocidade. O dispositivo móvel quando inclinado com um ângulo de *pitch* superior a 10°, a compensação da inclinação pelo acelerómetro e giroscópio não é determinada da melhor forma com os parâmetros utilizados.

6.4.2 Cenário 2

Para o segundo cenário, mantiveram-se os mesmos parâmetros mas num trajeto maior de aproximadamente 1,1Km por forma a corroborar as conclusões do cenário anterior. (figura 71)

A linha azul do percurso assinalado na figura 71, representa a determinação da localização e o percurso efectuado com o auxílio do GPS, a linha vermelha o percurso determinado pelo INS e a verde o percurso real. A bola azul pretende referenciar o local onde foi simulada uma falha na obtenção da localização pelo GPS.

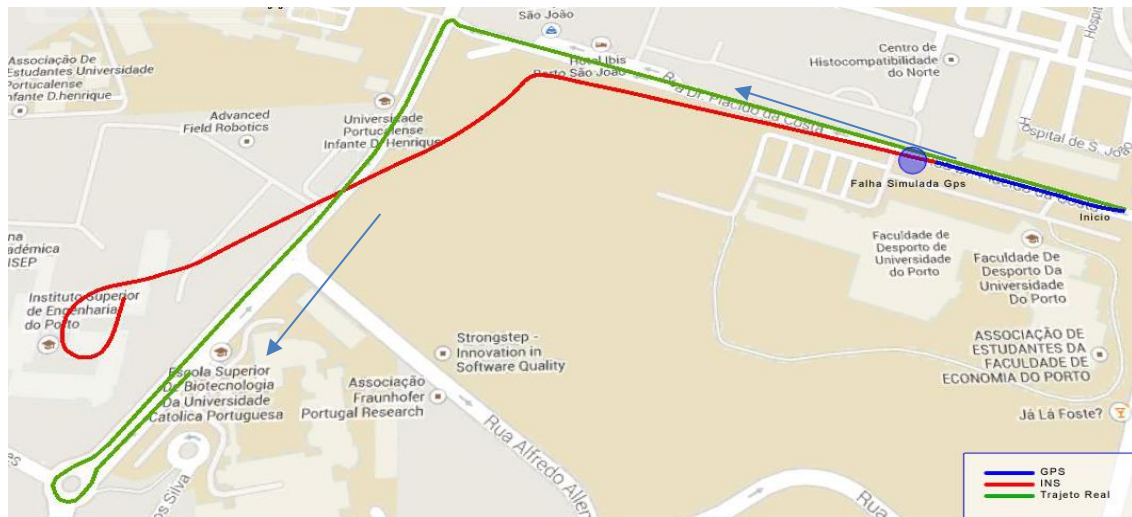
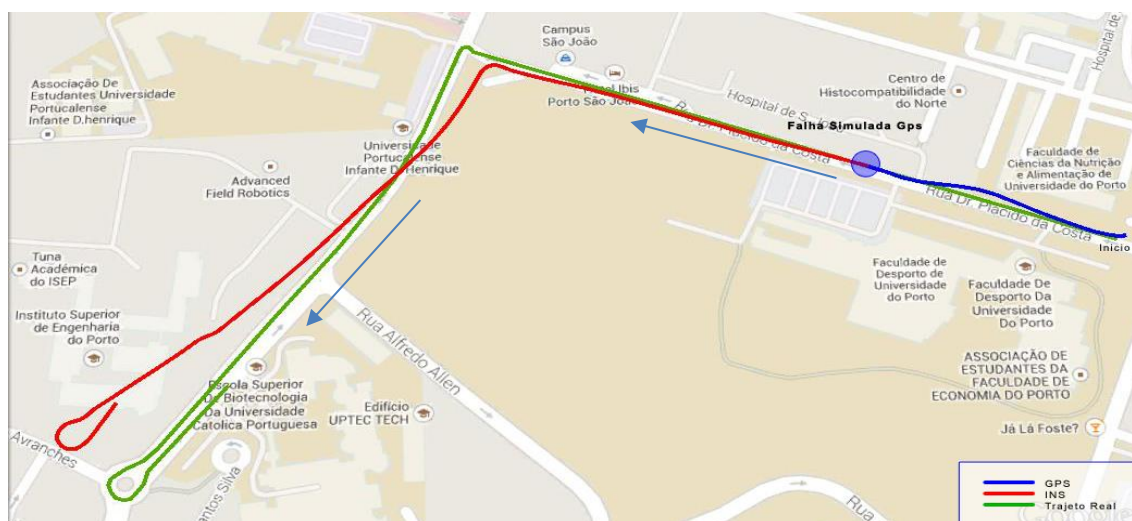


Figura 71 - Trajeto relativo ao cenário 2.

Após analisar os trajetos, tudo leva a crer que a direção determinada pelo magnetómetro encontra-se de alguma forma adulterada. Se fosse meramente um caso de desvio, a direção obtida seria acompanhada por um *offset* ou *bias*, mas neste caso o magnetómetro não parece estar a determinar da melhor forma a direção e sentido do campo magnético terrestre. Tendo em conta que o comportamento do magnetómetro foi testado fora do veículo com algum sucesso, podemos dizer que o veículo poderá estar a gerar campos magnéticos na sua proximidade constatado pelo comportamento errático em determinadas direções, corrompendo o resultado final.

Para validar esta situação, o dispositivo foi colocado na zona central do veículo para a realização deste e dos restantes cenários. A figura 72 mostra o trajeto após colocação do dispositivo móvel na zona traseira do veículo, sujeito a menor perturbação magnética.



6 Desenvolvimento da solução

Figura 72 - Trajeto relativo ao cenário 2 após colocação do dispositivo móvel no centro do veículo para limitar as interferências magnéticas.

Ao analisar o percurso, verificam-se dois aspetos importantes. O primeiro tem a ver com a estabilização da localização GPS no início do trajeto, visto que o mesmo desenha uma curva. Este facto está relacionado com a localização mais resguardada (centro do veículo) do dispositivo móvel, provocando uma menor precisão na obtenção do sinal GPS.

O segundo aspeto a ter em conta é a melhoria existente na determinação da direção constatada na reta e na primeira curva à esquerda. Na reta quase não existiu o desvio provocado pela interferência magnética anterior e a curva foi bem determinada.

6.4.3 Cenário 3

O cenário 3 passa-se na mesma zona onde o trajeto do cenário 1 foi efetuado. Desta vez, por forma a avaliar o comportamento do magnetómetro, o dispositivo foi colocado na zona central do veículo por forma a evitar os campos magnéticos existentes na frente do veículo.

Os parâmetros da aplicação foram alterados e neste cenário o método de determinação da gravidade escolhido foi a utilização da matriz de rotação, processo definido na secção 3.2.2. No filtro complementar, foi definido que o giroscópio teria maior peso relativamente ao magnetómetro na obtenção da direção. A figura 73 mostra o resultado do trajeto com um total de 700 metros. A linha azul do percurso assinalado na figura 73, representa a determinação da localização e o percurso efectuado com o auxílio do GPS, a linha vermelha o percurso determinado pelo INS e a verde o percurso real. A bola azul pretende referenciar o local onde foi simulada uma falha na obtenção da localização pelo GPS.

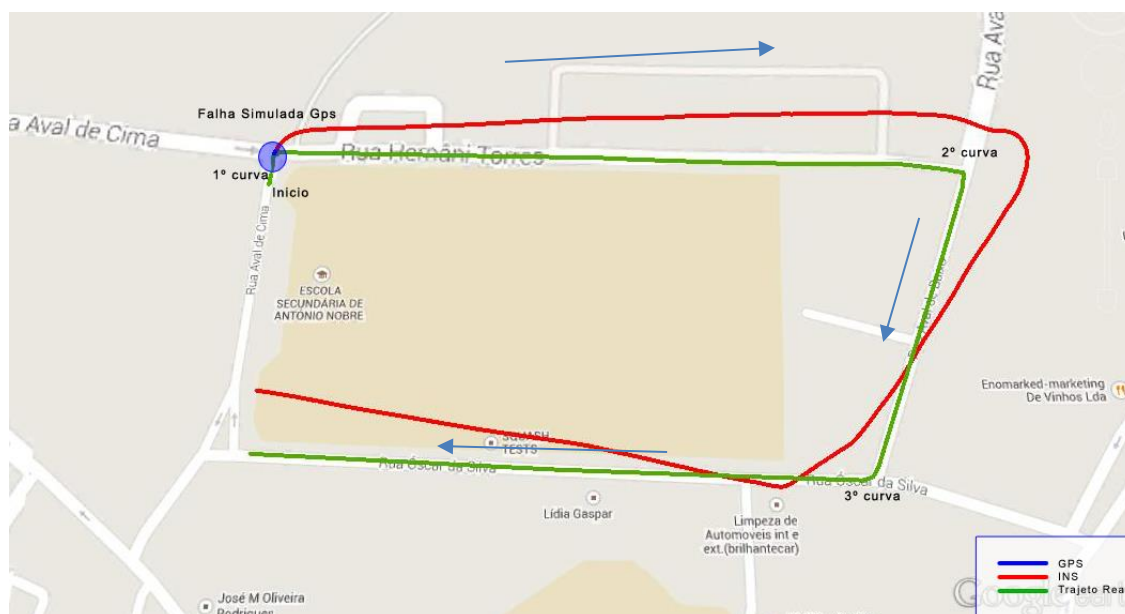


Figura 73 - Trajeto relativo ao cenário 3.

Desde logo sobressaem dois pontos importantes. Primeiro o facto de o GPS estar colocado no interior do veículo, o erro de localização inicial é maior devido à menor precisão, visto que o início do trajeto é antes de chegar à primeira curva (sinalizada na figura 73), mas o GPS

marcou no início do trajeto em cima da curva que por sua vez fez com que o trajeto do INS marcasse o início da curva mais afastada.

O segundo ponto é o facto da grande maioria das interferências causadas pelos campos magnéticos do veículo terem sido anuladas, pois a definição das curvas e direção é melhor acompanhada pelo INS. A velocidade determinada também melhorou devido ao uso da matriz de rotação no cálculo da gravidade nos eixos por forma a determinar a aceleração linear.

6.4.4 Cenário 4

Um cenário desde logo interessante para demonstrar o sistema aqui tratado num ambiente subterrâneo onde o sinal de GPS é totalmente inexistente. Este trajeto evidencia a regulação disponível na aplicação do filtro complementar, ou seja, o quanto queremos que seja o giroscópio ou o magnetómetro a ditar a direção, visto que o ambiente subterrâneo onde o mesmo foi testado encontra-se sujeito a imensas perturbações magnéticas.

Para este cenário fizemos 2 trajetos com valores diferentes para os diferentes pesos dos sensores no filtro. No primeiro trajeto o magnetómetro assume o peso maior na obtenção da direção o que originou um percurso conforme a figura 74. A linha azul do percurso assinalado na figura 74, representa a determinação da localização e o percurso efectuado com o auxílio do GPS, a linha vermelha o percurso determinado pelo INS e a verde o percurso real. A bola azul pretende referenciar o local onde foi simulada uma falha na obtenção da localização pelo GPS.

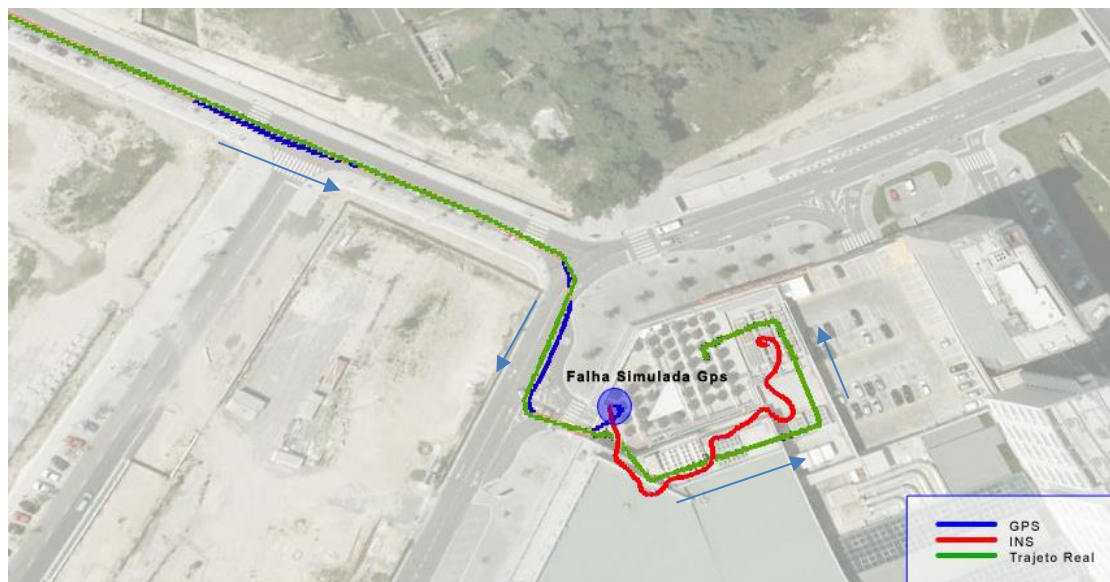


Figura 74 - Trajeto relativo ao cenário 4.

No percurso obtido pelo INS dentro da garagem, nota-se uma semelhança entre este e o trajeto real. Ainda assim as curvas acentuadas em algumas zonas do percurso dentro da garagem obtido pelo INS demonstram os efeitos dos campos magnéticos existentes no local, efeitos esses que provocam alterações na direção obtida pelo magnetómetro. No local é possível encontrar cablagem e equipamento eléctrico em excesso.

6 Desenvolvimento da solução

Para atenuar este efeito, o próximo trajeto foi efetuado com o filtro complementar configurado para dar mais ênfase ao giroscópio. Dando mais ênfase ao giroscópio, conseguimos suprimir as variações provocadas pelo campo magnético, visto que o giroscópio fará grande parte do trabalho na determinação da direção.

A figura 75 demonstra o percurso estimado pelo INS quando parametrizado de forma a dar mais relevo aos valores obtidos pelo giroscópio em detrimento dos valores obtidos pelo magnetômetro. Pode-se verificar uma maior aproximação ao percurso real. Obviamente a longo prazo a direção acaba por perder exatidão devido ao *drift* do giroscópio não estar a ser compensada pelos sensores acelerómetro e magnetómetro.

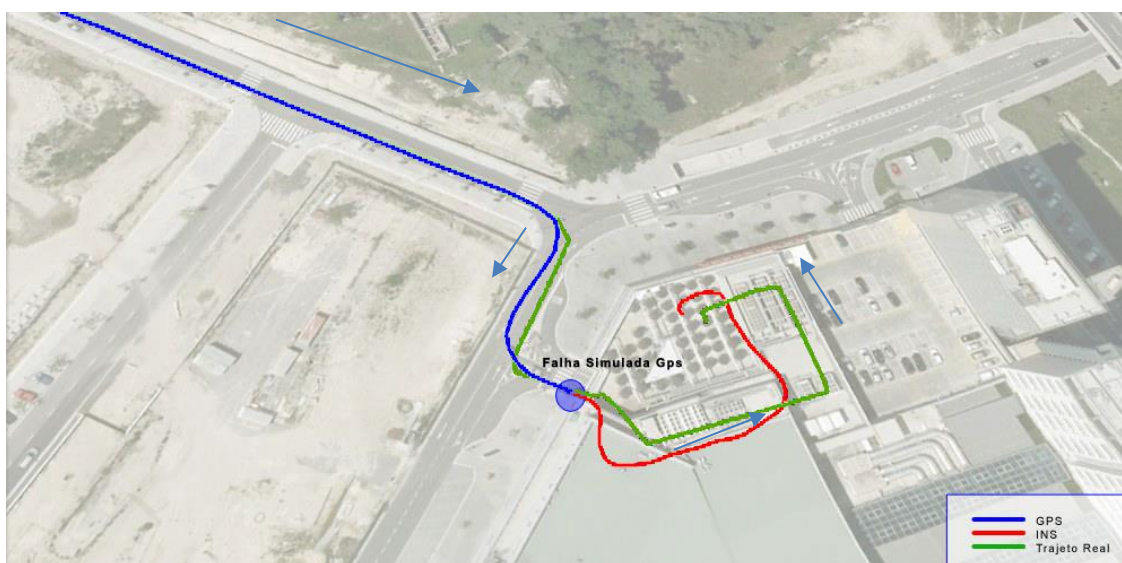


Figura 75 - Trajeto relativo ao cenário 4.

Este resultado leva à conclusão que o magnetómetro não oferece as melhores condições para a obtenção de um percurso estimado pelo INS. O magnetómetro pode e deve ser usado para determinar a orientação inicial mas o seu uso deve ser limitado. O filtro complementar pode ser ajustado para que este uso seja limitado.

7 Conclusão

O propósito deste trabalho foi o estudo e desenvolvimento de algoritmos capazes de estimar a localização de um dispositivo móvel em caso de falha do GNSS, recorrendo aos sensores MEMS de baixo custo que acompanham muitos dos dispositivos móveis hoje em dia.

Apesar do GPS ser um sistema em pleno funcionamento e com imensas aplicações, este continua a ser um sistema falível que pode ser melhorado recorrendo a outros sistemas de navegação.

A base do desenvolvimento de um sistema de navegação inercial consiste na equação do movimento. Esta equação do movimento traduz as acelerações medidas, em velocidade. Os sistemas de navegação por inércia só por si, não conseguem manter uma boa precisão da localização em longos períodos de tempo.

Das análises efetuadas relativas ao sistema desenvolvido, foi possível verificar que os componentes de um sinal relativo a qualquer um dos sensores usados para o efeito, contém ruídos e outros elementos que não são do interesse para o objetivo final de manter a localização. Foi necessário recorrer a filtros para atenuar o ruído e para disponibilizar uma solução de orientação com os sensores acelerómetro, magnetómetro e giroscópio. O filtro desenvolvido dá pelo nome de filtro complementar, pois os sensores complementam-se nas suas mais-valias.

Apesar do Android disponibilizar técnicas de obtenção da orientação, estas não têm em conta os erros de calibração, erros esses que integrados provocam estimativas irrealistas.

O filtro complementar proposto vai de encontro a uma solução de orientação que tem por objetivo a obtenção de uma orientação usando o acelerómetro para a obtenção da gravidade, o magnetómetro para usar como compasso e um giroscópio cuja integração da velocidade angular permitirá obter a rotação do vetor gravidade e da direção, este último mais preciso com tempos de resposta muito curtos. Apesar de os primeiros dois terem como principal desvantagem o ruído, já a desvantagem do giroscópio é o *drift*. Para evitar tanto o *drift* do giroscópio como o ruído do magnetómetro e acelerómetro, o output do giroscópio é aplicado em curtos espaços de tempo, enquanto os dados do acelerómetro e magnetómetro são aplicados em longos intervalos de tempo. Este processo equivale a filtrar os dados do

7 Conclusão

acelerómetro e magnetómetro com um filtro passa baixo e filtrar os dados do giroscópio com um filtro passa alto.

Conseguiu-se demonstrar que em curtos intervalos de tempo aquando da falha na obtenção da localização por GPS, é possível o dispositivo móvel manter uma estimativa coerente da localização mediante a parametrização dos coeficientes que compõem o processo de navegação por inércia.

No entanto, existem diversos fatores que alteram as expectativas do proposto e que podem causar erros de estimativa. O magnetómetro, quando na presença de campos magnéticos tais como os encontrados no interior e ambiente externo do veículo utilizado, teve um comportamento menos correto, provocando desvios de trajeto consideráveis. Este comportamento pode ser melhorado dando maior ênfase ao giroscópio conforme demonstrado. Também a obtenção da gravidade com o dispositivo em movimento, tornou-se uma tarefa de difícil concretização. Apresentaram-se três métodos diferentes de obtenção da gravidade com o dispositivo em movimento, disponibilizados pela aplicação desenvolvida para análise e todos eles apresentaram resultados ligeiramente diferentes consoante o tipo de piso e sinuosidade do percurso. Por fim, o método de obtenção da gravidade utilizado com melhores resultados foi a transformação do vetor gravidade recorrendo à matriz rotação obtida com o filtro complementar.

Outro aspeto que deverá ser realçado é a necessidade do sistema, quando em modo INS apenas, efetuar paragens para recalibração do vetor gravidade, caso contrário as estimativas de velocidade perdem rigor.

Muitos dos trabalhos investigados aquando da análise do estado da arte, referem o filtro de Kalman como um estimador ótimo. No entanto o filtro de Kalman obriga a modelar muito bem o sistema e definir as equações que caracterizam o seu comportamento tornando o processo complexo. A exigência de capacidade computacional, hoje em dia, acaba por ser anulada com a existência de dispositivos multi-cores capazes de dar resposta aos requisitos do filtro.

Relativamente aos resultados obtidos, comparando com os resultados dos sistemas estudados, o sistema implementado oferece uma solução baseada num algoritmo que exige pouca capacidade computacional, como no caso do filtro complementar. Outros sistemas usam o filtro de Kalman e implementam sensores com um maior grau de fiabilidade e frequência e talvez por isso resultem num melhor resultado final. O que se pretendeu demonstrar foi a exequibilidade do sistema INS/GPS num dispositivo móvel com sensores de baixo custo.

Em termos de considerações futuras, antes do sistema e seus algoritmos aqui apresentados poderem ser implementados em cenários reais, alguns aspetos deverão ser tidos em conta.

Falou-se em integração INS/GPS mas para este trabalho o único aspeto tido em conta foi a correção da velocidade por parte do GPS e a manutenção da localização num cenário de integração *loosely coupled*. Poder-se-á elaborar um modelo que passa por colocar ambas as soluções de navegação a referenciar-se num único sistema de coordenadas e assim estimar de forma mais precisa outros elementos tais como a direção ou mesmo a gravidade recorrendo a arquiteturas que necessitam de maior grau de conhecimento entre módulos tais como a *Tightly coupled* e *Deeply Coupled*.

Outro aspeto que poderia melhorar o sistema apresentado, seria a inclusão de um algoritmo de *map matching* referido no estado da arte. Os algoritmos de correspondência em mapa geralmente consistem em duas etapas distintas: identificação da ligação da rua onde o dispositivo móvel muito provavelmente estará a viajar e uma estimativa da posição do veículo na rua seleccionada. Este algoritmo é baseado no teorema de Bayes, decorrência imediata do teorema da probabilidade total de forma a seleccionar o segmento de rua onde o veículo está a passar.

Por fim, a inclusão de um método de adaptação do filtro complementar a condições ambientais diferenciadas melhoraria a obtenção da direcção por parte do magnetómetro ou giroscópio. Conforme demonstrado na subsecção 6.4.4, a obtenção da direcção é maioritariamente assumida pelo magnetómetro. No entanto, em condições onde o sistema deteta perturbações magnéticas, o filtro poderia adaptar-se dando mais ênfase aos dados do giroscópio, realizando uma trajetória imune aos campos magnéticos.

Referências

- [Angrisano, et al., 2010] Angrisano, A., Petovello, M. & Pugliano, G., 2010. *GNSS/INS Integration in Vehicular Urban Navigation*. Portland, s.n., pp. 1505-1512.
- [Angrisano, et al., 2012] Angrisano, A., Petovello, M. & Pugliano, G., 2012. Benefits of Combined GPS/GLONASS with Low-Cost MEMS 7 IMU for Vehicular Urban Navigation. *Sensors*.
- [Assis, 2010] Assis, P. M., 2010. *Sistema de rastreamento de veículos para empresas de transporte utilizando navegação por satélite*, Brasília: UniCEUB.
- [Brown, et al., 1997] Brown, R. G. & Hwang, P. Y., 1997. *Introduction to Random signals and applied Kalman filtering. 3rd Ed.*. 3rd Edition ed. New York: John Wiley & Sons.
- [Cardeira, 2009] Cardeira, B. M. S. C., 2009. *Arquitecturas para Navegação Inercial/GPS com aplicação a veículos autónomos*, Lisboa: Instituto Superior Técnico.
- [Colton, 2007] Colton, S., 2007. *The Balance Filter*. [Online] Available at: <http://web.mit.edu/scolton/www/filter.pdf> [Acedido em 23 07 2012].
- [Davidson, et al., 2010] Davidson, P., Hautamäki, J., Collin, J. & Takala, J., 2010. *Improved Vehicle Positioning in Urban Environment through Integration of GPS and Low-Cost Inertial Sensors*, s.l.: s.n.
- [Davidson, et al., 2009] Davidson, P., Vazquez, M. & Piche, R., 2009. *Uninterrupted portable car navigation system using GPS, map and inertial sensors data*. s.l., s.n., pp. 836 - 840.
- [ESA, 2008] ESA, 2008. *European Space Agency*. [Online]

Available at: http://www.esa.int/esaNA/SEMGVUZXUFF_index_0.html
[Acedido em 02 2012].

- [Figueiredo, et al., 2007] Figueiredo, L. J., Gafaniz, A. R., Lopes, G. S. & Pereira, R., 2007. *Aplicações de Acelerómetros*, Lisboa: Instituto Superior Técnico.
- [Filho, 2007] Filho, E. A. M., 2007. *Navegação através de um sistema integrado GPS/INS baseado em IMU não giroscópica*, São José dos Campos: Ministério da Ciência e Tecnologia - Instituto nacional de pesquisas espaciais.
- [Gartner, 2013] Gartner, 2013. *Gartner Says Smartphone Sales Grew 46.5 Percent in Second Quarter of 2013 and Exceeded Feature Phone Sales for First Time*. [Online]
Available at: <http://www.gartner.com/newsroom/id/2573415>
[Acedido em 08 2013].
- [Georgy, 2010] Georgy, J. N. A., 2010. *Low-Cost Post-Mission Positioning and Orientation Solution for Land-Based Mobile Mapping Using Nonlinear Filtering*. Portland, s.n., pp. 991-1003.
- [Giremus & Tourneret, 2006] Giremus, A. & Tourneret, J.-Y., 2006. *Controlling particle filter regularization for gps/ins hybridization*, Toulouse: IRIT/ENSEEIH.
- [Harch, 2006] Harch, N., 2006. *Testing Mapping Grade GPS Carrier Phase Accuracy*, Queensland: University of Southern Queensland.
- [Higgins, 1975] Higgins, W. T., 1975. *A Comparison of Complementary and Kalman Filtering*, Arizona: IEEE.
- [Hunter, et al., 2010] Hunter, A. et al., 2010. *Portable Navigation System*. Canada, Patente N.º 20100198511.
- [J. Sum, 1996] J. Sum, C. L. a. L. C., 1996. *Extended kalman filter in recurrent neural network training and pruning.*, s.l.: s.n.
- [Jorge & Sancho, 2009] Jorge, R. & Sancho, C., 2009. *Sistema de controlo distribuido baseado em barramento CAN*, Aveiro: s.n.
- [Kam, et al., 1997] Kam, M., Zhu, X. & Kalata, P., 1997. Sensor Fusion for Mobile Robot Navigation. *Proceedings of the IEEE*, pp. 108-119.
- [Kraft, 2000] Kraft, M., 2000. *Micromachined Inertial Sensors - State of the Art and a Look into the Future*, Highfield, Southampton, SO17 1BJ: University of Southampton.

- [Kumar, 2004] Kumar, V. N., 2004. *Integration of Inertial Navigation System and Global Positioning System Using Kalman Filtering*, Bombay: INDIAN INSTITUTE OF TECHNOLOGY.
- [Lachapelle & Petovello, 2007] Lachapelle, G. & Petovello, M., 2007. *GNSS Solutions*. [Online] Available at: [http://www.insidegnss.com/auto/JanFeb07GNSSolutions%20\(secured\).pdf](http://www.insidegnss.com/auto/JanFeb07GNSSolutions%20(secured).pdf) [Acedido em 01 2012].
- [Lehtinen, 2001] Lehtinen, A., 2001. *Doppler Positioning with GPS*. TAMPERE: s.n.
- [Lima, 2005] Lima, S. R. S. d., 2005. *Integração GPS/INS utilizando sensores inerciais baseados em sistemas microelectromecânicos (MEMS)*, CURITIBA: Universidade Federal do Paraná.
- [Matos, 2007] Matos, J., 2007. *Geodesia e Sistemas de Referência*. [Online] Available at: https://dSPACE.ist.utl.pt/bitstream/2295/187023/1/Geodesia%20e%20Sistemas%20de%20Referencia_v1_1.pdf [Acedido em 02 02 2012].
- [Monico, 2000] Monico, J. F. G., 2000. *Posicionamento pelo NAVSTAR-GPS, Descrição, fundamentos e aplicações*. 1 ed. s.l.:UNESP.
- [Natarajan, 1994] Natarajan, B. K., 1994. *A general technique for filtering random noise*., Palo Alto: Hewlett-Packard..
- [Neto, et al., 2009] Neto, A. A., Macharet, D. G., Campos, V. C. d. S. & Campos, M. F. M., 2009. Adaptive complementary filtering algorithm for mobile robot localization. vol.15(Computer Vision and Robotic Laboratory).
- [Open Source Project, 2012] Open Source Project, A., 2012. *Android Dev Guide*. [Online] Available at: <http://developer.android.com/guide/index.html>
- [Perez, s.d.] Perez, C., s.d. *TA Mobile Computing Course*, s.l.: s.n.
- [Petovello, et al., 2007] Petovello, M., Lachapelle, G., Gebre-Egziabher, D. & Wieser, A., 2007. Weighting GNSS Observations and Variations of GNSS/INS Integration. *InsideGNSS*.
- [Positioning, 2012] Positioning, T., 2012. *TPI technology*. [Online] Available at: http://www.trustedpositioning.com/TPI_technology.html [Acedido em 22 01 2012].
- [Racelogic, 2011] Racelogic, 2011. *VBOX 3i User Guide*. [Online] Available at:

http://www.racelogic.co.uk/downloads/vbox/Manuals/Data_Loggers/RLVB3i_Manual%20-%20English.pdf

[Acedido em 01 12 2011].

- [Ristic, 2004] Ristic, B. A. S. a. G., 2004. *Beyond the Kalman Filter - Particle Filters for Tracking Applications*, s.l.: Artech House.
- [Sandoval, et al., 2004] Sandoval, G. L., Chávez, E. E. & Caballero, J. P., 2004. A development platform and execution environment for mobile applications. *CLEI ELECTRONIC JOURNAL*, Volume 7.
- [Schmidt, 2011] Schmidt, G. T., 2011. *INS/GPS Technology Trends*. Lexington(Massachusetts): NATO Research & Technology Organisation.
- [Schmidt & Phillips, 2011] Schmidt, G. T. & Phillips, R. E., 2011. *INS/GPS Integration Architectures*. s.l.:NATO Research & Technology Organisation.
- [Schultz, 2006] Schultz, C. E., 2006. *INS and GPS integration*, Lyngby: s.n.
- [Shome, et al., 2012] Shome, S. K. et al., 2012. Performance Evaluation of Different Averaging based Filter Designs. *International Journal of Signal Processing*, 5(Image Processing and Pattern Recognition).
- [Skog, 2007] Skog, I., 2007. *GNSS-aided INS for land vehicle positioning and navigation*, Stockholm: KTH (Royal Institute of Technology).
- [Smith, 1997] Smith, S. W., 1997. *The Scientist and Engineer's Guide to Digital Signal Processing*. Caliornia: California Technical Publishing.
- [Sousa, 2004] Sousa, C. M. F. d., 2004. *A Integração do Sistema GPS/INS para a Monitorização da Linha de Costa do Litoral do Algarve*, Lisboa: Faculdade de Ciências - Universidade de Lisboa.
- [Sousa, 2005] Sousa, C. R., 2005. *INTERFERIDORES DE GPS: ANÁLISE DO SISTEMA E DE POTENCIAIS FONTES DE INTERFERÊNCIA*, Rio de Janeiro: Instituto Militar de Engenharia.
- [Sousa, 2006] Sousa, O., 2006. *Threads*. [Online] Available at: <http://www.dei.isep.ipp.pt/~orlando/so2/threads.htm> [Acedido em 2012].
- Su, C.-M.et al., 2010. *Sensor-Aided Personal Navigation Systems for Handheld Devices*. s.l., s.n., pp. 533-541.
- [Syed, 2009] Syed, Z. F., 2009. *Design and Implementation Issues of a Portable Navigation System*, Calgary: THE UNIVERSITY OF CALGARY.

- [Thiagarajan, et al., 2011] Thiagarajan, A. et al., 2011. *Accurate, Low-Energy Trajectory Mapping for Mobile Devices*. California, USENIX - Advanced Computing Systems Association.
- [Thiagarajan, et al., 2009] Thiagarajan, A. et al., 2009. *VTrack: Accurate, Energy-Aware Road Traffic Delay Estimation Using Mobile Phones*. California, s.n.
- [Welch & Bishop, 2006] Welch, G. & Bishop, G., 2006. *An Introduction to the Kalman Filter*, s.l.: University of North Carolina.
- [White, et al., 2000] White, C. E., Bernstein, D. & Kornhauser, A. L., 2000. *Some map matching algorithms for personal navigation assistants*, Princetown: s.n.
- [Woodman, 2007] Woodman, O. J., 2007. *An introduction to inertial navigation*, Cambridge: University of Cambridge.