Multi-sensor systems and models for high accuracy indoor positioning

Petrina Constantinou

A thesis presented for the degree of

Doctor of Philosophy

Centre for Transport Engineering and Modelling

Department of Civil and Environmental Engineering

Imperial College London

Statement of originality

In the process of undertaking this research I received supervisory support from my supervisor Professor Washington Yotto Ochieng and assistant supervisor Dr. Felix Feng, in line with PhD supervision procedures of Imperial College London. The data used in this thesis were captured in collaboration with Dr. Felix Feng. The subsequent analysis and data processing was a result of my efforts.

Given the above, I declare that apart from the supervisory support, the work presented in this thesis has been carried out by myself. Any work, presented in this thesis, that is not my own, has been appropriately cited.

Petrina Constantinou
29^{th} February 2024

The copyright of this thesis rests with the author. Unless otherwise indicated, its contents are licensed under a Creative Commons Attribution-Non Commercial 4.0 International Licence (CC BY-NC). Under this licence, you may copy and redistribute the material in any medium or format. You may also create and distribute modified versions of the work. This is on the condition that: you credit the author and do not use it, or any derivative works, for a commercial purpose. When reusing or sharing this work, ensure you make the licence terms clear to others by naming the licence and linking to the licence text. Where a work has been adapted, you should indicate that the work has been changed and describe those changes. Please seek permission from the copyright holder for uses of this work that are not included in this licence or permitted under UK Copyright Law

Acknowledgements

I would like to express eternal gratitude and appreciation to my supervisor, Professor Washington Yotto Ochieng for offering me this opportunity. His insightful guidance, patience and support through these years. Without his support this thesis could not have been accomplished. In addition, I am immensely grateful for the supervision of Dr. Felix Feng. His guidance, support and patience through this journey as well as his invaluable assistance in data collection was pivotal for the completion of this thesis.

Thank you to Dr. Panagiotis Angeloudis for giving me the opportunity to work with him and develop my technical skills and Dr. Jose Escribano, for his invaluable advice and discussion around my work.

In words of gratitude far fewer than they deserve, I would like to thank my parents, Kaiti and Kostas, for being by my side through the ups and downs, always with unconditional love and dedication, supporting me in every way possible. I would like to thank my boyfriend, George for all the love, support, and patience. To my mother, father and George, the amount of effort put in this thesis is as much mine as it is yours.

A big thank you to my Imperial friends, thank you for being there daily through the good and the bad, thank you for all the lunch breaks that always made my day better. Special thanks for the support, especially close to the end: Karim, Leah, Surabhi, He-in, Napameth, Renos, Eduardo and Leandro. I would also like to thank my friends outside academia, for their continued support. This journey would not have been the same without you.

Thank you also to the EPSRC and Imperial College London both for their financial support as well as their administrative help throughout this amazing journey I had over these years.

Dedication

To my parents, Kostas and Kaiti and my boyfriend George, who patiently stood by me.

Abstract

Technological advances over the years, gave rise to location based services (LBS) and their demand in terms of higher performance, measured by the required navigation performance parameters including accuracy and availability. Until recently this has been prominent in outdoor environments, where global navigation satellite systems (GNSS) facilitated outdoor LBS, triggering the rise in their popularity and making GNSS a de facto positioning technology in outdoor environments. Advances in position navigation and timing systems, and communication technologies, are gradually increasing the potential of high performance indoor positioning systems (IPS), increasing the demand for more indoor LBS. The complexity of indoor environments, presents serious challenges to IPS, and currently there is no consensus on a generic positioning technology for indoor environments.

Currently the approach taken involves identifying applications, specification of requirements and architectural design. However, a critical consideration of IPS design should be a detailed understanding of the 'operations'. This is to enable a complete identification of functions that require positioning.

Therefore, this thesis firstly takes the approach that identifies indoor applications and their requirements. It then specifies in detail the operational functional system architecture of a cognitive agent, and identifies these functions that require positioning within them, the location based functions (LBF). This enables confirmation and augmentation where necessary to meet the applications and their requirements. The thesis then explores the technologies required to implement LBF (i.e. physical architecture). Candidate technologies are reviewed and based on accuracy performance, UWB and INS are analysed individually and in integration. A UWB/INS integration using UWB-derived speed estimates, measurement corrections and filtering is developed. The results show the proposed integration improves accuracy (95%) performance up to 65.5% compared to UWB-standalone system and 36.6% compared to UWB/INS integration without UWB-derived speed estimates.

Contents

1 Introduction **19**

1.1 Background 19

1.2 Aims and objectives 21

1.3 Thesis structure 22

2 Location based services and their required navigation performance **24**

2.1 Operating environments of a moving agent 24

2.2 Location Based Services 27

2.3 Required Navigation Performance parameters 35

2.3.1 Service requirements 35

2.3.2 Required Navigation Performance parameters 37

2.3.3 Required Navigation performance targets 40

2.4 Summary 42

3 State-of-the-art functional system architectures of dynamic autonomous agents **44**

3.1 Cognitive agent 45

3.1.1 Situational Awareness 47

3.2 State of the art Functional System Architectures of Autonomous Vehicles . 48

3.2.1 Stanley entry in the DARPA Grand Challenge 2005 49

3.2.2 Boss entry in the DARPA Urban Challenge 2007 51

3.2.3 Carnegie Mellon University autonomous vehicle 53

3.2.4 A1 autonomous vehicle 55

3.2.5 Ulm University autonomous vehicle 58

3.2.6 Recent advances 59

3.3 Comparison of existing architectures 61
3.4 Levels of Decision and Planning 65
3.5 Summary 66

4 Enhanced functional system architecture for dynamic autonomous agents 69
4.1 Methodology 70
4.2 Proposed functional system architecture 71
4.2.1 Sensing, Communication & Control 73
4.2.2 Situational Awareness 74
4.2.3 Decision and Planning 81
4.3 Transferability of functional system architecture 83
4.3.1 Human driven vehicles 83
4.3.2 Visually impaired people 84
4.4 Location based functions within the functional system architecture 88
4.4.1 Sensing, Communication and Control 88
4.4.2 Situational awareness 89
4.4.3 Decision and planning 90
4.5 Summary 91

5 State-of-the-art indoor positioning and selection of candidate methods 95
5.1 Positioning Methods using radio waves and inertial systems 95
5.1.1 Geometric methods 97
5.1.2 Scene Analysis 103
5.1.3 Proximity 104
5.1.4 Dead Reckoning 104
5.2 Signal propagation vulnerabilities 105
5.2.1 Dilution of precision 106
5.3 State-of-the-art positioning systems 108
5.3.1 Beacon-based systems 109

5.3.1.1 Ultra Wideband . . . 109
5.3.1.2 Bluetooth low energy . . . 112
5.3.1.3 WiFi . . . 113
5.3.1.4 Radio frequency identification . . . 113
5.3.1.5 Ultrasonic sensor . . . 114
5.3.2 On-board systems . . . 115
5.3.2.1 Inertial Navigation System . . . 115
5.3.2.2 Vision-based systems . . . 118
5.3.3 Device-free . . . 119
5.3.4 Accuracy performance summary of technologies . . . 119
5.4 Candidate systems for evaluation . . . 125
5.4.1 Ultra-wideband . . . 125
5.4.2 Inertial Navigation System . . . 126
5.5 Integration of candidate systems . . . 127
5.5.1 Integration types . . . 128
5.5.2 Methods of integration . . . 128
5.5.3 IMU/UWB integration . . . 131
5.6 Summary . . . 133

6 Data capture scheme design and execution 134

6.1 Experiment set-up . . . 134
6.1.1 Room configuration . . . 135
6.1.2 Systems configuration . . . 137
6.1.3 Coordinate frames . . . 139
6.1.4 Data collection scenarios/configurations . . . 141
6.2 Data collection process . . . 143
6.2.1 IMU . . . 143
6.2.2 Cameras . . . 144
6.2.3 UWB . . . 145

6.3 Time Synchronisation between the different sensors 145
6.3.1 IMU 146
6.3.2 Cameras 147
6.3.3 UWB 150
6.4 Summary 153

7 Ultra-Wideband measurement and position domain performance characterisation 154
7.1 Methodology 154
7.1.1 Least Squares based position estimation 154
7.1.2 Multilateration method 157
7.1.3 Hyperbolic method 159
7.1.4 Accuracy performance evaluation 161
7.2 Measurement errors 161
7.2.1 Error distribution 161
7.2.2 Measurement errors of dynamic mode 171
7.2.3 Range error dependency on speed of motion 179
7.2.4 Measurement corrections 185
7.3 Position estimation based on collected data 186
7.4 Summary 197

8 Inertial Measurement Unit measurements and position domain performance characterisation 199
8.1 Methodology 200
8.1.1 Madgwick filter 200
8.1.2 INS initialisation 204
8.2 Raw acceleration and angular velocity data 205
8.3 Acceleration pre-processing 210
8.4 INS position estimation 219

8.5 Comparison of the three approaches 224
8.6 Summary 229

9 Ultra-Wideband/Inertial Measurement Unit integration: Models and performance 231

9.1 Mathematical model of ESKF 232
9.1.1 Nominal state 233
9.1.2 Error state 234
9.1.3 State prediction step 235
9.1.4 Measurement update step 236
9.1.5 Error reset 239
9.2 IMU/UWB integration 239
9.2.1 Measurement corrections 242
9.2.2 UWB speed estimation 246
9.2.3 Measurement filtering 251
9.2.4 Covariance matrices/ variance covariance estimation 254
9.2.5 Implementation 259
9.3 IMU/UWB integration results 261
9.3.1 Performance of UWB speed estimation methods 262
9.3.2 Sensitivity of speed estimation method on the level of standard deviation used 265
9.3.3 Comparison of different positioning methods 269
9.4 Summary 277

10 Conclusion & Further work 279

10.1 Conclusions 279
10.2 Further Work 281

List of Figures

1 Pedestrian trajectory scenario. 25
2 Vehicle trajectory scenario. 26
3 The conceptual interdependencies between the different RNPs (Zhu et al. 2018, p. 3). 38
4 Systematic and Random errors. 39
5 Accuracy and precision. 40
6 Positioning accuracy requirements for different autonomous driving applications (Wymeersch et al. 2017, p. 82). 41
7 Positioning accuracy and integrity requirements for autonomous road vehicles and drones (GOV 2018, p. 63). 42
8 A cognitive system interacting with the environment (Skrypchuk et al. 2018, p. 4). 46
9 The components of a Functional Architecture for autonomous driving (Behere & Törngren 2016, p. 139). 47
10 The Stanley's software flowchart (Montemerlo et al. 2006, p. 983). 49
11 The Boss software architecture (Urmson et al. 2009, p. 20). 51
12 The software architecture of CMU autonomous vehicle (Wei et al. 2013, p. 769). 54
13 The structure of the autonomous driving algorithm in A1 (Jo et al. 2015, p. 5121). 55
14 The planning component of A1 and its similarities to Boss and CMU autonomous vehicle. 57
15 The information flow between the modules of the A1 automated vehicle (Taş et al. 2016, p. 307). 58

16 The functional system architecture of Ulm University's autonomous vehicle (Taş et al. 2016, p. 308). 59
17 Main components of autonomous driving (Ren & Xia 2023). 60
18 The modular structure of AV1.0 presented by Wayve (Wayve 2024). 60
19 The different functions of the vehicle functions. 62
20 The different functions of the situational awareness functions. 63
21 The different functions of the decision and planning and the system management functions. 64
22 Method used for the development of the FSA. 71
23 The vehicle component of the functional system architecture. 77
24 The situational awareness component of the functional system architecture. 78
25 The decision and planning component of the functional system architecture. 79
26 The proposed functional system architecture. 80
27 The functional system architecture for human driven vehicles at automation level 0. 85
28 The functional system architecture for visually impaired person navigating using a smart cane. 87
29 The proposed LBF within the proposed FSA. 92
30 The location based functions within the proposed functional system architecture. 93
31 Direct and Two step positioning approach (Sahinoglu et al. 2008, p. 63). . 96
32 Classification of positioning methods. 97
33 TWR range estimation. 99
34 Using lateration to calculate the position of UD. 100
35 TDOA positioning method. 101
36 AOA positioning method. 102
37 Angle calculation of angle of AOA and AOD (Woolley 2019, p. 10). 102
38 AOA angle calculation. 103
39 Principle of dead reckoning. 105

40 Signal propagation interference due to the surrounding environment. 106
41 The impact of satellite geometry on the positioning solution (El-Rabbany 2006, p. 58). 107
42 Radio spectrum and its used in navigation, broadcasting and telecommunications (Groves 2013, p. 278) 110
43 Multipath resolution for a narrowband and UWB signal (Mazhar et al. 2017, p. 4470). 111
44 Error accumulating in a DR system (Groves 2013, p. 4). 116
45 Room configuration. 136
46 Plan view of the UWB and Bluetooth configuration. 136
47 The distribution of the UWB anchors around the room. 137
48 The dynamic platform equipped with the UWB tag, IMU and the markers tracked by the camera system. 138
49 The conversion between the IMU and the local coordinate frame. 140
50 The initial direction of motion, determined as the tangent of the true motion recorded by the cameras in the x-y plane. 140
51 Configuration of the collected static points. 141
52 Types of dynamic data. 143
53 Identification of FIM using IMU raw accelerations. 147
54 Identification of first instance of motion using epoch-to-epoch velocity calculated by the camera system. 148
55 Difference between the IMU and camera's first instance of motion. 149
56 Identification of first instance of motion of a circular path using UWB. . . 152
57 Identification of first instance of motion of a straight path using UWB. . . 153
58 Measurement error distributions from static and dynamic data of anchor 0x115b. 162
59 Measurement error distributions from static and dynamic data of anchor 0x113c. 163

60 Measurement error distributions from static and dynamic data of anchor 0x110b. 163
61 Measurement error distributions from static and dynamic data of anchor 0x110d. 164
62 Measurement error distributions from static and dynamic data of anchor 0x117f. 164
63 Measurement error distributions from static and dynamic data of anchor 0x1102. 165
64 Measurement error distributions from static and dynamic data of anchor 0x1163. 165
65 Measurement error distributions from static and dynamic data of anchor 0x1171. 166
66 Measurement error distributions from static data of anchor 0x115b. 167
67 Measurement error distributions from static data of anchor 0x113c. 167
68 Measurement error distributions from static data of anchor 0x110b. 168
69 Measurement error distributions from static data of anchor 0x110d. 168
70 Measurement error distributions from static data of anchor 0x117f. 169
71 Measurement error distributions from static data of anchor 0x1102. 169
72 Measurement error distributions from static data of anchor 0x1163. 170
73 Measurement error distributions from static data of anchor 0x1171. 170
74 Colour-coded measurement errors of two different anchors for three paths. The three paths are a straight line (diag4), a random path (path10) and a circular path (path15). In every figure, the relevant anchor is highlighted in red and the arrows illustrate the direction of motion. 172
75 Measurement error distributions of dynamic data for anchor 0x115b. 175
76 Measurement error distributions of dynamic data for anchor 0x113c. 175
77 Measurement error distributions of dynamic data for anchor 0x110b. 176
78 Measurement error distributions of dynamic data for anchor 0x110d. 176
79 Measurement error distributions of dynamic data for anchor 0x117f. 177

80 Measurement error distributions of dynamic data for anchor 0x1102. 177
81 Measurement error distributions of dynamic data for anchor 0x1163. 178
82 Measurement error distributions of dynamic data for anchor 0x1171. 178
83 The straight line dynamic paths. 179
84 Measurement errors against distance from the anchor for anchor 0x1163. The red lines illustrate the large deviations of the trend of measurement errors with distance from the anchor 180
85 Measurement errors against distance from the anchor for anchor 0x1163, with the linear trend for every path. 182
86 Measurement errors against distance from the anchor for anchor 0x113c, with the linear trend for every path. 182
87 Measurement errors against distance from the anchor for anchor 0x1102, with the linear trend for every path. 183
88 Measurement errors against distance from the anchor for anchor 0x117f, with the linear trend for every path. 183
89 The gradients of the linear trends with respect to the speed of motion. . . 184
90 UWB positioning solution of straight line diag4. 186
91 UWB positioning solution of random path path10. 187
92 UWB positioning solution of circular path path15. 187
93 UWB 2D positioning solution errors for straight line diag4. 188
94 UWB 2D positioning solution errors of random path path10. 188
95 UWB 2D positioning solution errors of circular path path15. 189
96 UWB 2D positioning solution errors for position pos1. 192
97 UWB 2D positioning solution errors for position pos2. 193
98 UWB 2D positioning solution errors for position pos3. 193
99 UWB 2D positioning solution errors for position pos4. 194
100 UWB 2D positioning solution errors for position pos5. 194
101 UWB 2D positioning solution errors for position pos6. 195
102 UWB 2D positioning solution errors for position pos7. 195

103 UWB 2D positioning solution errors for position pos8. 196
104 IMU position estimation method. 200
105 Raw acceleration measurements of path diag1. 205
106 Raw angular velocity measurements of path diag1. 206
107 Fourier transform of accelerations collected during the static part of the trajectory. 211
108 Fourier transform of accelerations collected during the dynamic part of the trajectory. 212
109 Fourier transform of accelerations collected during the static part of the trajectory. 213
110 Filtered accelerations 218
111 Approach 1 used to calculate the INS-based solution. 220
112 Using the distance travelled and heading angle to estimate the INS-based position. 221
113 Approach 2 used to calculate the INS-based solution. 222
114 Approach 3 used to calculate the INS-based solution. 224
115 The three approaches used for the calculation of IMU-only solution of path15. 225
116 The IMU solution drift when using the three approached for the straight line diag1. 228
117 IMU/UWB integration implementations. 242
118 Displacement estimation using UWB measurements. 247
119 Displacement estimation using UWB measurements and the standard deviation of the measurement errors. 249
120 UWB measurement filtering. 253
121 UWB speed estimates using static data from anchor 0x1102. 256
122 The usage of the different UWB σ values in the integration. 258
123 Integration 1. 260
124 Integration 2. 261

125 Estimated speed errors using the two methods of UWB speed estimation for path15. Methods 1 and 2 correspond to UWB speed estimation without and with the consideration of measurements uncertainty, respectively. . . . 263
126 2D position errors when using UWB speed estimation methods 1 and 2 for path15. 265
127 2D Position errors with varying level of σ. 267
128 Positioning solution of straight line, random and circular path. 270
129 2D Position errors of random and circular paths. 272
130 2D Position errors of straight line paths. 275

List of Tables

1 The environments of operation of a moving agent. 27
2 Location Based Services in indoor environments 31
3 LBS requirement parameters 36
4 Summary of IMUs classification and performance (El-Sheimy & Youssef 2020) 117
5 Summary Table of accuracy performance of different technologies, with advantages and disadvantages obtained from Xia et al. (2022). 122
6 Data collection process of dynamic paths 142
7 IMU collected data (x-io Technologies 2022, p. 17). 144
8 The mean and standard deviation of measurement errors when the robot is moving towards and away from the anchor. 174
9 The correlation coefficients between the gradient of measurement errors and speed of motion. Calculated using Pearson correlation 184
10 Accuracy at 95% and RMSE of multilateration and hyperbolic methods for dynamic paths. 190
11 Accuracy at 95% and RMSE of multilateration and hyperbolic methods for static points. 192
12 The average measurements during the static phase of the trajectories. The bias of the measurements is the same as the average except for the acceleration along the z-axis which is compared to gravity. 208
13 The standard deviation σ of measurements during the static phase of the trajectories, representing the noise in the measurements. 209
14 The standard deviation σ of the raw and filtered acceleration measurements during the static phase of the trajectories, representing the noise in the measurements. 215

15 The standard deviation σ of the raw and filtered acceleration measurements during the dynamic phase of the trajectories, representing the noise in the measurements. 216
16 Percentiles of measurement errors during the first $24s$ of the trajectories, time required to reach the $2m$ position error and position drift rates. . . . 228
17 The variable used in the ESKF 233
18 Offset of UWB tag from the Camera tracking marker in the IMU frame . . 240
19 Variance of accelerations and angular velocities ω, calculated using the static phase of the trajectories 255
20 The variance of the UWB speed estimations. 257
21 The dynamic and static variances of UWB measurements used in the integration. 258
22 Number of speed estimates used during the integration depending on the speed estimation method used. Methods 1 and 2 correspond to UWB speed estimation without and with the consideration of measurements uncertainty, respectively. 263
23 Accuracy (95%) and RMSE of the 2D positioning solution using the methods 1 and 2 of UWB speed estimation. 264
24 Accuracy (95%) of the UWB speed estimation integration with varying level of σ 268
25 Accuracy (95%) of Integrations 1, 2 and UWB based positioning solutions. Table also shows % improvement in accuracy when Integration 2 is compared to Integration 1 and UWB based solutions. 273
26 RMSE of Integrations 1, 2 and UWB based positioning solutions. Table also shows % improvement in accuracy when Integration 2 is compared to Integration 1 and UWB based solutions. 274
27 Accuracy (95%) performance when gradually including UWB measurement filtering and correction 276

Chapter 1

Introduction

1.1. Background

Advances in position navigation and timing (PNT), and communication technologies over the years are gradually increasing the availability and demand for high performance location based services (LBS). LBS use the geospatial information of the user to provide the required service. An example of a LBS is a service that provides navigation and way-finding of an agent in an unknown environment. LBS have seen a major increase in demand in recent years. This is evident by the value of the LBS market, which was USD 36.2 billion in 2019, and is expected to rise to USD 126.4 billion by 2025 (MI 2020). A major contributor to the growth in the LBS market is the increasing number of smart-phone users, reported by United Nations, to be around 41.5% of the world's population in 2019 (MI 2020). Improvements in technology together with increasing number of smart-phone users, increase the availability of LBS, in turn increasing the demand for such services.

The market value of indoor LBS, was USD 17.99 billion in 2023 and is expected to grow to USD 45.16 billion by 2030 (GVR 2022). The increase in indoor LBS market value, illustrates the potential of indoor LBS, the significance of which is expected to increase with increasing connectivity and technological advancements, including indoor positioning systems (IPS) able to provide position and navigation (PN) information in indoor environments.

In outdoor environments, GNSS is the primary system for LBS. However, in indoor environments, there is still no consensus on a generic technology. A number of IPS systems exists, relying on technologies such as WiFi, Bluetooth and ultra-wideband (UWB). UWB technology operates in the 3.1GHz - 10.6GHz radio frequency spectrum (Dardari

et al. 2009) and its wide bandwidth (500MHz) results in a pulse signal which has some inherent advantages over narrowband signals. It allows for multipath resolution, and signal penetration through obstacles, resulting in signal propagation through a non line of sight (NLOS) paths (Elsanhoury et al. 2022, Perakis & Gikas 2018). UWB results in high accuracy range measurements and consequently positioning solutions at the mm level (Alarifi et al. 2016). However, complex environments challenge its performance, resulting in non-trivial multipath resolution (Elsanhoury et al. 2022, Perakis & Gikas 2018, Dardari et al. 2009), and NLOS propagation errors (Perakis & Gikas 2018, Fan & Du 2022). A number of studies tried to reduce measurement errors by calibrating the UWB measurements and detecting NLOS propagation (Perakis & Gikas 2018, Fan & Du 2022). In addition, other studies use UWB in integration with alternative sensors (Marković et al. 2022, Zhong et al. 2018, Hol et al. 2009, Yao et al. 2017)

Bluetooth is a narrowband signal operating at 2.4GHz. Its received signal strength (RSS) or angle measurements are used to calculate positioning solutions. RSS measurements are either used in the method of lateration or fingerprinting and their accuracy performance is between $1m$-$3m$ (Mendoza-Silva et al. 2019). Alternatively, Bluetooth angle measurements in the form of either angle of arrival (AOA) or angle of departure (AOD) achieve the highest positioning accuracy in Bluetooth-based methods, at the sub-meter level (Zhuang et al. 2022). Their performance is however compromised in challenging environments, while it is prone to interference (Alarifi et al. 2016). Another popular system is WiFi technology, where RSS measurements are commonly used either in lateration or fingerprinting methods (Yassin et al. 2016). RSS-based lateration relies on signal propagation models, introducing inaccuracies in the RSS-based range measurement when the model is not representative of the environment (Khalajmehrabadi et al. 2017). Alternatively, RSS fingerprinting is more commonly used and results in 5m-10m of accuracy (Basiri et al. 2017).

Popular systems like WiFi and Bluetooth have weak anti-interference ability and their accuracy performance is at a meter level (Xia et al. 2022), vision based systems cannot operate in environments with too many obstructions and sound systems are characterised

by a short ranges (Xia et al. 2022, Yassin et al. 2016, Mendoza-Silva et al. 2019). UWB's increasing popularity stems from high accuracy performance (Alarifi et al. 2016), when signal propagation errors are reduced. Thus the complexity of indoor environments presents a challenge to the performance of existing technologies, giving rise to the need for an integrated system.

Integrating systems complimentary to one another lead to improved performance compared to standalone systems. However, when considering system integration, there exists a large number of possible combinations of systems and technologies to integrate. This gives rise to the additional challenge of establishing a generic IPS. The approaches presented in the literature for the development of IPS, consider the technologies, without an exhaustive understanding of their use and need at the operational level. This lack of a systematic approach contributed to the development of a multitiude of IPS.

The limitations identified in the literature relate to the approach taken to solve the indoor positioning problem and the lack of consensus on a generic IPS. The former refers to current approaches taken, which involve the identification of applications and their requirements which leads directly to the development of the physical architecture describing the technology and systems used to achieve the requirements. However, the technologies are considered without an exhaustive understanding of the operational level and their need within that level. This lack of an exhaustive understanding of the operational level, leads to the latter limitation, where various IPS are developed making it challenging to identify a generic IPS.

1.2. Aims and objectives

Given the limitations of the current IPS, this thesis aims to develop a systematic approach in addressing the indoor positioning problem, with a focus on the system's accuracy. The objectives formulated to achieve this aim are:

1. Identify location based services in indoor environments and specify the accuracy requirements of critical applications.

2. Develop a functional system architecture (FSA) for the operation of a cognitive agent and identify location based functions (LBF) within the FSA.

3. Identify state-of-the-art technologies used in IPS systems and determine their strengths, weaknesses and accuracy performance in order to select candidate technologies that can support the LBFs in objective 2.

4. Collection of empirical data using the candidate systems for a detail characterisation of their performance.

5. Detail characterisation of the selected UWB and INS systems.

6. Development of the physical architecture that allows the integration of the two systems and improves the accuracy performance and availability of the system.

1.3. Thesis structure

This thesis consists of ten chapters. The first chapter presents the background of the research to formulate the aims and objectives. Chapter 2 introduces the different environments of operation of an agent and identifies the indoor location based services (LBS) identified through a review of the literature. In addition, the required levels of accuracy for an IPS are identified using the requirements of safety critical services. Chapter 3, introduces the concept of a cognitive agent and presents a review of the literature for the development of the FSA. The literature review focuses on the use case of an autonomous vehicle (AV), as it is an example of a cognitive agent operating in complex environments. The findings of chapter 3, are used in chapter 4, where the FSA of a cognitive agent, using the use case of an AV is developed, and its tranferability to other agents is illustrated. Following the FSA development, the LBF are identified in chapter 4.

Once the FSA is developed and LBF are identified, chapter 5, presents a review of the literature of the state-of-the-art technologies used in IPS. It discusses their strengths and weaknesses and through a comparison of their accuracy performance, it selects the candidate systems of UWB and INS for further investigation. Chapter 6, presents the

data capture scheme used for the collection of real UWB and IMU data for analysis. Chapter 7, provides a detailed characterisation of UWB by analysing the UWB data and calculating UWB-standalone positions. In chapter 8, a detailed characterisation of the INS positioning solutions is presented. Chapter 9 presents the proposed integration of the two systems (UWB and INS), and evaluates its performance. Finally, chapter 10 presents the main conclusions of this research and suggestions for further work.

Chapter 2

Location based services and their required navigation performance

2.1. Operating environments of a moving agent

This section uses two different type of agents, a human and vehicle to understand the operational environments, including their similarities and differences. An ordinary trajectory, followed by a pedestrian, is initially considered when identifying different types of environments. Then an ordinary trajectory of a vehicle is considered. These two trajectories are used to classify the different environments, depending on the surrounding conditions.

In order to identify the environments in which a pedestrian operates, a pedestrian trajectory scenario is considered. Pedestrians generally operate in both outdoor and indoor environments, hence the said trajectory traverses from an outdoors into an indoors environment and vice versa. More precisely, a scenario is considered where a pedestrian navigates around the campus of Imperial College London and accesses one of the buildings (for example Skempton Building) as shown in Figure 1. Within this scenario three different environments are identified, namely outdoors, indoors and transitions, which are considered to be the areas of intersection between outdoors and indoors.

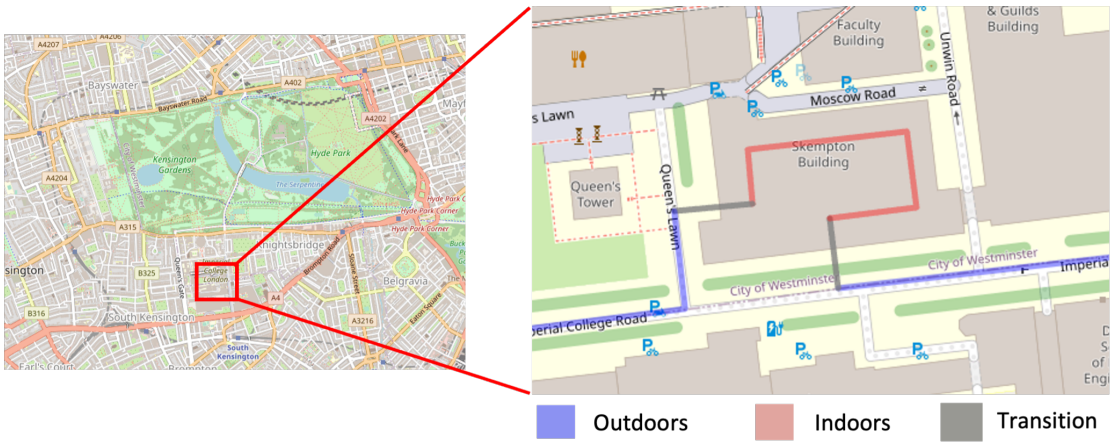


Figure 1: Pedestrian trajectory scenario.

As opposed to a pedestrian, a vehicle mainly operates in outdoor environments, although, some outdoor environments have very similar characteristic to the transition and indoor environment identified in the pedestrian trajectory. The trajectory considered for the vehicle scenario is illustrated in Figure 2. The main categories observed here are open-sky conditions, complex environments and transition between open-sky and complex areas. The classification of these environments is based on Global Navigation Satellite Systems (GNSS) signal propagation environments, as GNSS is a widely used system for providing position information in an outdoor setting.

Open-sky environments are characterised by the vehicle having an unobstructed view of the sky, usually in the case of rural areas. Alternatively, complex environments are usually urban areas which are characterised by tall buildings surrounding the roads (a phenomenon known as urban canyon) restricting the sky view. More examples of complex areas include tunnels and indoor parking areas where the sky view is fully blocked (referred to as non-sky environments), as in the case of indoor environments. Lastly, a transition zone is where a vehicle moves from an open-sky to a complex environment and vice versa.

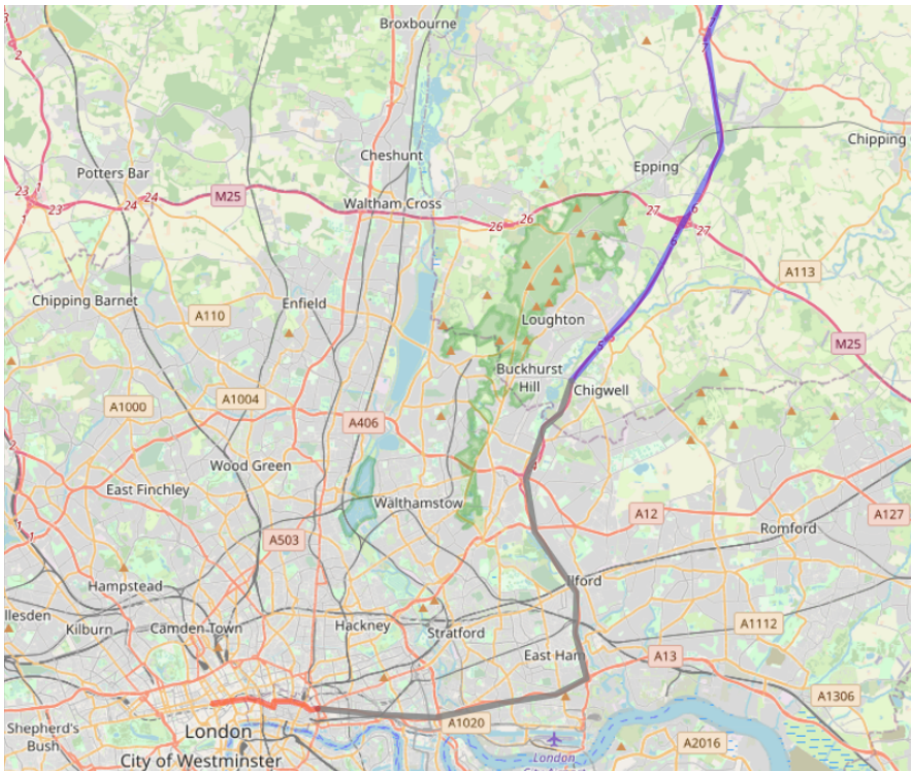


Figure 2: Vehicle trajectory scenario.

In summary, there are three types of environments presented in both the vehicle and pedestrian trajectories. The vehicle can operate in complex, open sky and transition environments, while the pedestrian operates in indoor, outdoor and transition environments. The environments identified in the vehicle and pedestrian trajectories, have many similarities. Table 1 presents the way the environment definitions of the vehicle and pedestrian trajectory correlate and states the way these environments are referred to in this thesis. Complex environments refer to no-sky/ indoor and urban/ outdoor environments, clear environments refer to open-sky/outdoor environments and partially-clear environments refer to transition areas between complex and clear environments.

Table 1: The environments of operation of a moving agent.

Term used in the thesis	**Vehicle**		**Pedestrian**
Complex environment	Complex	No-sky	Indoor
		Urban	Outdoor
Clear environment	Open-sky		Outdoor
Partially-clear environment	Transition		Transition

An interesting thing to note is that the complex environment identified in the case of a vehicle, can be both an indoor and outdoor environment in the case of pedestrians. A complex environment with no-sky conditions is equivalent to an indoor environment, while a complex urban environment correspond to an outdoor environment. This distinction is particularly useful when considering the technologies to use for indoor positioning. When considering the operation of any agent in an outdoor environment, a common positioning system to use is GNSS. GNSS can calculate with high accuracy the position of an agent, in open-sky conditions without obstructions to the signal. Complex environments on the other hand and due to special attenuation, are challenging areas for GNSS positioning, without any augmentations. In no-sky environments, such as tunnels and indoor parking spaces the GNSS signal is completely blocked and cannot be used. In those situations alternative or integration systems are required. Understanding the environment within which an agent operates is therefore crucial in developing a system able to provide required positioning performance.

This thesis addresses complex environments and more specifically indoor environment. The development of a positioning solution for an indoor environment can be applied to any complex environment (i.e. urban canyons) as they all belong in the same category and have very similar characteristics.

2.2. Location Based Services

Location Base Services (LBS) use geospatial information of the user to provide the required service. Advances in position navigation and timing (PNT) and communication

technologies over the years are gradually increasing the availability and demand for LBS. The identification of LBS is an essential step in understanding the positioning requirements, hence the remaining of this section describes the indoor LBS identified in the literature.

The different indoor LBS are illustrated in table 2. According to a survey conducted by Basiri et al. (2017), there are five application categories of indoor LBS: Indoor navigation and tracking, Marketing, Entertainment, Location-based information retrieval and Emergency and security applications. A comparison between Basiri et al. (2017) and Naser et al. (2023), Qi et al. (2024), Che et al. (2023), Werner (2014), Yassin et al. (2016), Mendoza-Silva et al. (2019), and Mautz (2012), reveals that the five categories satisfy most of the LBS with the exception of Robotic applications and Logistics and operations. Theses have been included as additional categories in table 2.

Indoor navigation and tracking is a category that mainly includes pedestrian navigation and asset tracking and finding. Pedestrian indoor navigation is particularly useful when operating in spaces such as malls, hospitals and airports, where the user may not be familiar with the environment. Moreover, pedestrian indoor/outdoor navigation is useful for navigation and way-finding by vulnerable people such as visually impaired people (VIP).

In the category of marketing, location data of users can be used in applications such as location-based marketing, reward sharing and dealing, proximity-based offers and advertisements. These are applications where the user receives offers and advertisements, based on their proximity to a specific location. Using location-based marketing allows for a more targeted way of approaching customers, while also enabling customers to be aware of available options and offers in their proximity.

The category of Entertainment/ Leisure includes applications like location-based social networking, gaming, friend finding and information of events in the vicinity. These applications are based on proximity measurements, where the location of the user is used to connect the user with others in the vicinity, or provide information of leisure and entertainment events in the vicinity.

Location-based information retrieval category includes applications like location-based query, proximity searching, tourist guidance, transport information and place of interest

notifications. These applications are used when the user would like to know more about the facilities and operations in their vicinity. For example tourist guidance and transportation information uses the location of the user to help navigate to a required destination or a point of interest. Alternatively, proximity searching can be used by the user to identify facilities and services in the area.

Emergency, safety and security applications include the use of user location in safety critical scenarios like emergency rescue, ambient assisted living and safe guiding of visually impaired people (VIPs). The applications in this categories can vary from search and rescue operations of emergency services to more supportive operations such as the navigation of VIP. All of the applications in this category are safety critical as their user relies on them for their safety and well-being. An additional application, is the use of proximity information to monitor the spread of disease such as in the case of a pandemic. The importance of this application was seen recently during the spread of the COVID19 pandemic, where proximity information was used to monitor and manage infections.

The category of robotic applications refers to services where robots are used. Robots can be used for performing precise operations (even surgeries), used in industry and especially in smart factories or can even be used domestically such as a robot vacuum. All of the aforementioned robotic applications require the location of the robot to be known, for its operation.

Finally, logistics and operations is a category that mainly includes applications from an industrial setting. Applications such as cargo management, process observation and asset tracking in a warehouse, require location information of the assets. Alternatively applications such as process observation and optimization require location information of staff in addition to assets.

As can be seen from table 2, applications vary from non-safety critical (such as Entertainment/ Leisure) to safety critical (such as Emergency, safety and security). The criticality as well as the functionality of each application are the key factors in determining both their required performance parameters and their associated required targets. As can be seen from table 2, Basiri et al. (2017) provides the Quality of Service (QoS) require-

ments for the identified applications, which vary depending on the application category. QoS represents the requirements of the LBS categories and hence the required performance of the IPS. When considering the development of any system, it is essential to take the performance requirements into consideration. The following section considers the LBS requirements for indoor positioning.

Table 2: Location Based Services in indoor environments

	Application	Examples	Quality of service requirement	Ref
Indoor navigation and tracking	Indoor navigation	• Pedestrian navigation • Path finding and routing • Tracking • Asset finding	– Response in near-real-time – Accuracy within a few meters – Seamless availability (indoors and outdoors) – Good reliability and continuity of service – Low–medium power consumption – Reasonable or cheap price – Strong privacy preservation	Che et al. (2023), Qi et al. (2024), Basiri et al. (2017), Mautz (2012), Werner (2014), Mendoza-Silva et al. (2019), Yassin et al. (2016)
	Navigation Service	• Personalised interactive maps-the user can select their route preferences		Werner (2014)
	Pedestrian Navigation	• Aid visually impaired people		Mautz (2012)
Marketing	Marketing	• Location-based social marketing • Location-based reward sharing • Location-based dealing • Proximity-based offers • Advertisement	– Medium to low availability – Response in few minutes – Accuracy in the order of hundreds of meters – Medium reliability and continuity – Very low power consumption – Free or very inexpensive – Medium to strong privacy preservation	Qi et al. (2024), Naser et al. (2023), Basiri et al. (2017), Werner (2014), Yassin et al. (2016)
	Retail and Commerce	• Proximity marketing		Werner (2014)

Entertainment/ Leisure	Entertainment	• Location-based Social Networking • Location-based Gaming • Find your friend	– Medium to high availability (seamless indoors and outdoors) – Response in real-time or a few seconds – Accuracy in the order of tens of meters – High reliability and continuity – Low power consumption – Reasonable or cheap price – Medium privacy preservation	Basiri et al. (2017), Mautz (2012) Werner (2014)
	Social networking and joint activities			Werner (2014)
		• Up-to-date data on cinemas • Events in the vicinity		Yassin et al. (2016)
Location-based information retrieval	Location-based information retrieval	• Location-Based Q&A (Query) • Proximity Searching • Tourist Guidance • Transportation information	– Medium availability – Response in real-time or a few seconds – Accuracy from a few meters to hundreds of meters – High reliability and continuity – Low power consumption – Reasonable or cheap price – Medium privacy preservation	Basiri et al. (2017), Werner (2014)
	Information Services	• Map application- smartphone provides user with a map of his surrounding • Place of interest notification		Werner (2014)

Emergency, safety and security applications	Safety and Security	• Emergency services • Emergency alert services • Ambient assisted living • Security surveillance	– Very high availability (seamless indoors and outdoors) – Response in real-time or few seconds – Accuracy of tens of meters or lower –Very high reliability and continuity – Low power consumption – Reasonable or cheap price – Medium or low privacy preservation	Basiri et al. (2017)
	Safety-of-life applications	• Emergency call • Ambient assisted living		Werner (2014), Yassin et al. (2016), Mautz (2012)
	Medical care	• Tracking of medical personnel in hospitals • Patient and equipment tracking		Che et al. (2023), Mautz (2012)
	Emergency services	• Used in search & rescue services		Qi et al. (2024), Naser et al. (2023)
	Guiding of vulnerable people	• Aid visually impaired people		Mautz (2012)
	Monitoring spread of the pandemic	• Proximity		

Robotic applications	Robot navigation		Yassin et al. (2016)
	Industry	• Industrial robots • Smart factories • Robotic guidance	Che et al. (2023), Qi et al. (2024), Mautz (2012)
		Robots performing precise operations	Mendoza-Silva et al. (2019) Mautz (2012)
	Domestic robots	• Robot vacuum	Che et al. (2023), Qi et al. (2024)
Logistics and operations	Industry	• Indoor positioning awareness • Find tagged tools	Mautz (2012)
	Management	• Process Observation	Werner (2014)
	Logistics and Optimization	• Location of assets and staff members for process optimization in complex systems • Positioning for cargo management systems at airports and ports to increase efficiency	Che et al. (2023), Naser et al. (2023), Mautz (2012)
	Information Services	• Smart logistics- external online service to gather traffic info	Werner (2014)
	Warehouse operations	• Asset management and tracking	Che et al. (2023), Naser et al. (2023)

2.3. Required Navigation Performance parameters

For the development of an indoor positioning system able to support indoor LBS, it is essential to understand the LBS requirements related to the provision of PNT information. These requirements are also used to evaluate the performance of PNT systems and are represented by the required navigation performance (RNP) parameters of accuracy, integrity, continuity and availability (Ochieng et al. 1999). The following sub-section presents the LBS requirements identified in the literature.

2.3.1. Service requirements

For the development of a system able to support indoor LBS, it is essential to understand and take into consideration the LBS requirements. Table 2 illustrates some of the requirements identified by the survey carried out by Basiri et al. (2017). A collection of the LBS requirements for an indoor positioning system from the literature, is shown in table 3.

For the provision of LBS, there are many stakeholders in place, the requirements of which do not necessarily align. When presenting their survey data, Basiri et al. (2017) split the participants into four groups: 1) the LBS users, 2) LBS application developers, 3) LBS content providers that produce the content to be delivered through the LBS application (such as maps, advertisements etc) and 4) the LBS research and market analyst. The results of the survey illustrate that different participant groups prioritise their LBS requirements differently. As an example LBS users, perceive privacy as one of the most important features, except in the case of Emergency, safety and security applications. Alternatively, LBS developers perceive cost and user interface to be more important than privacy (Basiri et al. 2017). The fact that different participant groups prioritise their requirements differently, is a challenge in identifying the most important user requirement. The prioritisation of the requirements seem to be both user and application dependent.

The user requirements listed in table 3, include the four RNP parameters, which are commonly used to measure the performance of a navigation system (Ochieng et al. 1999). These parameters reflect the PNT requirements of the LBS and are essential to know when

developing a PNT (in this case IPS) system. The four RNPs, are discussed in detail in the next section.

Table 3: LBS requirement parameters

Requirement parameters	**Ref**
Real time positioning	Basiri et al. (2017)
Accuracy	Basiri et al. (2017), Mendoza-Silva et al. (2019), Wirola et al. (2010), Mautz (2012)
Availability	Basiri et al. (2017), Wirola et al. (2010), Mautz (2012)
Reliability	Basiri et al. (2017)
Service continuity	Basiri et al. (2017), Wirola et al. (2010), Mautz (2012)
Power consumption	Basiri et al. (2017), Mendoza-Silva et al. (2019), Mautz (2012)
Price	Basiri et al. (2017), Mautz (2012)
Privacy	Basiri et al. (2017), Mendoza-Silva et al. (2019), Wirola et al. (2010), Mautz (2012)
Cost of the infrastructure	Basiri et al. (2017), Mautz (2012)
Coverage	Mendoza-Silva et al. (2019), Mautz (2012)
Complexity	Mendoza-Silva et al. (2019)
Robustness	Mendoza-Silva et al. (2019), Mautz (2012)
Scalability	Mendoza-Silva et al. (2019), Mautz (2012)
Speed	Wirola et al. (2010)
Size	Mautz (2012)
Market maturity	Mautz (2012)
Integrity	Mautz (2012)
Update rate	Mautz (2012)
Interface	Mautz (2012)
Number of users	Mautz (2012)

Table 3 continued from previous page

Maintenance	Mautz (2012)
Security and Safety	Mautz (2012)
Weight of the device	

2.3.2. Required Navigation Performance parameters

Required Navigation performance parameters, are those used to evaluate the performance of a PNT system. The four RNP parameters of accuracy, integrity, continuity and availability originate from the field of aviation, and are used to measure a navigation system's performance (Ochieng et al. 1999). The target level of the RNP parameters depend upon the application for which the positioning system is used.

Accuracy corresponds to the degree of conformance of the estimated position with a reference value, which is ideally the truth (Ochieng et al. 2003). The accuracy parameter is commonly specified at the 95^{th} percentile and corresponds to the maximum position error obtained 95% of the times. Integrity represents the level of confidence in the derived PNT information and reflects its ability to return timely and valid warnings when the system should not be used (Ochieng et al. 2003). Continuity is defined as the ability of the system to function, without interruption during the intended period of operation (Ochieng et al. 2003). The continuity of the system is the ability of the system to provide PNT information of required accuracy and integrity. Availability, represents the percentage of times the system is available for use. When the system is available the accuracy, integrity and continuity requirements need to be satisfied (Ochieng et al. 2003)

According to Zhu et al. (2018), the four RNPs are interdependent. The conceptual connectivity between them is illustrated in Figure 3. The accuracy is found at the base of the performance pyramid (Zhu et al. 2018). Integrity is connected to accuracy, since it is evaluated based on the accuracy requirements of the positioning solution. Moreover, a system is thought to be continuous only if an accurate solution of high integrity can be maintained for the whole duration of operation. Finally, the availability parameter, is based on the reliability of the obtained information. In order for the system to be reliable,

it needs to comply with the accuracy, integrity and continuity parameters (Ochieng et al. 2003, Zhu et al. 2018). As this thesis focuses on the accuracy of IPS, a more detailed explanation of the accuracy at the base of the pyramid follows.

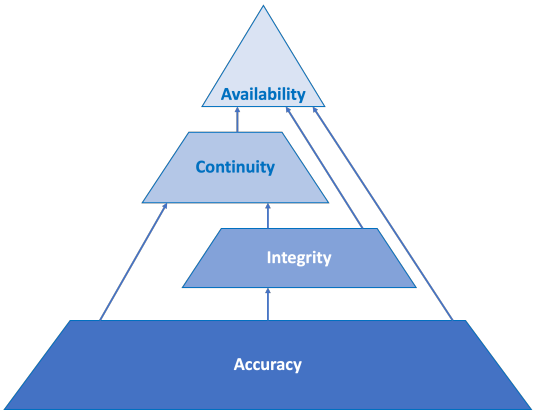


Figure 3: The conceptual interdependencies between the different RNPs (Zhu et al. 2018, p. 3).

Accuracy is a term used to express the deviation of a measurement or calculated solution from the true value. The accuracy of the positioning solution is influenced by the errors in the measurements, which can be random or systematic. The errors in the measurements are defined as the difference between the measurement and the true value as illustrated in (1). Systematic errors refer to biases that shift the measurements in a specific direction away from the true value. Alternatively, random errors are usually small and unpredictable in magnitude or sign. A graphical illustration of the impact of random and systematic errors is illustrated in Figure 4.

$$Error = Measurement - TrueValue \tag{1a}$$

$$Correction = -Error \tag{1b}$$

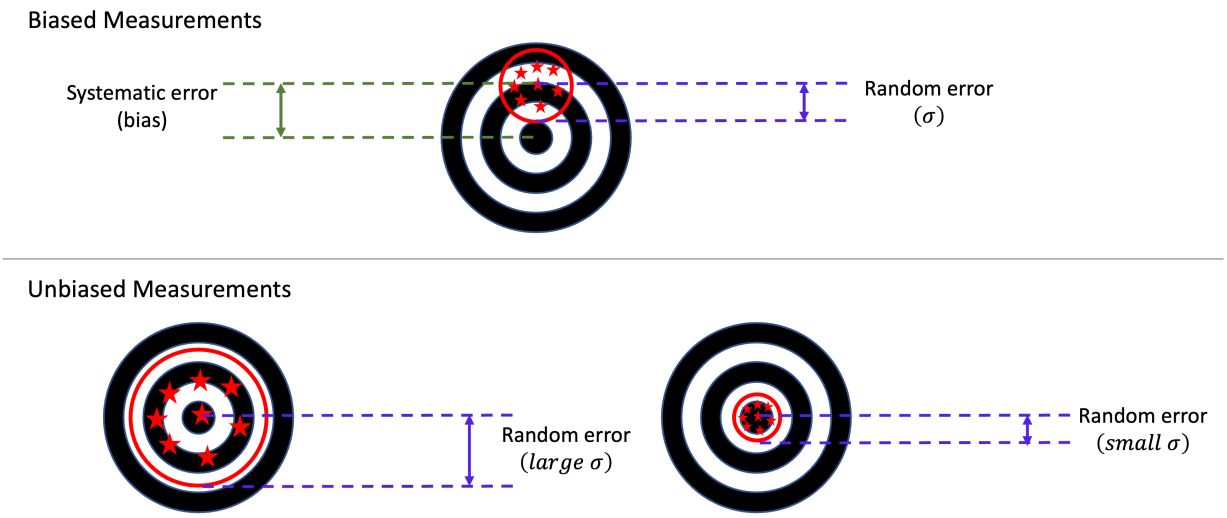


Figure 4: Systematic and Random errors.

Random errors are a representation of the measurements' precision. The lower the random errors, the higher the precision of the measurement. Precision is sometimes associated with accuracy, however the two terms are not synonyms. Measurements can be precise but inaccurate as shown in figure 5.

In the field of GNSS, accuracy of the system is expressed at the 95^{th} percentile and corresponds to positioning errors obtained 95% of the times (Ochieng et al. 2003). This metric is also applied in this thesis and as will be discussed later, accuracy performance is evaluated at the 95^{th} percentile. In the reviewed literature, references to accuracy requirements and performance are made, however, the accuracy metric is not specified. As a result, in this thesis, the accuracy performance and requirements from the literature, are assumed to refer to accuracy (95%), unless otherwise specified.

According to GOV (2018), great care should be taken when deciding on the importance of the different RNP parameters. Usually a lot of emphasis is placed on accuracy and even though it is important, it is not the only parameter required for a safety critical system to operate adequately. A combination of the four RNPs with their corresponding targets need to be defined for every application, to ensure the application can operate as expected. The next section sets the RNP targets for this thesis focusing on the parameter of accuracy.

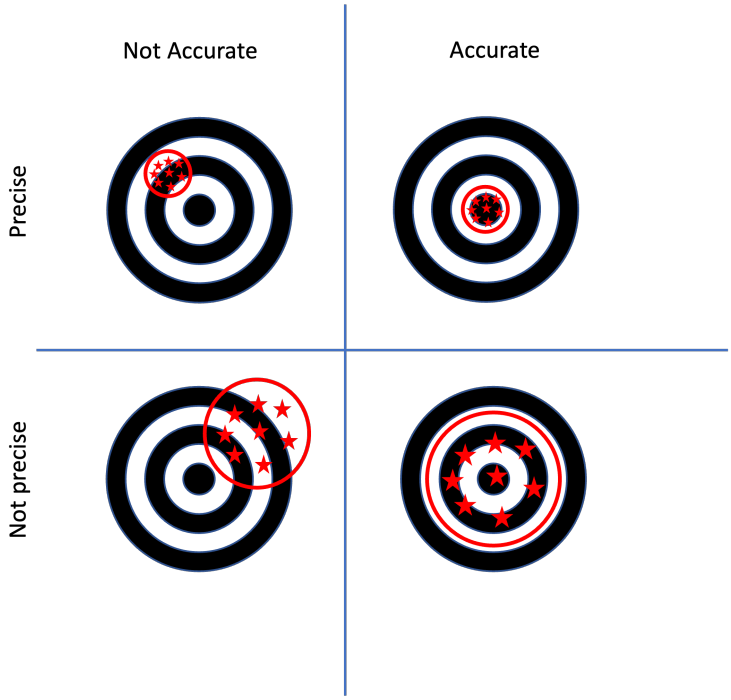


Figure 5: Accuracy and precision.

2.3.3. Required Navigation performance targets

It is important to know the required RNP targets when developing a PNT system. Defining clear targets for RNP parameters, allows the development of a system able to provide safe positioning and navigation techniques. The adopted PNT system should be able to support the most stringent positioning requirements since this ensures a system that can support any application. The LBS with the most stringent RNP requirements are safety critical applications. Safety critical applications tend to have the stringent RNP requirements to ensure the safe operation of the system. A safety critical application identified from table 2 is the guidance of vulnerable people, such as VIPs, in indoor environments.

Determining RNP targets can be challenging as, in addition to being an application dependent decision, it also depends on the surrounding conditions. As discussed by Mendoza-Silva et al. (2019), a comfortable positioning accuracy threshold for pedestrians, is related to the free space around them. In open sky conditions and in an outdoor setting, 5m

accuracy is sufficient for pedestrian navigation. However, in an indoor environment with narrow corridors, 5m accuracy is insufficient (Mendoza-Silva et al. 2019). According to Mendoza-Silva et al. (2019), an accuracy level of 1m-2m is adequate for most indoor LBS, Wise et al. (2012) suggest that a strict accuracy of 2m can support the majority of VIP users, while Che et al. (2023) states that indoor positioning systems need to deliver a solution of high precision and centimeter-level accuracy.

Another safety critical application is the operation of an autonomous vehicle (AV). In addition to being a safety critical operation, AV is an example of an agent operating in complex environments with intensive interaction with other road users. As a result, PNT requirements of an AV provide an indication of the RNP requirements of moving agents navigating through complex environments. Figure 6 illustrates the accuracy requirements of different AV use cases and the ability of different technologies to achieve these requirements (Wymeersch et al. 2017). Alternatively, the required levels of position integrity and accuracy, proposed by GOV (2018), are illustrated in Figure 7. In both cases the most stringent positioning accuracy requirement is at a centimetre-level.

Use case	Accuracy requirement	GPS	4G	5G	lidar, camera, radar + map
Automated overtake	30 cm	×	×	✓	✓
Cooperative collision avoidance	30 cm	×	×	✓	✓
High density platooning	30 cm	×	×	✓	✓
See through vehicle	> 1 m	✓	✓	✓	✓
Bird's eye view	> 1 m	✓	✓	✓	✓
Vulnerable road user discovery	10 cm	×	×	✓	✓
	Cost	Low	Low	Low	High
	Latency	High	Medium	Low	Medium
	Reliability	High in open spaces	Medium/low	High (with D2D, cooperative)	Medium/high (weather-dependent)

Figure 6: Positioning accuracy requirements for different autonomous driving applications (Wymeersch et al. 2017, p. 82).

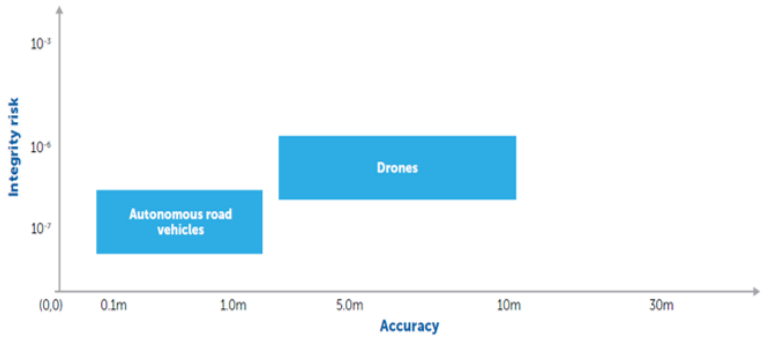


Figure 7: Positioning accuracy and integrity requirements for autonomous road vehicles and drones (GOV 2018, p. 63).

Both cases of a VIP operating in indoor environments and an AV in complex environments, refer to safety critical applications of agents navigating in complex environments. Safety critical applications, commonly have the most stringent RNP requirements. Although defining RNP targets for the development of a PNT system is application dependent, defining the targets based on safety critical LBS ensures the development of a PNT system able to support other, non-safety critical LBS. Using the examples of a VIP and an AV operating in complex environments, the accuracy target for the purpose of this thesis is set at a decimeter level (at 30 cm), with the error in position not exceeding 2m.

2.4. Summary

This chapter has presented the three operational environments of clear, partially clear and complex environments using the two scenarios of a vehicle and a pedestrian trajectory. With the focus of the thesis being the navigation of a cognitive agent in a complex environment, and in particular an indoor environment, indoor LBSs were identified from the literature and classified into seven categories. Subsequently, LBS requirements were identified, while the RNPs were introduced, with accuracy being the system requirement this thesis focuses on. The target level of accuracy (95%) of an agent moving in a complex environment is set to be decimetric (30cm), while it should not exceed 2m. The target accuracy was set using the use case of a VIP and an AV, as both are moving agents operating in complex environments.

Having identified the LBS and their requirements, it is essential to understand the way a moving agent operates before developing a PNT system to support the operation of such an agent. Understanding the way an agent operates, involves the development of a functional system architecture (FSA). The FSA illustrates the different functions and their inter-connectivity required for the operation of a cognitive agent. The next chapter focuses on a review of the literature for the development of an FSA for the operation of a moving agent. The development of the FSA focuses on the case of an AV, as it is a moving agent operating in a complex environment, while it is also being adapted for the operation of a human-driven vehicle and the navigation of a VIP. These three cases illustrate the FSA's ability to represent the operation of cognitive agents as well as the operation of multi-agent systems, like the cooperation of a human driver and a vehicle and the navigation of a VIP using a smart cane.

Chapter 3

State-of-the-art functional system architectures of dynamic autonomous agents

In chapter 2, the indoor LBS and their requirements were identified after a review of the literature. The RNP parameters, evaluating the performance of the PNT system and target level of positioning accuracy, were also defined for the operation of a cognitive agent in complex environments. In order to address the topic of the thesis it is essential to understand the wider requirements and functional system architecture (FSA) of dynamic autonomous agents. This functional architecture will then allow for the identification of the elements that are location based and their subsequent positioning requirements.

A FSA illustrates the structure of the system with the required functions and their inter-connectivity, to achieve the desirable operation. Inter-connectivity shows the way information flows from one function to another and aids in understanding the workflow and complexity of the system. This helps in understanding the criticality of the different functions before proceeding to its development and implementation.

Dynamic autonomous agents operate in different environments and can be classified based on their operational environment. Xu et al. (2023) classifies mobile robots into ground, marine and air mobile robots. The classification of the different ground environments was presented in section 2.1 and autonomous agents can be split into agents that operate in complex, clear and partially-clear environments. As can be seen in table 1, both indoor and outdoor environments can be classified as complex, depending on the complexity of the surrounding. As a result the functional architecture of an agent operating in outdoor complex environments is the same as the architecture of an agent operating a

an indoor environment. This is also evident by Niloy et al. (2021), Ponnan et al. (2022), where the structure of the functional architecture of mobile robots is the same irrespective of the environment of operation (outdoor or indoor). As a result, this chapter focuses on the operation of an autonomous vehicle (AV), as a dynamic agent that operates in complex environments.

This chapter provides a review of the literature for the development of a FSA for the operation of a cognitive agent and more specifically the operation of an autonomous vehicle (AV). The chapter begins by introducing the concept of a cognitive agent and its main components. It then proceeds to the review of some existing AV FSAs and the identifications of their key components. A comparison between the reviewed FSAs then follows, which leads to a discussion of the different layers of decision and planning. The chapter concludes with a summary of key findings, used in chapter 4 as the basis for the development of the proposed FSA, for the operation of a cognitive agent.

3.1. Cognitive agent

A cognitive system operates as a rational agent, who is able to observe the surrounding environment, make decisions and take actions within the perceived environment (Taş et al. 2016). The concept of a cognitive agent can be applied in a variety of different fields, spanning from robotics to human behaviour. Different fields use different terminologies, but the basic principle is the same. A cognitive agent should be able to perceive the environment around it (or be situationally aware), make decisions and take actions within the perceived environment. A simplified representation of this process is illustrated in Figure 8.

The term 'situational awareness' (SA) is generally used to describe the cognitive behaviour of humans (Endsley 1995, Skrypchuk et al. 2018), while in the field of AVs and robotics, the term perception is more frequently used (Behere & Törngren 2016, Taş et al. 2016, Russell & Norvig 2010, Poole & Mackworth 2010, Berger & Rumpe 2014). Similarly, the decision component is sometimes referred to as path planning (Lin et al. 2018, Endsley 1995) and the action component as vehicle control in the case of AVs. The three compo-

nents of the cognitive agent will be referred to in this thesis as SA, decision and action. These will correspond to the three components illustrated in Figure 8.

Autonomous robots are cognitive agents operating within their surrounding and their workflow follows a see-thing-act structure (Arents & Greitans 2022, Niloy et al. 2021, Ponnan et al. 2022, Xu et al. 2023), which corresponds to the SA, decision and action components of the cognitive agents. As an AV is an example of an autonomous agent, its structure also aligns with the three components of a cognitive agent.

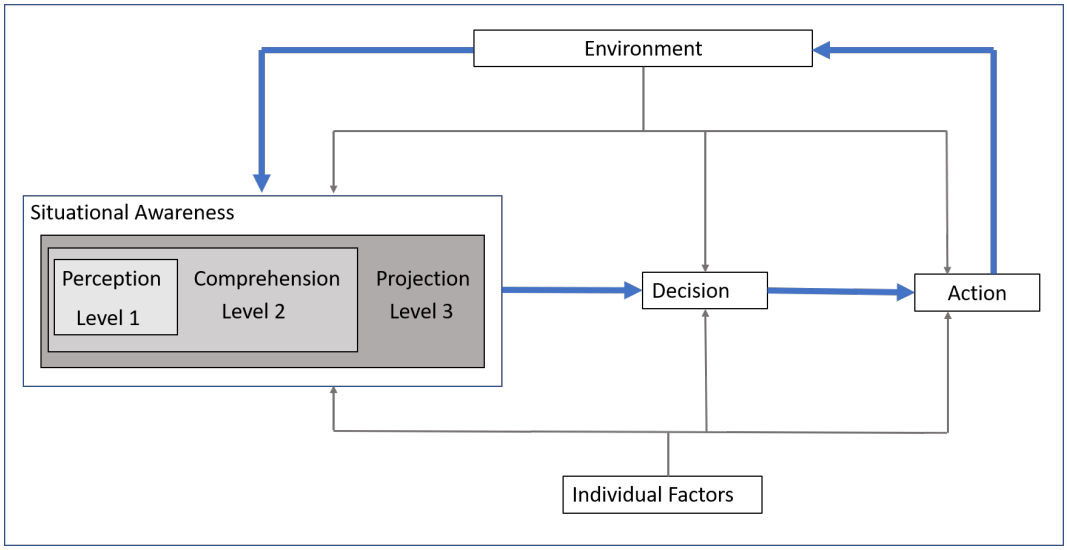


Figure 8: A cognitive system interacting with the environment (Skrypchuk et al. 2018, p. 4).

A high-level architecture of an AV, equivalent to Figure 8, is presented by Behere & Törngren (2016) (shown in Figure 9), in which perception component deals with the interpretation and understanding of the surrounding environment as well as the localisation of the AV. The decision and control component decides on vehicle motion within the perceived environment, while the vehicle platform manipulation deals mostly with actuation of a vehicle to achieve closed-loop control of the desired motion (Behere & Törngren 2016). According to the definitions provided, the perception, decision & control and vehicle platform manipulation components in Figure 9, corresponds to the SA, decision and action components of a cognitive agent in Figure 8, respectively.

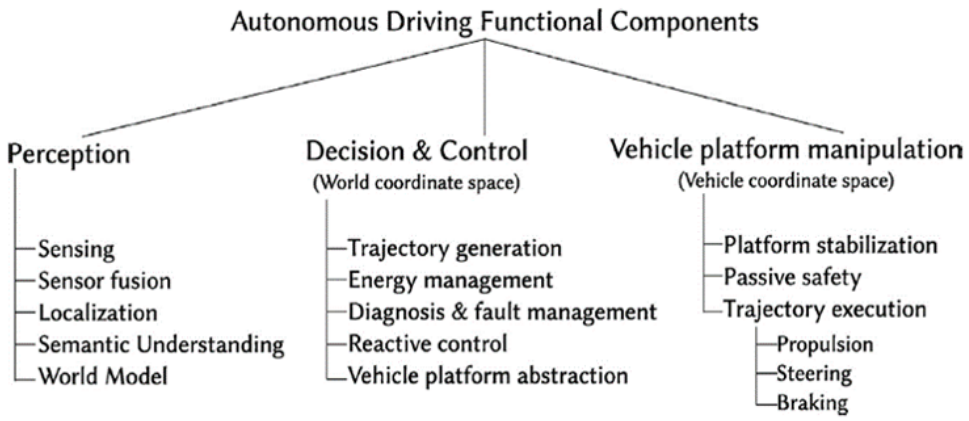


Figure 9: The components of a Functional Architecture for autonomous driving (Behere & Törngren 2016, p. 139).

Figure 9, shows a number of functions within the three components. Comparing the high-level architectures in Figures 8 and 9, shows that the SA component in Figure 8, is divided into the three levels of: perception, comprehension and projection, however the perception component of in Figure 9 does not have a layered structure. Although the SA component in Figure 8 and the perception component in Figure 9 have the same definition, the former one has a layered structure while the latter one does not. In this thesis the layered structure and the term SA is adopted, the definition of which is discussed in detail in section 3.1.1.

3.1.1. Situational Awareness

SA, is a widely used term, especially in the field of humans acting as rational agents. The widely accepted definition of human SA is defined as 'the perception of the elements in the environment within a volume of time and space, the comprehension of their meaning and the projection of their status in the near future' (Endsley 1995). The definitions of the three levels within SA are:

1. **Perception**: The first level of SA is to 'perceive the status, attributes and dynamics of relevant elements in the environment' (Endsley 1995).

2. **Comprehension**: In this level, the agent integrates the perceived information with its goals and objectives. It involves the prioritisation of the observed information, with regards to its importance towards the mission of the agent. Hence, at this level the agent is not only aware of the elements around it but also develops an understanding of their significance with regards to the mission goals (Endsley 1995). This is done by synthesising disjointed data from perception and comparing the information to the goals and objectives (Endsley 2016).

3. **Projection**: This level is the ability of the agent to predict future states and actions of the elements in the environment based on current perception, comprehension and background knowledge (Endsley 1995).

Although the SA definition above was developed considering humans, it can be used for any cognitive agent, with the three levels applicable to autonomous systems (Gehrke 2009, Adams 2007). In the rest of this thesis, Endsley (1995) definition of SA is adopted.

Having identified the three main components of SA, decision and actuation of the high level architecture, as well as the three levels of SA, a review of existing AV functional system architecture (FSA) is presented in the following section.

3.2. State of the art Functional System Architectures of Autonomous Vehicles

The identification of the three main components of a cognitive agent lead to a more detailed investigation of existing FSAs. Existing FSAs give an insight into the operation of existing AV prototypes. This section presents existing FSAs for a better understanding of the functions and structure commonly used.

Some of the reviewed architectures are related to vehicles that took part in the Defense Advanced Research Project Agency (DARPA) challenges, where the aim of the challenge was to accelerate the development of autonomous vehicles and their required technologies.

The functional architectures discussed in this section are the Stanford entry in the 2005 DARPA challenge, the BOSS entry in the DARPA Urban challenge in 2007, the CMU, the A1 and Ulm university's autonomous vehicle architecture and the more recent architectures of Waymo, Mobileye, Wayve, Cruise and Udacity's Carla.

3.2.1. Stanley entry in the DARPA Grand Challenge 2005

The winner of the 2005 DARPA Grand Challenge was the Stanford Racing team, with the Stanley autonomous vehicle. The FSA of the Stanley AV is shown in Figure 10. As data pass through the different components of the architecture, they are transformed from sensor data, to internal models, abstract plans and concrete vehicle controls (Montemerlo et al. 2006). The specific FSA is composed of six major components, the sensor interface, perception, planning and control, vehicle interface, user interface and the global services.

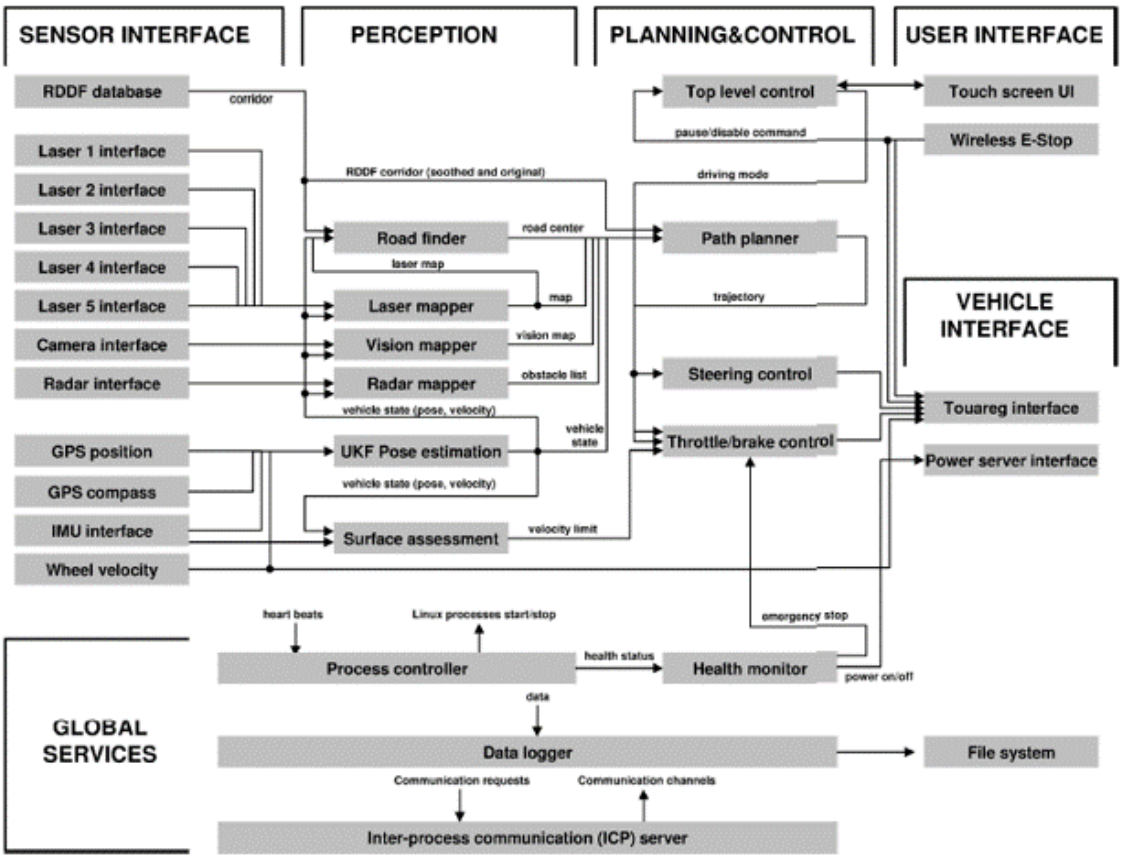


Figure 10: The Stanley's software flowchart (Montemerlo et al. 2006, p. 983).

The sensor interface links the sensors to the rest of the system enabling the input of the time-stamped data to the perception component. This component updates the internal model of the environment by mapping the data from the sensor interface (Montemerlo et al. 2006, Russell & Norvig 2010). The pose estimation function determines the vehicle

position, its velocity and orientation. The three mapping functions use information from the cameras, lasers and radars to develop a 2D environment model, while the road finder uses the outcome of the laser mapper to find the road boundaries which help the vehicle to centre itself within the lane. The sensor interface and perception components work together to develop the model of the environment within which the vehicle operates. The functionality of both components correspond to the SA component of a cognitive agent, where the agent understands the sensor information through perception and develops a model of the environment through comprehension. However, the last SA layer of projection does not appear to be considered in the Stanley FSA. The exclusion of the projection level might not have been an issue for the operation of the vehicle in 2005 DARPA Grand Challenge, since the vehicle was operating in a relatively controlled environment. However, excluding the projection level will compromise the safety of the operation when the vehicle operates in uncontrolled environments with surrounding agents acting in unpredictable ways.

The planning and control component of Stanley's FSA uses the environment model of perception to plan the path to be followed. The planned trajectory is then used to determine the steering, throttle and brake response of the vehicle. This component includes the functions that make decisions about the way the vehicle should move and hence is equivalent to the decision component of the cognitive agent.

The vehicle interface communicates the controls set by steering and throttle/brake control to the vehicle interface, to execute the actions. Additionally the user interface layer provides the means for interaction between the user and the vehicle. Both of these components illustrate how the vehicle interacts with its environment. This interaction is more obvious through the vehicle interface, however the user interface is the component where the vehicle interacts with its user. Both of these components can be considered as being equivalent to the actuation component of a cognitive agent as this is the component where the agent interacts with its surrounding.

The final component of the Stanley's FSA is the global services layer that monitors the operation of the whole system (Montemerlo et al. 2006). This is an essential component

in ensuring the system operates as expected. Another important aspect of this FSA is that the processes run in parallel (Russell & Norvig 2010). For example, the perception component processes the current and most recent sensor data, while planning and control takes decisions based on slightly older data. The whole process runs asynchronously in a similar way to how the human brain operates (Russell & Norvig 2010), enabling the vehicle to behave as a cognitive agent. Although the components are named differently and the structure seems to have more than three main components, the logic of a cognitive agent can still be seen through the Stanley FSA.

In addition to winning the 2005 DARPA Grand Challenge, the Stanford Racing team won the second place in the 2007 Urban Challenge. Stanford Racing team's entry in the 2007 Urban Challenge was 'Junior', which had a very similar FSA to Stanley, presented in Figure 10.

3.2.2. Boss entry in the DARPA Urban Challenge 2007

The vehicle that won the first place in the 2007 DARPA Urban Challenge, was 'Boss' developed by the Tartan Racing team from Carnegie Mellon University (CMU) (Luettel et al. 2012). The FSA of Boss is illustrated in Figure 11. The four major components of Boss are the perception, mission planning, behavioural executive and motion planning.

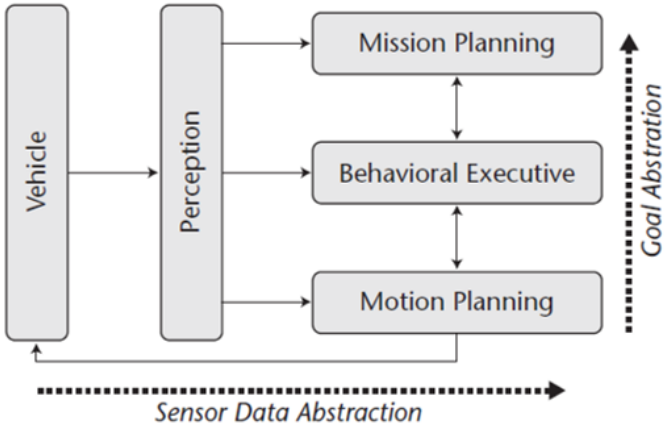


Figure 11: The Boss software architecture (Urmson et al. 2009, p. 20).

The perception component of Boss, develops a model of the environment for the rest of the components to use. The environment model is developed by fusing the information obtained from the sensors and includes information about the current state of the vehicle, current and future state of other vehicles, and moving and static obstacles. This results in the environment model providing information about three main features in the environment, namely, a map of any static obstacles, a list of any moving vehicles and the location of the vehicle relative to the road (Urmson et al. 2008). The description of Boss's perception component matches the SA component of a cognitive agent and hence Boss's perception is considered equivalent to the SA of a cognitive agent. However, as with Stanley's and Junior's FSAs, the projection component of SA is not considered. Instead, these FSAs execute only the perception and comprehension component of SA and stop when the current model of the environment is being developed. The information provided to the decision making components is hence the current state of the environment without any predictions.

Boss's mission planning component is responsible for determining the optimal path to a particular checkpoint. For the comparison of the different routes, it considers any relevant information about road blockage, maximum legal speed limits and time required to make one manoeuvre versus another (Urmson et al. 2009, 2008). A graph that determines the connectivity of the environment is also developed using data provided in a Road Network Definition File (RNDF), given to the teams. This implies that an RNDF is required for the operation of the vehicle, which might not be a functional solution at a larger scale. In situations where a new path needs to be modified, the relatively small size of the graph allows for a very quick re-planning, which results in a near-immediate response to detected changes (Urmson et al. 2008). However, this can be used as an indication that a larger graph does not result in a near-immediate response, which might lead to unsatisfactory results.

Behavioural executive is responsible for providing the motion planning component with a sequence of incremental goals to execute (Baker et al. 2008). It uses the global route information determined by the mission planner in combination with the environment model

provided by perception, to determine the appropriate motion goal (Urmson et al. 2009). It considers the rules of the road to make lane-change, precedence and safety decisions on road and at intersections (Urmson et al. 2008). An additional functionality of this component is its responsibility to handle error recovery in anomalous situations (Urmson et al. 2008).

Motion planning receives the motion goal from the behavioural executive and determines the trajectory that drives the vehicle through the required path. This component is responsible for avoiding any dynamic or static obstacles, while achieving the required goal. This component consists of two sub-components, structured and unstructured driving. The former consists of a planner that aims to follow the road lane avoiding any obstacle in its way. Alternatively, the latter consists of a planner that manoeuvres in an unstructured environments, such as a parking space. The outcome of the motion planner is then received by the vehicle controller for execution (Urmson et al. 2008).

The mission planning, behavioural executive and motion planning of Boss, represent different levels of planning and are in turn equivalent to the decision component of a cognitive agent. All three planning components of Boss receive information from the perception component to make decisions which are sent back to the vehicle. This makes the vehicle equivalent to the action component of a cognitive agent, as 'vehicle' is Boss's component that interacts with the environment, either passively or actively. The passive interactions is through the sensors, where the agent observes the environment where the active interaction is through the actuators where the agent acts within that environment.

3.2.3. Carnegie Mellon University autonomous vehicle

Following the development of Boss for the DARPA Urban Challenge in 2007, CMU introduced a new autonomous vehicle known as the CMU autonomous vehicle, considered closer to series production in comparison to Boss (Taş et al. 2016). The FSA of the CMU autonomous vehicle is shown in Figure 12. The vehicle has a very similar FSA to Boss, shown in Figure 11. It consists of the same four major components of perception, mission planning, behavioural generation and motion planning as Boss. Figure 12 illustrates the

flow of information through the different components in more detail, compared to Figure 11, indicating the input and output information of the different components.

The perception and environment modelling components create a model of the world, by fusing data from multiple sensors (Wei et al. 2013). The mission planner uses information from the road network to determine the optimal route, while the behaviour component makes decisions for the execution of the optimal route. Finally, the motion planner is responsible to avoid any static and dynamic obstacles. The decision of the motion planner is sent back to the controllers as vehicle commands. The structure of both vehicles is the same, as both were developed by the same team, and both seem to align with the concept of a cognitive agent.

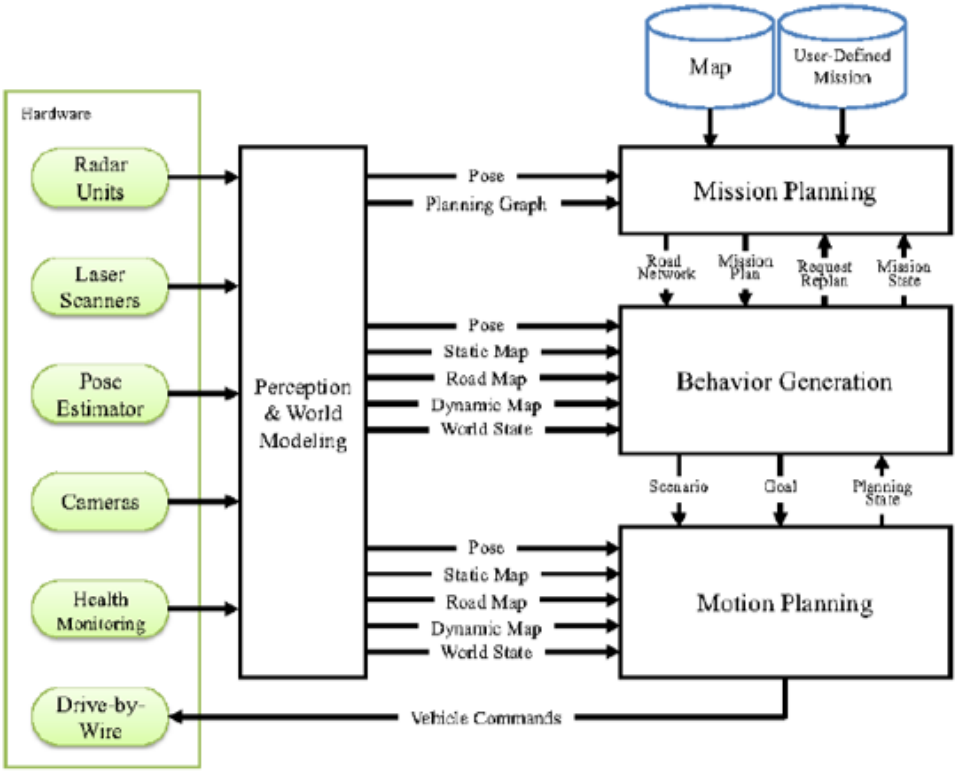


Figure 12: The software architecture of CMU autonomous vehicle (Wei et al. 2013, p. 769).

3.2.4. A1 autonomous vehicle

Another competition, known as the Autonomous Vehicle Competition (AVC) was held in 2012 in Korea, by Hyundai Motor Group (Taş et al. 2016, Jo et al. 2015). Participating vehicles were required to demonstrate their ability to operate in urban driving environments. The winner of this competition was the autonomous vehicle A1, which was able to successfully execute all the required missions. The FSA of A1 is composed of five basic functions, namely, localisation, perception, planning, vehicle control and system management, as illustrated in Figure 13.

The localisation component uses localisation sensors to determine the vehicle's position on the map. This is essential information used by the vehicle to plan the optimum path. The perception component utilises the information collected through the sensors to develop a model of the surrounding environment within which it is operating (Jo et al. 2015). The planning component uses the information from localisation and perception to determine the trajectory to be followed for safe navigation. These components correspond to the SA and decision components of a cognitive agent. The integration of the localisation and perception component of A1 is equivalent to the SA of the cognitive agent, while planning is equivalent to the decision component of a cognitive agent.

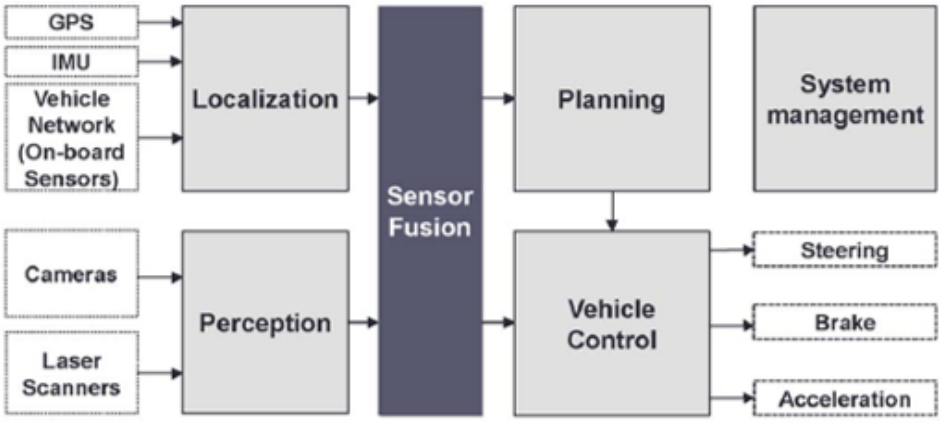


Figure 13: The structure of the autonomous driving algorithm in A1 (Jo et al. 2015, p. 5121).

The planning algorithms are divided into three layers, namely global routing, behaviour reasoning and local motion planning. The global routing sub-component determines the most optimum path for the whole journey (from origin to destination). Behaviour reasoning is responsible for determining the behaviour of the vehicle by assessing the driving situation and taking into consideration the perception data and path from the global route planner. Within this sub-component, the vehicle considers the road rules and determines the appropriate behaviour while complying with traffic regulations (Jo et al. 2015).

The local motion planning sub-component considers the global route and behaviour reasoning to determine the motion of the vehicle. The aim of this sub-component is to avoid any static and dynamic obstacle, while navigating through the predetermined route. The local motion planner is further divided into two types of path planning algorithms, the road-map-based and free-form path planning. The road-map-based path planning is utilised in structured environments such as roads, to execute tasks such as lane keeping. Alternatively, the free-form path planning is used in unstructured environments, such as roads under construction and parking areas (Jo et al. 2015).

The definitions of the planning sub-components of vehicle A1, are similar to some of the components used for both Boss and the CMU autonomous vehicle. Global routing of A1 is equivalent to mission planning of CMU and Boss. Behaviour reasoning of A1 is the same as behavioural executive of CMU and Boss and while local motion planning of A1 corresponds to as motion planning of CMU and Boss. Moreover, the sub-components of A1's local motion planning namely, road-map-based and free-form path planning are equivalent to structured and unstructured driving respectively, presented in Boss. This is illustrated in Figure 14.

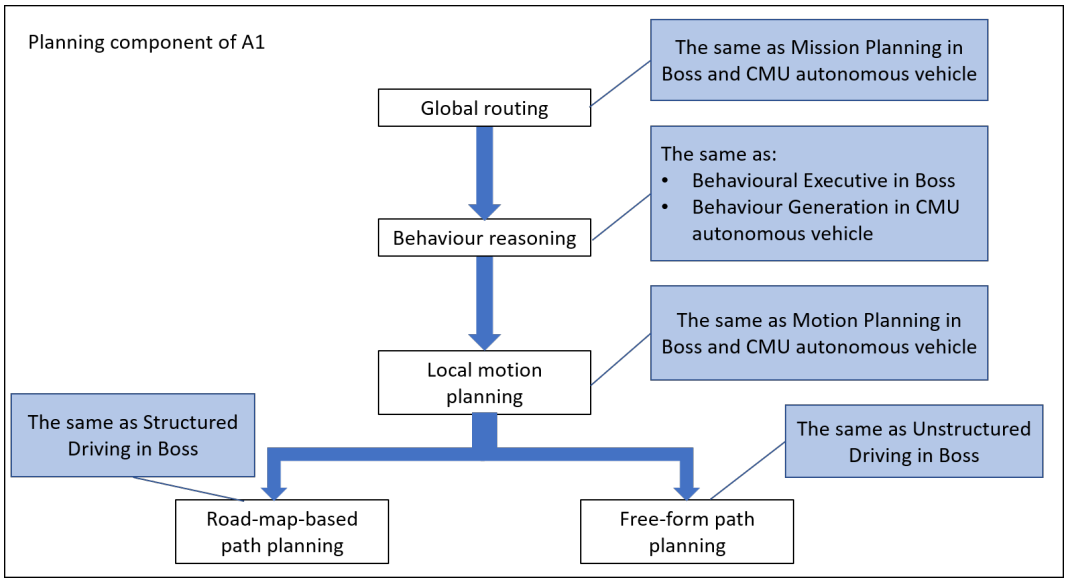


Figure 14: The planning component of A1 and its similarities to Boss and CMU autonomous vehicle.

The next basic component of A1's FSA is vehicle control. The vehicle control function guides the vehicle along the trajectory derived by the planning function. This component is divided in two sub-components, namely lateral controller and longitudinal controller. The former determines the target steering angle, while the longitudinal controller determines the target position of the acceleration and brake pedals.

The vehicle control component of A1 together with the sensors of the vehicle correspond to the actuation component of a cognitive agent. It determines the exact commands for the actuators based on the decisions of the planners, while the sensors observe the environment. Both of these components interact with the external environment either actively or passively as mentioned earlier. As a result, these two components are considered equivalent to the actuation component of the cognitive agent.

The final component in the A1's FSA is the system management, which is responsible for supervising the entire autonomous driving system. An equivalent component is in Stanley's FSA, is the global services component in Figure 10. The presence of this component in the reviewed architectures, emphasises the importance of management system and it is hence

considered as an essential component of the FSA.

The FSA of A1 is also presented by Taş et al. (2016), with a slightly different structure, where the five components in Figure 13 as split between the three components of perception and scene understanding, behaviour and motion planning and Vehicle Control and actuation as shown in Figure 15. Figure 15 shows a structure equivalent to a cognitive agent with its three components corresponding to the SA, decision and action component of a cognitive agent.

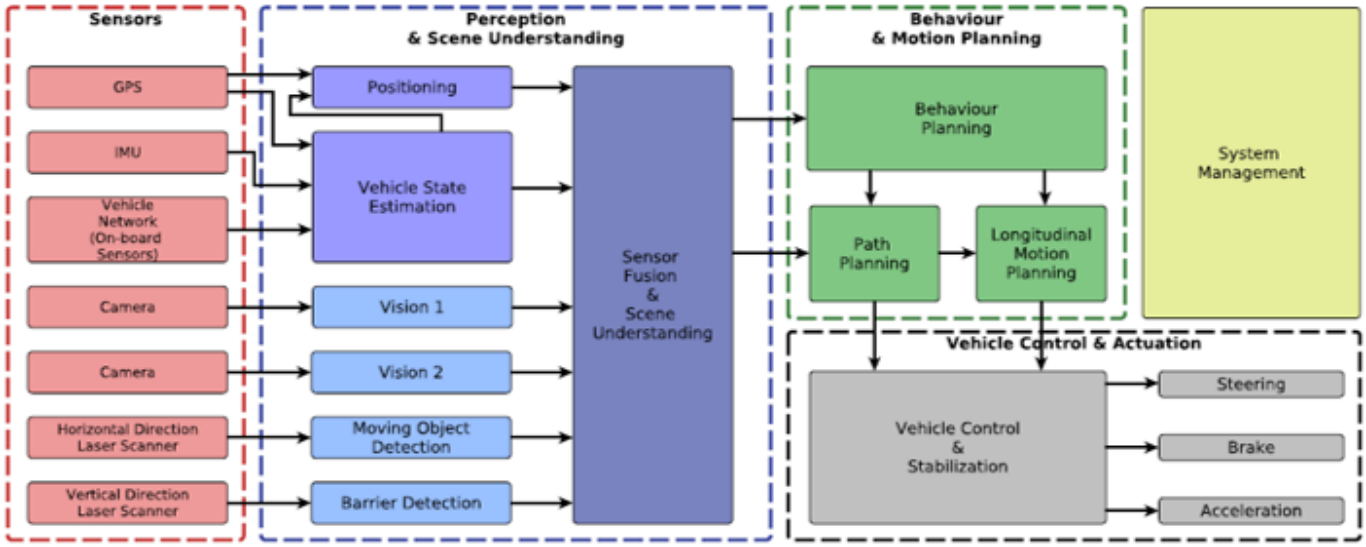


Figure 15: The information flow between the modules of the A1 automated vehicle (Taş et al. 2016, p. 307).

3.2.5. Ulm University autonomous vehicle

Another example of autonomous driving is the autonomous vehicle of Ulm University. The major components of the specific FSA are sensor interface, environment perception, motion planning and vehicle control (Kunz et al. 2015). Environmental perception is divided in three layers, object-tracking, localisation and grid mapping. Grid mapping divides the environment intro grid cells and estimates the occupancy of each cell, localisation estimates the vehicle's pose and object tracking detects and tracks surrounding objects. Each of these layers fuses the data collected from different sensors to develop an element of

perception. The three perception layers are then combined together to develop the environment model. The perception component of this FSA is equivalent to that of a cognitive agent, since it develops the model of the environment.

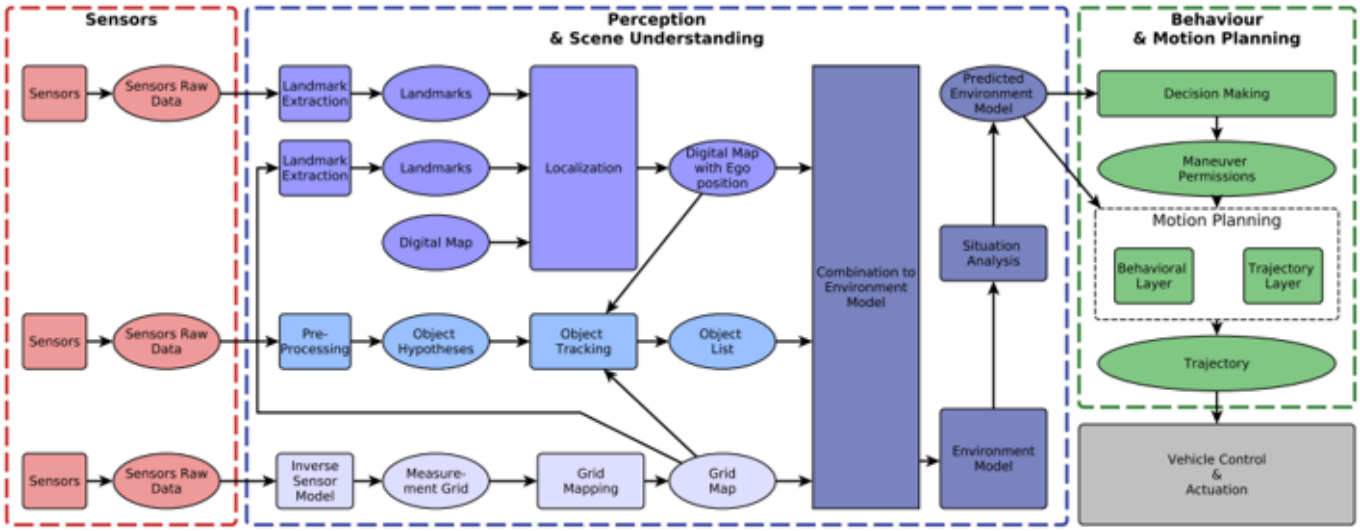


Figure 16: The functional system architecture of Ulm University's autonomous vehicle (Taş et al. 2016, p. 308).

The behaviour and motion planning part, is equivalent to the decision component of a cognitive agent. It receives the environment model and plans the trajectory from origin to destination. As with Jo et al. (2015), Wei et al. (2013), Urmson et al. (2009), behaviour and motion planning of the Ulm autonomous vehicle is formed with different layers of planners ranging from decision making, to trajectory planning. This illustrates that the structure of this component is similar to the one presented in Figure 14. The layered structure of the decision component is further discussed in section 3.4.

3.2.6. Recent advances

Some of the most recent developments in autonomous driving are presented by Waymo, Mobileye, Cruise, Wayve and Udacity (Waymo 2024, Mobileye 2024, Zucker 2019, Wayve 2021, Udacity 2020). Autonomous driving in Waymo, Cruise, Mobileye and Udacity's Carla, consist of generally the three main components of perception, decision and vehicle

control.

The typical structure for autonomous driving is presented by Ren & Xia (2023), as shown in Figure 17. The modular architecture with the three main components, is equivalent to the structure of a cognitive agent discussed in section 3.1. Wayve, refers to this modular architecture as AV 1.0 (shown in Figure 18) and presents an alternative architecture AV 2.0, where perception and decision making are replaced with a neural network taking as inputs the raw sensor data and returning the driving tasks.

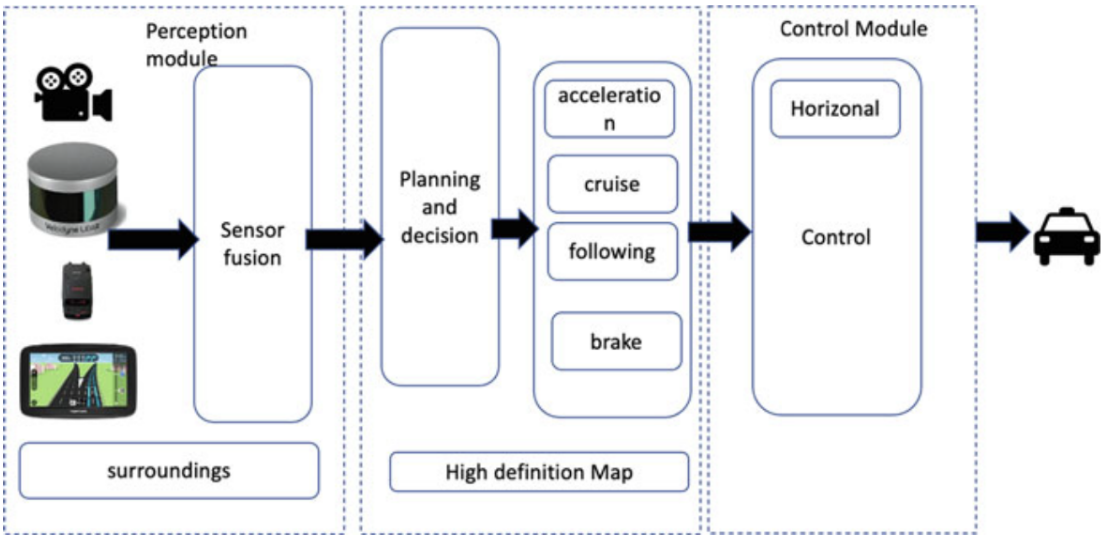


Figure 17: Main components of autonomous driving (Ren & Xia 2023).

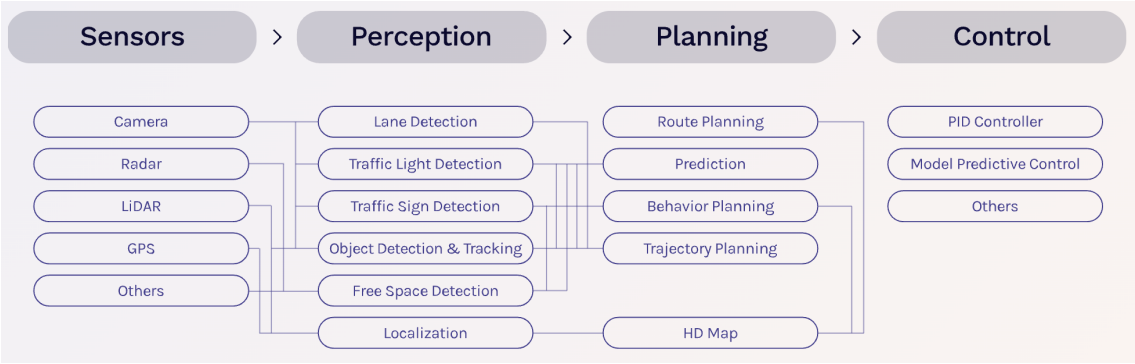


Figure 18: The modular structure of AV1.0 presented by Wayve (Wayve 2024).

The reviewed FSAs, show some similarities in their structure and functionality. The

following section compares the FSAs reviewed, to illustrate the main components and functions.

3.3. Comparison of existing architectures

The reviewed architectures discussed in section 3.2, have many similarities in their structure, as well as in the functions they use. In order to identify the key functions in the FSA, a summary table is developed. The functions are split into three components, namely the "Sensing, Communication and Control", "SA" and "Decision and Planning" component. These correspond to the three main components of a cognitive agent discussed in section 3.1.

The sensing, communication and control component includes vehicle sensors and actuators. It is the component where a vehicle interacts with the external environment in either a passive or active way, by receiving information or taking actions within the surrounding environment. The SA component is responsible for providing an understanding of the surrounding environment, determining vehicle location, developing a model of the surrounding environment and predicting how the elements within that model will act in the near future. Finally, the decision and planning component uses the predicted model of the environment to make decisions about the appropriate vehicle manoeuvres. The use of these three components follows the rationale presented in section 3.1.

An additional component, monitoring the operation of the whole system was used in some of the reviewed FSAs (Montemerlo et al. 2006, Jo et al. 2015, Wei et al. 2013). Although this component is important in ensuring the safe operation of the system, it is not related with the way a cognitive agent operates and it is more applicable to autonomous agents. For this reason, even though it is included in this thesis as "system management", its functionality is not considered. The four main components of the FSA are hence "Sensing, Communication and Control", "SA", "Decision and Planning" and "System management" and the three components considered in more detail are the component of a cognitive agent: "Sensing, Communication and Control", "SA", "Decision and Planning".

In Figures 19, 20 and 21, the FSA functions, are distributed to the four components

based on their definition. The colour of each of the cells in the table illustrates whether the specific function is present in the architecture of the specific vehicle. The green colour illustrates the function is present, and the corresponding name of the function as presented in the reviewed FSA. The yellow colour illustrates the a function is not clearly stated within the specific FSA, but a description of its functionality is provided, while the white colour illustrates the specific function is not mentioned within the FSA. Figures 19, 20 and 21, provide a visual representation of the split of functions used in the reviewed FSAs, and enable the identification of functions commonly presented in the literature (such as localization, motion planner etc.).

Year	Vehicle	Sensing Communication and Control		
		Sensors	Actuators	
			Steering control	Brake/Throttle control
2005	Stanley		Reactive controller	Reactive controller
2007	Boss			
2012	A1		Vehicle Control	Vehicle Control
2013	CMU		Path Tracker	Speed Tracker
2013	Bertha		Lateral control	Longitudinal control
2015	Ulm		Lateral controller in the vehicle control module	Longitudinal controller in the vehicle control module
	Proposed Architecture		Lateral Controller	Longitudinal Controller

Figure 19: The different functions of the vehicle functions.

Year	Vehicle	Situational Awareness											
		Moving Obstacle Detection and Tracking	Static Obstacle Detection and Mapping	Object classification	Roadmap Localization	Road shape estimation	Vehicular Communication as a sensor	Surface assessment	Traffic Light Recognition	Multi-layer fusion	Combination to environment model	Situation Analysis	Predicted Environment model
2005	Stanley		Mapping modules		UKF vehicle state estimation	Road finder							
2007	Boss												
2012	A1	Moving Object Detection	Barrier Detection	Vision algorithm	Positioning & Vehicle state estimation				Done in the vision system		Sensor fusion		
2013	CMU				Localization system	Lane marking detection using cameras							
2013	Bertha	Vehicle and pedestrian Recognition		Object classification	Two complementary vision algorithms: point feature based and lane marking based localization						Object fusion		
2015	Ulm	Multi-object Tracking	Grid mapping	Use classifiers to detect vehicles	Localization	Use of boundary lines which limit the drivable space				Hierarchical Modular Environment Perception System (HMEP) 3-layer fusion (Grid mapping, Localization	Combination module		
	Proposed Architecture	Moving Object detection and tracking	Static object detection and mapping	Found in both Object detection components	Localization	Found in Road evaluation	Vehicular communication	Found in Road evaluation component	Found in Static object detection and mapping	5-layer fusion (Moving object detection and tracking, Static object deterction and mapping, Road evaluation,	Environment Model	Environment Projection	Predicted Environment model

Figure 20: The different functions of the situational awareness functions.

Year	Vehicle	Decision and Planning				System Management
		Route Planner	Mission Planner	Behavioural Planner	Motion Planner	
2005	Stanley		Top level control		Path planner	
2007	Boss					
2012	A1	Global routing		Behavior reasoning	Local motion planning	System management module.
2013	CMU					
2013	Bertha	Optimal driving paths have been calculated in an off-line step.		Behavior generation	Trajectory planner	
2015	Ulm	Route	Decision making	Behavioral Layer in Motion Planning	Trajectory Layer in Motion Planning	
	Proposed Architecture	Route Planner	Mission Planner	Behavioural Planner	Trajectory Planner	System Management

Figure 21: The different functions of the decision and planning and the system management functions.

Comparing the existing architectures helps in understanding the functions within each component of the FSA, required for the operation of an AV. However, reviewed architectures, only support the operation of an AV within its surrounding environment, without considering how the AV moves from origin to destination (O-D), at a macro level. The consideration of a macro level is only reflected in the decision and planning component. The different layers of planners correspond to different levels of planning granularity, with the route planner considering the AV O-D trajectory and motion planner the manoeuvres within the AV's immediate environment. Even though, the decision and planning component has a layered structure, this is not reflected by the rest of the components. This leads to potential issues in understanding the resolution of information required from the different decision and planning layers, which subsequently leads in lack of understanding of PNT requirements of each function. The presence of a layered structure within the reviewed FSAa is discussed in the following section.

3.4. Levels of Decision and Planning

The current architectures focus on the operation of an AV within its immediate environment. As opposed to the driving task being focused only on the immediate environment, a number of studies suggest that there are different levels of driving tasks (Donges 1999, Du et al. 2004, Matthaei & Maurer 2015, Ulbrich & Maurer 2015). These levels vary in resolution, accuracy and scale. The levelled structures is mainly seen in the decision and planning component. Three levels of planning are defined by Donges (1999), namely navigation, guidance and stabilisation. These levels are referred to, by Ulbrich & Maurer (2015), as strategical, tactical and operational levels respectively and represent different resolutions of planning. Strategical considers the overall path the vehicle will follow. Tactical decides on the behaviour of the vehicle as it executes the planned path and operational develops a more refined scale of the path based on both strategical and tactical. Matthaei & Maurer (2015) use the same levels and map them to the different scales of navigation presented in Du et al. (2004), namely, macro-scale, meso-scale and micro-scale respectively. Although the name of the levels might vary, it is clear that a hierarchical and levelled structure is commonly used within the planning component.

A levelled structure is also sometimes presented in the sub-component of localisation. Meng et al. (2018) and Basnyake et al. (2010) use four levels of localisation, namely 'Which road', 'Which lane', 'Where in lane' and 'Active control'. These levels correspond to different localisation resolutions. 'Which road' identifies the road on which the vehicle is moving. It is used by functions that require accuracy of at least 5m, such us journey planning from origin to destination. 'Which lane' identifies the lane in which the vehicle is moving. This corresponds to at least 1.5m accuracy and can be used for applications such as speed adaptation. 'Where in lane' corresponds to at least 0.5m of accuracy, which can be used for applications like collision avoidance. Finally, 'Active control' relates to an accuracy better than decimeter, which is usually applicable for applications such us autonomous vehicle control. These four levels illustrate another implementation of a levelled structure corresponding to different localisation resolutions.

Additionally, a widely used terminology in the field of AVs, is lane-level accuracy. Lane level accuracy is mainly used by functions that focus on keeping the vehicle in lane. It is the level of accuracy generally required by advanced driving assistance systems (ADAS) (Kang et al. 2020). Lane-level accuracy is equivalent to either 'Where in lane' or 'Active control' localisation described earlier. This emphasises the fact that different functions have different accuracy requirements which are determined by the levels of localisation. Hence localisation has different resolutions and can be used by functions with the equivalent accuracy requirements. The layered structure is again presented in the case of accuracy requirements.

The definitions of the four localisation elements seem to translate into a specific level of accuracy. However, it can also be seen that the greater the accuracy requirements, the more safety critical a function can be. This can be related to higher requirements in terms of integrity, which as mentioned earlier is the level of trust in the system. A layered structure will hence aid in identifying safety critical functions and determine the most stringent system requirements.

Even though there is evidence of layered components, they all seem to focus on the component of decision and planning. The literature has not yet considered any levels required from other components. The different decision and planning levels require information of equivalent resolution (or refinement). Although this is seen by the different levels of localisation, it is not evident in the rest of the components required for decision and planning. In addition to localisation, information received from the SA and vehicle components will also need to be of equivalent refinement. This layered structure for the vehicle and SA components will aid the operation of decision and planning.

3.5. Summary

This chapter has reviewed the literature of existing FSAs for the operation of a cognitive agent and in particular an autonomous vehicle. Initially, the concept of a cognitive agent was introduced. The three main components of SA, Decision and Actuation required for the operation of a cognitive agent were identified. The SA component develops an

understanding of the surrounding environment. Decision component uses the developed understanding to take decisions about how the agent will operate in that environment and finally the actuation element executes the decision. The term SA was introduced with its definition emerging from the field of human SA. The three layers of SA is the perception, comprehension and projection, through which a moving agent develops a good understanding of its surrounding. The structure of a cognitive agent is hence composed of the three components of SA, decision and actuation and the three SA sub-components of perception, comprehension and projection.

An AV needs to operate autonomously within the network and its surrounding environment and is hence used as an example of a cognitive agent for the development of the FSA. A review of the literature of existing FSA for the operation of AV was carried out. A comparison of the reviewed FSAs demonstrated that although the names of functions presented in different FSAs vary, the definitions of some functions are similar, illustrating consistent presence in most of the architectures. All the reviewed FSAs have components and sub-components which include the different functions. However, the components and the split of the functions between the component are not consistent amongst the FSAs. The lack of a coherent structure makes it hard to understand the inter-dependencies between different components and functions. Without a concrete understanding of the inter-dependencies, inefficiencies and safety concerns might arise.

Furthermore, the reviewed FSAs mainly focus on the operation of the AV within its immediate environment. However, it is important that there is an understanding of the vehicle's operations at a higher level. This is partially considered in the literature through the component of Decision and Planning, where in some cases up to 4 levels of planning are being considered (Donges 1999, Du et al. 2004, Matthaei & Maurer 2015, Ulbrich & Maurer 2015). Each level represents a different resolution of planning starting with a route planner and refining the planning all the way down to a trajectory generator. The layered structure is also observed within the localisation functions of some architectures, where localisation can vary from road to lane level. This is a consequence of the layered decision and planning component, since every planning component require a different level of localisation that is

usually determined by its accuracy. However, the literature does not consider any layers in the components of SA and actuation. A layered structure in these components will ensure the different levels of decision receive the required information at the required resolution. This will further aid the process of understanding the requirements of every function.

This chapter presented a review of the literature and identified the lack of structural consistency amongst existing FSAs and the lack of a layered structure across all components as limitations in the literature. The combination of these lead to challenges when identifying critical functions in the FSA. The issues identified in this chapter necessitate the specification of a comprehensive FSA to support the development of the corresponding functional and physical architectures with a particular focus on LBS and hence PNT. To develop a more coherent structure, the proposed FSA follows the structure of a cognitive agent. The functions required for the operation of the AV are hence distributed in the three main components of a cognitive agent. Additionally, a layered structure is proposed, where different resolution of information is used to support the different levels of decision and planning. The development of the proposed AV FSA, its transferability for the operation of other cognitive systems and the identification of location based functions (LBFs) within the FSA are presented in Chapter 4.

Chapter 4

Enhanced functional system architecture for dynamic autonomous agents

In this chapter, the proposed functional system architecture (FSA) of a cognitive agent is developed to address the limitations of structural inconsistency amongst existing FSAs and lack of a layered structure across all components, outlined in sections 3.3 and 3.4. The FSA is developed based on the concept of an autonomous vehicle (AV) and is adapted to other cognitive agent systems, such as human driven vehicle and visually impaired people using a smart cane. The FSA builds on strengths of existing architectures reviewed in chapter 3 with some enhancements. The structure of a cognitive agent is adopted, with three main components of SA, decision and actuation. In addition, a layered structure is proposed, where the cognitive agent considers its state at a macro-scale level to support the different resolutions of information required for different levels of decision making. The proposed FSA is finally used to identify the location based functions (LBFs), as the functions that require position information for their operation.

Section 4.1 presents the methodology implemented for the development of the proposed FSA. The three main components of the FSA, including their functions and the enhanced AV FSA are presented in section 4.2. Section 4.3 illustrates the transferability of the enhanced FSA, by adapting it for the case of human driven vehicles and visually impaired people using a smart cane. Finally, section 4.4 identifies the LBFs within the developed AV FSA.

4.1. Methodology

The method used for the development of the FSA of a cognitive agent, is illustrated in Figure 22. The initial step identifies the three main components of a cognitive agent (section 3.1), with the three SA sub-components defined in section 3.1.1. The SA sub-components of perception, comprehension and projection, are defined based on human SA proposed by Endsley (1988). The structure of the cognitive agent and the components of SA are used in this Chapter as the basic structure for the development of the FSA.

Following the definition of a cognitive agent and SA, a review of the literature of existing FSAs was carried out in section 3.2, revealing some similarities in the structure as well as functions used within existing FSAs. In section 3.3, the functions of the reviewed FSAs were distributed between the three components of a cognitive agent, using their corresponding definitions. In this Chapter, the definition of the functions identified in the literature, is used to distinguish between functions and sub-functions. This leads to the development of the proposed FSA for an AV operating within its immediate environment.

A closer look into the decision and planning component, in section 3.4, showed that there are different levels of planning, using different resolution of the same information. For example, all levels of planning require position information, however, at the higher levels of planning, less accurate position information is sufficient to determine the path from origin to destination. This leads to an additional level being introduced in the FSA proposed in this chapter, referred to as the Marco scale level. This level considers the state of the vehicle within the network and completes the proposed FSA of an AV operating in its immediate environment as well as the network.

The developed FSA for AV operation is then adapted to represent the operation of human driven vehicles as well as visually impaired people (VIP). This illustrates the tansferability of the proposed FSA to other cognitive agents other than AVs. The FSA adapted for VIP navigation, has been implemented during a project focusing on VIP navigation and way finding, illustrating the applicability of the proposed FSA.

Following the development of the AV FSA, the location based functions (LBF) are

selected. The selection of LBF is achieved by considering the functionality and information required from each function, for its operation. The LBFs, by definition demonstrate the need for positioning information.

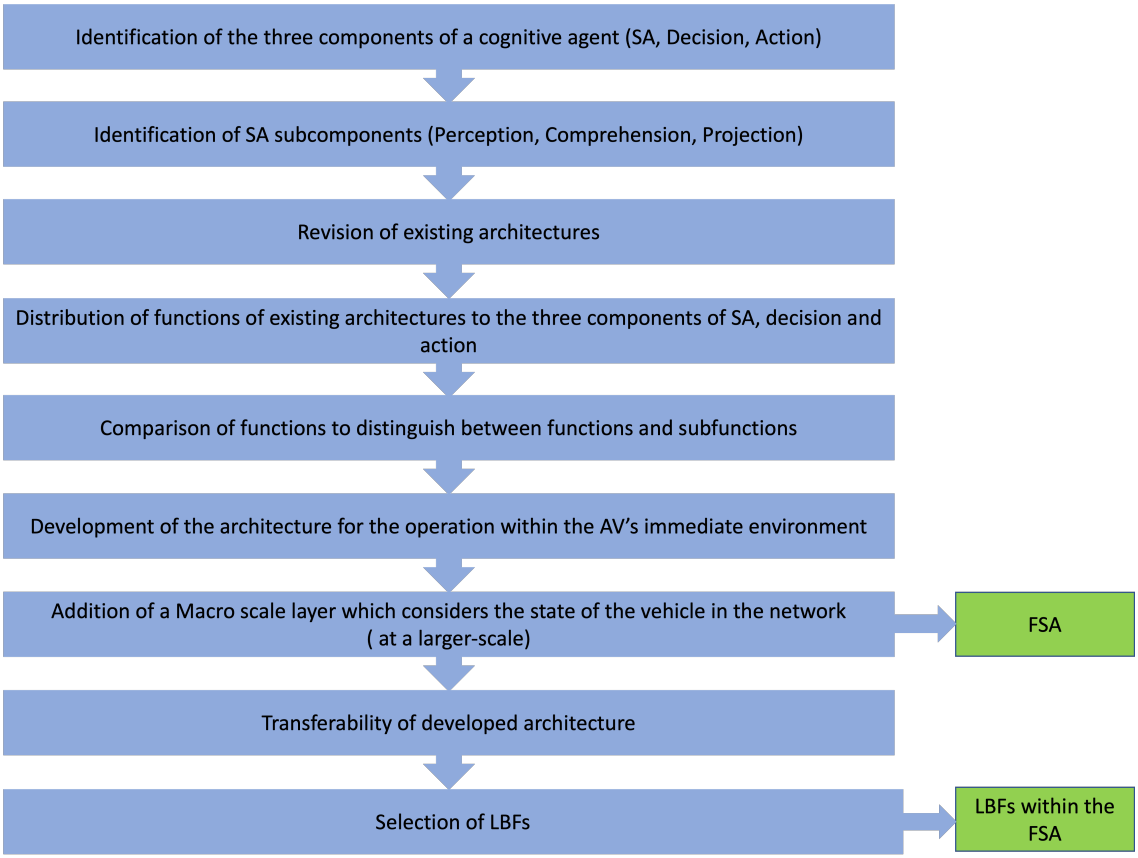


Figure 22: Method used for the development of the FSA.

4.2. Proposed functional system architecture

The comparison of the reviewed FSAs in section 3.3 and their functions presented in Figures 19, 20 and 21, are used for the development of the proposed FSA. The three main components of a cognitive agent are used as the main three components of the FSA: Sensing, Communication & Control; SA; and Decision & Planning. Each corresponds to actuation, SA and decision of the cognitive agent respectively. For AV FSA development, the three components are developed separately and then integrated together.

In section 3.3, the functions of the reviewed FSAs, were distributed in the the compo-

nents. When considering the definition of those functions, it is observed that some should be sub-functions. For example, in the FSA presented by Kunz et al. (2015), there is the function of grid mapping, within the SA component. This is a process where the surrounding environment is segregated into a grid and the occupancy of each grid cell is estimated. It is mainly used for identifying stationary objects and boundaries. The definition of grid mapping suggests that is a sub-function of static object detection and mapping, rather than a stand-alone function. This shows that although some functions, like grid mapping, in the literature are presented as standalone functions (Kunz et al. 2015), their definition shows that they should be sub-functions of other functions. The classification of identified functions in section 3.3, into functions and sub-functions is hence done using their definitions.

The classification between functions and sub-functions and their split between the three components of a cognitive agent, are used for the development of a FSA for an AV operating within its immediate environments. Additionally, the layered decision and planning component used in some studies, mentioned in section 3.2, is used as a guidance for the development of a two-level FSA. This allows the vehicle to be aware of its state in the network, as well as within its immediate environment. In the proposed FSA, all three components are composed of different levels. For the Sensing, Communication & control, as well as the SA component, there is the split between the following two levels:

- **Macro-scale:** At this level, the vehicle aims to develop a good understanding of its position on the map at a road level resolution, the position of the destination, the layout of the network and appropriate information about the traffic flow. At this level the vehicle is aware of its state in the network.
- **Meso- and Micro-scale:** At this level, the vehicle develops a good understanding of the immediate environment. It creates a model of its surrounding and determines the position, velocity and physical features of surrounding agents and objects.

The Decision & Planning component is split into the strategical, tactical and operational levels. The definitions provided by Matthaei & Maurer (2015) are adopted:

- **Strategical level:** This level plans the path from origin to destination and breaks it down into smaller missions for the vehicle to execute. The information that it requires is at a macro-scale resolution, such as information about the network and traffic flows.
- **Tactical level:** This level makes decisions as the vehicle moves along the path defined by the strategical level. It uses a meso-scale resolution, where static and moving features are identified.
- **Operational level:** This level is for reactive stabilization. It uses micro-scale resolution, where exact geometric values are calculated to make decisions that allow for the execution of the path determined by the tactical level.

As noted, the tactical and operational levels of decision & planning component, require information from meso and micro scale. This distinction is made primarily due to the positioning resolution required by the two planners. The meso and micro scales correspond to positioning accuracy classifications of 'where in the lane' and 'active control' introduced in section 3.4, corresponding to 0.5m and decimetric accuracy respectively. These two accuracy levels combined, are referred to as lane-level pose estimation in this thesis. Both meso and micro scales refer to information associate with the immediate environment of the agent, necessary for the development of SA. As the two tactical and operational planners require information of the immediate environment but different positioning resolutions, meso and micro scales are combined into one to develop the model of the immediate environment.

The remaining of this section presents the proposed FSA, shown in Figure 26 and building on Figures 23, 24 and 25 on the components. It illustrates the functions that are allocated in the three components and different layers of the FSA, as well as the flow of information between them.

4.2.1. Sensing, Communication & Control

The Sensing, Communication & Control component illustrated in Figure 23, is responsible for the interaction of the vehicle with the surrounding environment and the network.

It consists of a sensing & communication, and an actuation element. The sensing & communication element is responsible for communicating with the external environment and obtaining any data relevant to safe driving. It uses sensors (LiDAR, radar, cameras etc) to observe the immediate environment. In addition it uses communication links (V2X) to communicate with other agents, the infrastructure as well as the passenger.

Alternatively, the actuation element is responsible for implementing any decisions taken by the vehicle. It includes a lateral and longitudinal controller, that take as input the outcome of the decision and planning component (discussed later in section 4.2.3), and converts it into actuator commands. The lateral controller determines the commands for steering, i.e the target steering angle, while the longitudinal controller determines the desired acceleration or deceleration.

The Sensing & Communication element, has two levels (Macro-scale and Meso & Micro-scale), while the Actuation element has only one (Meso & Micro-scale). At the Macro-scale level of Sensing & Communication, the vehicle obtains information about the network and traffic flows (through maps and its communication links), the required destination, (through its communication with the passenger) and the origin of the journey (using localisation sensors for road-level localisation). At the Meso & Micro level, the vehicle uses onboard sensors and communication links to obtain data about its surrounding environment. The information collected through sensing & communication is used in the SA component for the development of the environment model.

4.2.2. Situational Awareness

The SA component is split into Macro-scale and Meso & Micro-scale level. At the Macro-scale level, the vehicle obtains all the relevant data from sensing and communication to map the origin and destination of the journey as well as develop a good understanding of the network and traffic flows. At the Meso & Micro-scale, the vehicle develops an accurate model of its immediate surrounding.

The Meso & Micro-scale layer is structured based on the the definition of human SA, presented in section 3.1.1. This is done since an AV and any cognitive agent should be able

to observe, reason and act within its surrounding environment (Skrypchuk et al. 2018) in a way equivalent to a human. The definition of the functions identified in Figure 20 allows for their classification amongst the three SA sub-components of perception, comprehension and projection. This results in the structure of the SA component illustrated in Figure 24.

The perception component is about understanding the information captured through the sensors and communication links. Comprehension gathers the understanding of the perceived information and prioritises it with regards to the information's importance towards the driving goal. This information is then used to create a model of the surrounding environment. The projection component uses the model created by comprehension to make predictions about the behaviour of surrounding agents in the near future, (within a few seconds).

The perception component is composed of the five main functions of object detection, localisation, self-perception, road evaluation and vehicular communication. Object detection refers to both moving and static objects detection. Moving object detection and tracking is responsible for detecting any dynamic objects, classifying the objects by identifying what they are and tracking their movement. Static object detection and mapping uses grid mapping, to determine the occupancy of the grid, object detection and classification, for identifying any static objects and object mapping, to map the objects around the vehicle.

Road evaluation is responsible for characterising the state of the road. The road shape estimation uses either boundary lines, that limit the drivable space, or measurements (such as the curvature, position and heading of the roads near the vehicle) to determine the shape of the road (Kunz et al. 2015, Urmson et al. 2008). Estimating the shape of road is useful when road geometry is unknown a priori. Lane detection and tracking enables lane following, whenever the road markings are clearly visible to the vehicle. Alternatively, surface assessment, evaluates the state of the road surface (e.g whether the surface is icy) in order to determine safe driving practices, such as safe vehicle speeds (Montemerlo et al. 2006).

There are two different levels of localisation, road-level and lane-level. The distinc-

tion between the two is their scale (Macro-scale or Meso and Micro-scale respectively) and consequently their level of accuracy. Road-level localisation identifies the position of the vehicle in the network, which is used by the Route and Mission planner, as explained in section 4.2.3, requiring positioning accuracy of at least $5m$ (Meng et al. 2018). Alternatively, lane-level localisation identifies the lane within which the vehicle moves as well as its position within the lane. This level of localisation is used by the behavioural and trajectory generator, as discussed in section 4.2.3 and requires positioning accuracy of at least $0.5m$ (Meng et al. 2018).

The self-perception function monitors the state and motion of the vehicle as it moves within its surrounding. It interprets the state of the vehicle itself, the state of the sensors and actuators as well as fuel or battery levels. The state of the vehicle refers to its position, orientation, velocity and acceleration in all three orthogonal directions. This can be determined using inertial sensors that monitor the vehicle's motion as well as using the state of the actuators.

The vehicular communication function interprets the data collected through the communication links. It interprets the information collected from the roadside infrastructure (i.e. identifies and understands traffic signs and signals), V2V and V2I communication, and system operator e.g. information about the road network, traffic flow and any non-recurrent disruptions (e.g. accidents, road closure). Finally, weather data can be used to determine the reliability of on-board sensors under the specific weather conditions.

Communication information is used by the comprehension component when developing the environment model. Vehicular communication processes information from other agents, but is also expected to report the state and intentions of the vehicle to its surrounding agents. Therefore, it needs feedback from the environment model and the trajectory generator. This is achieved through the bidirectional link between vehicular communications and environment model and unidirectional link from the trajectory generator. The specific functionality allows for the communication of aspects of the environment that the vehicle considers important or even dangerous, to other agents.

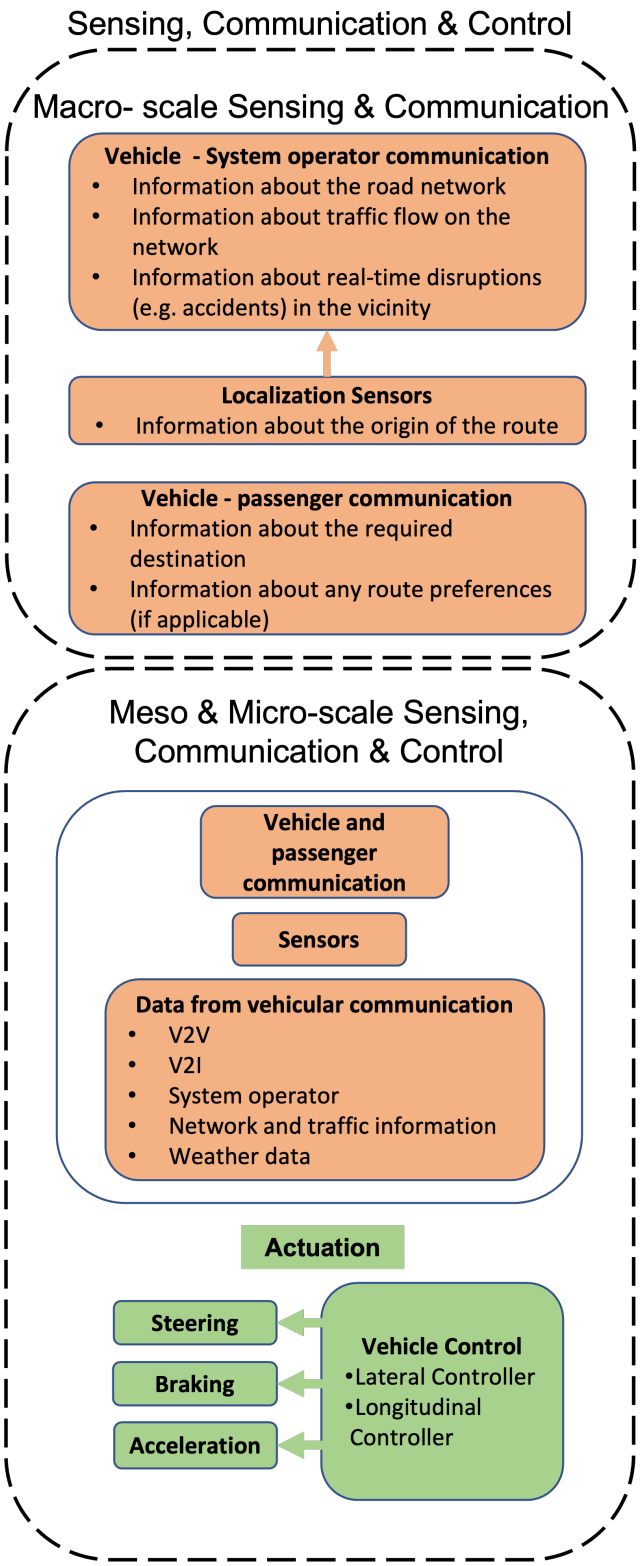


Figure 23: The vehicle component of the functional system architecture.

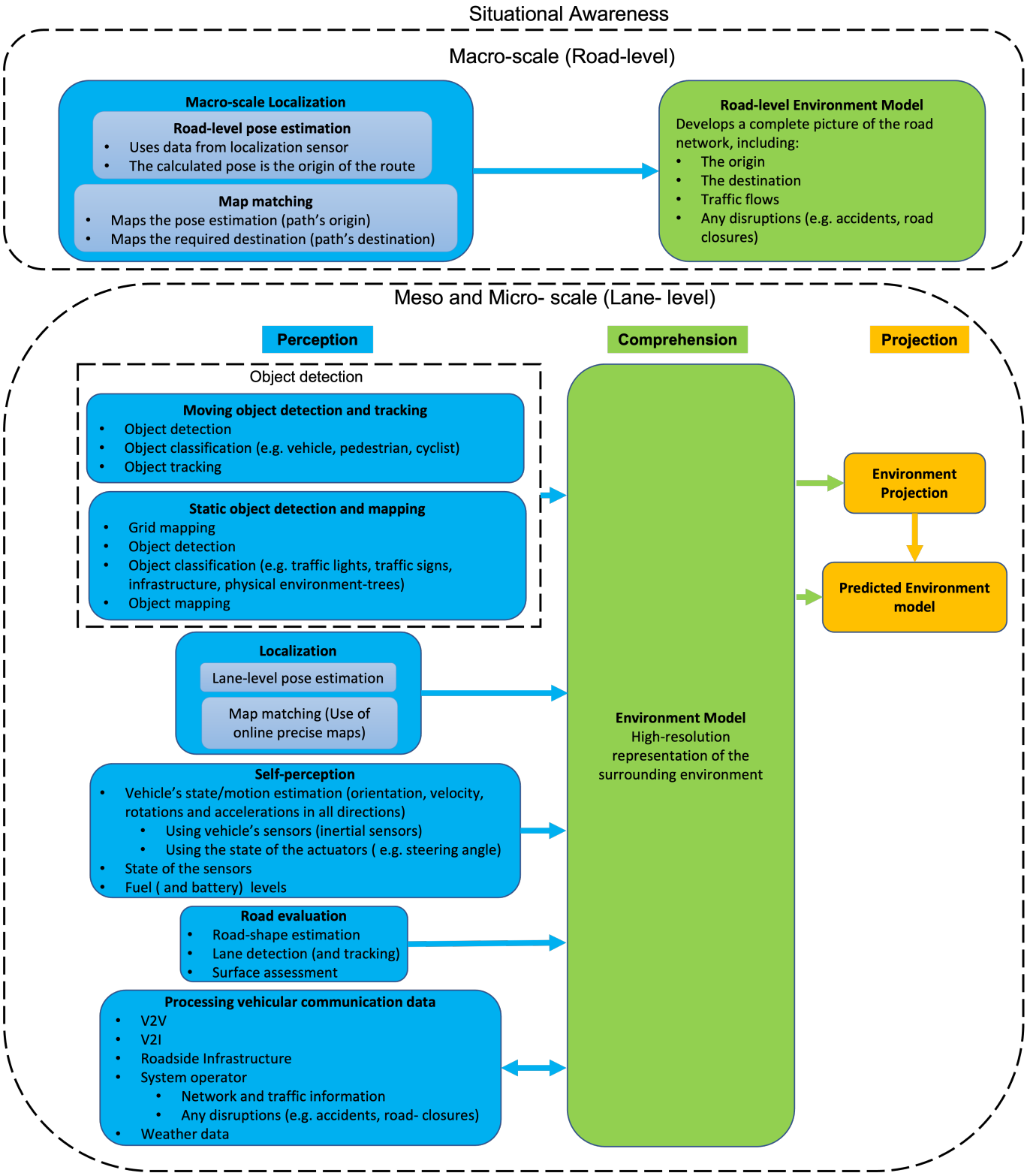


Figure 24: The situational awareness component of the functional system architecture.

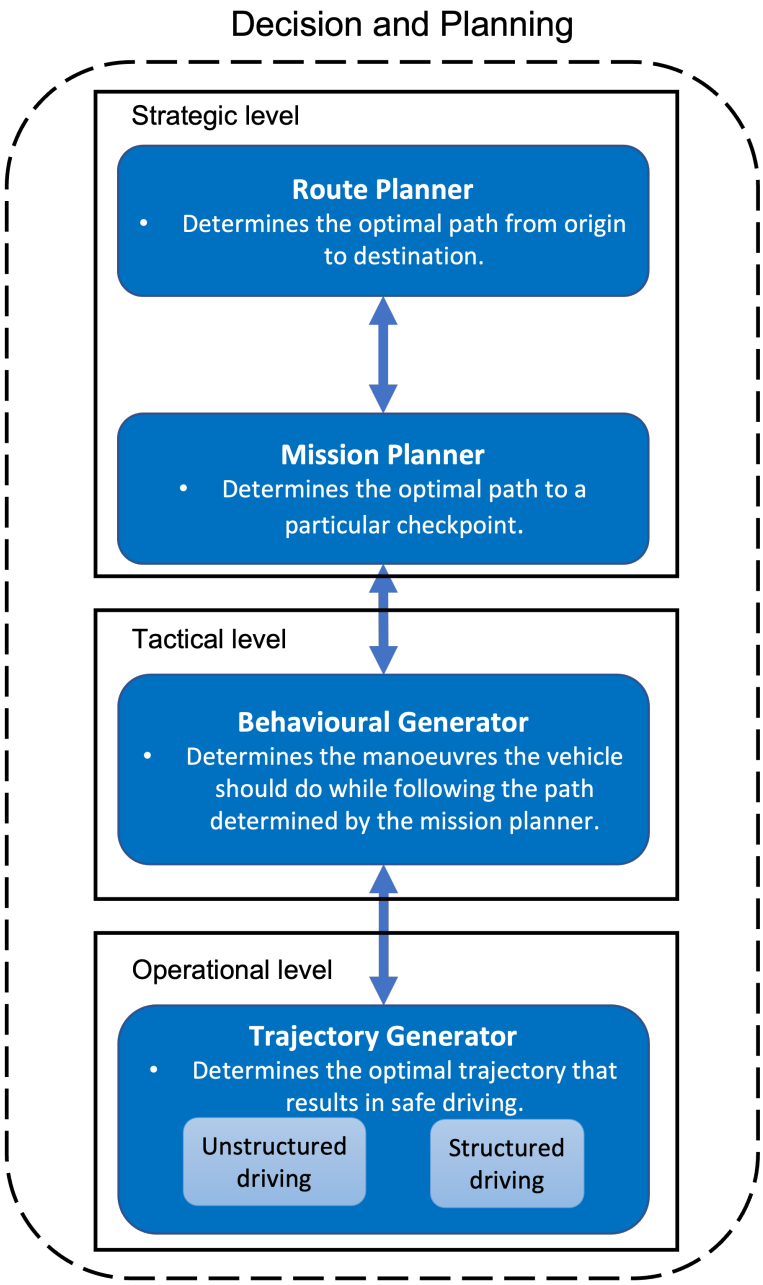


Figure 25: The decision and planning component of the functional system architecture.

The comprehension component collects the information from the five functions of perception and fuses the data together to develop a model of the surrounding. It also receives information from the trajectory generator to keep track of the vehicle's motion within the developed environment model. This model is communicated to the environment projec-

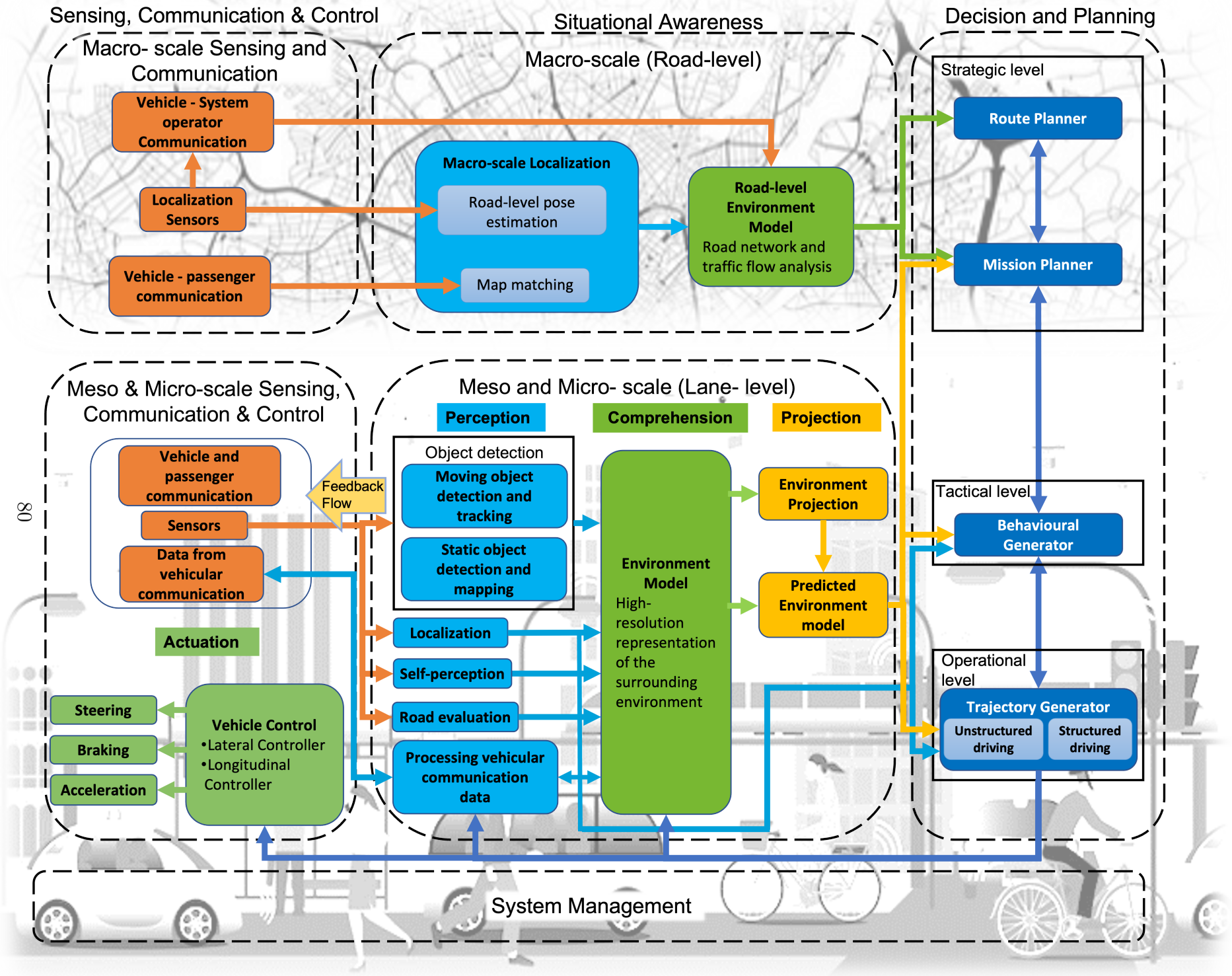


Figure 26: The proposed functional system architecture.

tion function. Here, predictions of the movement of surrounding agents in the near future are made. Finally, the predicted environment model fuses together the predictions from environment projection and the environment model into a predicted environment model.

4.2.3. Decision and Planning

The decision and planning component determines the path and trajectory, for the vehicle to follow. According to Sciavicco & Siciliano (2012), the path is solely a geometric description of the motion, while the trajectory further describes any dynamic values, such as position, velocity and acceleration, along that path.

This component is composed of four main functions, route and mission planner, and behavioural and trajectory generator, and implements a hierarchical framework, which is widely used (Wei et al. 2014). A hierarchical framework is used to reduce the complexity of a problem, as at each layer of the system, the task given from a higher-layer, is decomposed into sub-tasks, which are easier to manage (Wei et al. 2014). The four functions are split into three planning layers, namely, strategic, tactical and operational levels, as show in Figure 25.

At the strategical level, the route planner determines the path from origin to destination. It uses the understanding developed at the Macro-scale SA to calculate the optimum path, by considering the network and traffic flow. The mission planner breaks down the planned path of the route planner into smaller missions and determines the optimum path in-between checkpoints. When a route needs to be re-evaluated, such as when there is a road closure, the route or mission planner is invoked, depending on the scale of the disruption caused to the planned route. If there is a large disruption and a large portion of the route needs re-evaluation, the mission planner requests a recalculation of the optimum path from the route planner. Alternatively, in case of a small scale disruption (i.e if it affects an area in-between checkpoints), the mission planner is responsible for handling the situation.

At the tactical level, the behavioural generator, analyses the predicted environment model and decides on the right maneuvers required for the execution of the path calculated

by the mission planner. It makes decisions about actions such as lane-changing, behaviour of the vehicle at a junction and over-taking. The decision of the behavioural generator is then passed on to the trajectory generator, which is responsible for calculating the optimum trajectory for the vehicle to follow. The trajectory generator usually explores a variety of trajectories and identifies the one which avoids any moving or stationary objects and ensures a safe driving behaviour. It includes two sub-generators, responsible for either structured or unstructured driving respectively. The former is used in situations such as road following, while the latter is used in unstructured environments such as parking spaces (Urmson et al. 2008).

The trajectory determined by the trajectory generator is then passed on to the vehicle controller, as shown in Figure 26. The controller converts the trajectory into actuation commands for steering, braking and accelerating through the lateral and longitudinal controllers.

In some existing FSAs, (Urmson et al. 2009, Wei et al. 2013), the mission, behaviour and trajectory planner obtain different levels of detail from the SA component. As noted by Wei et al. (2014), this causes some vulnerabilities in the system since at higher levels of decision making, the planner does not have a detailed picture of the surrounding environment, while at the lower levels, the planner does not have the authority to reassess the decision made at higher levels. This means that the mission planner takes decisions while ignoring details within the environment and the trajectory planner does not have the authority to change the decision of the mission planner. In the FSA shown in Figure 26, it is proposed that the mission planner receives both levels of SA and communicates with the behavioural generator, so that it can intervene when necessary.

Finally, the system management component is responsible for supervising all the components, functions and information flow within the FSA, identifying any errors and intervening when necessary. The proposed FSA, illustrating the information flow between the components and functions is shown in Figure 26.

4.3. Transferability of functional system architecture

This section illustrates the transferability of the FSA developed in section 4.2. The tranferability of FSA to different types of cognitive agents, is used to illustrate that the developed FSA is a generic FSA that can easily be adapted to reflect the operation of any cognitive agent. The two case studies presented in this section focus on the case of a human driven vehicle and a visually impaired person (VIP) navigating within their surrounding environment using a cane. These cases reflect situations where two counterparts cooperate for the operation of the system. In the case of a human driven vehicle, there is the cooperation between the human agent and vehicle, while for the navigation of a VIP there is the cooperation between the human agent and the cane.

4.3.1. Human driven vehicles

The FSA presented in section 4.2 is adapted to represent the functionality in a human-driven vehicle, equivalent to automation level 0. The FSA in this section takes into consideration the cooperation between the human driver and the non-automated vehicle. The FSA of such a system is shown in Figure 27.

The FSA of a human driven vehicle is split between the driver and the vehicle. The majority of functions are executed by the driver, while the vehicle provides additional information through onboard sensor (if available) and the actuators for the driver to use. An interesting addition within the FSA presented here is driver characteristics, which consider the knowledge and skills of the driver (Liu & Ozguner 2007). The knowledge considers the familiarity of the driver with traffic rules, while the skills refer to the ability to drive (Liu & Ozguner 2007).

An additional aspect of knowledge that can impact the operation of a human driver, is their familiarity with the specific environment. A driver operating in a familiar environment is likely to have a higher level of confidence and hence response within that environment, compared to a driver operating in the same but unfamiliar environment. Similarly when considering the skills of the driver, a driver that has been exposed to similar environments, or situations in the past can operate differently to a driver that has not.

The impact of driver characteristics can influence all the levels of decision and planning. For example a driver who is familiar with a particular route can plan at the strategic level taking into consideration network and traffic information obtained through past experiences. Alternatively, a very experienced driver, with confidence in their skills is likely to have a different behavioural plan and generated trajectory compared to a less experienced driver.

In addition to the decision and planning component, driver characteristics might also influence the sensing, communication & control, as well as SA components. During the sensing, communication & control phase, an experienced driver, familiar with the surrounding environment might look for specific information through their sensors and try obtaining specific data from the external information received. This is because they might be aware of elements of the surrounding that they need to pay particular attention to. Similarly, in the SA component, the same driver can focus on more targeted elements of their developed environment model.

The driver characteristics is hence an element that can influence all of the three main components of the human driver FSA, as shown in Figure 27.

4.3.2. Visually impaired people

The FSA of VIPs considers the navigation of a VIP using a cane or even a smart cane. Depending on the reliance of the VIP on the smart cane, the FSA could be adapted to represent the ratio of functions executed by the VIP in comparison to the smart cane. This ratio between these counterparts will result in a combination between the AV and human driver FSAs presented in sections 4.2 and 4.3.1, respectively. For the development of VIP FSA, it is assumed that the VIP is using a smart cane that can help with navigation. In the case of a VIP, the distinction between functions executed solely by a VIP is not as clear as the split between a vehicle and a driver in the case of a human driven vehicle.

The split between the VIP and the smart cane depends on the VIP characteristics. The functions that the VIP is willing to execute, is dependent on their confidence, environment familiarity, environment factors and the visual impairment classification. An integration

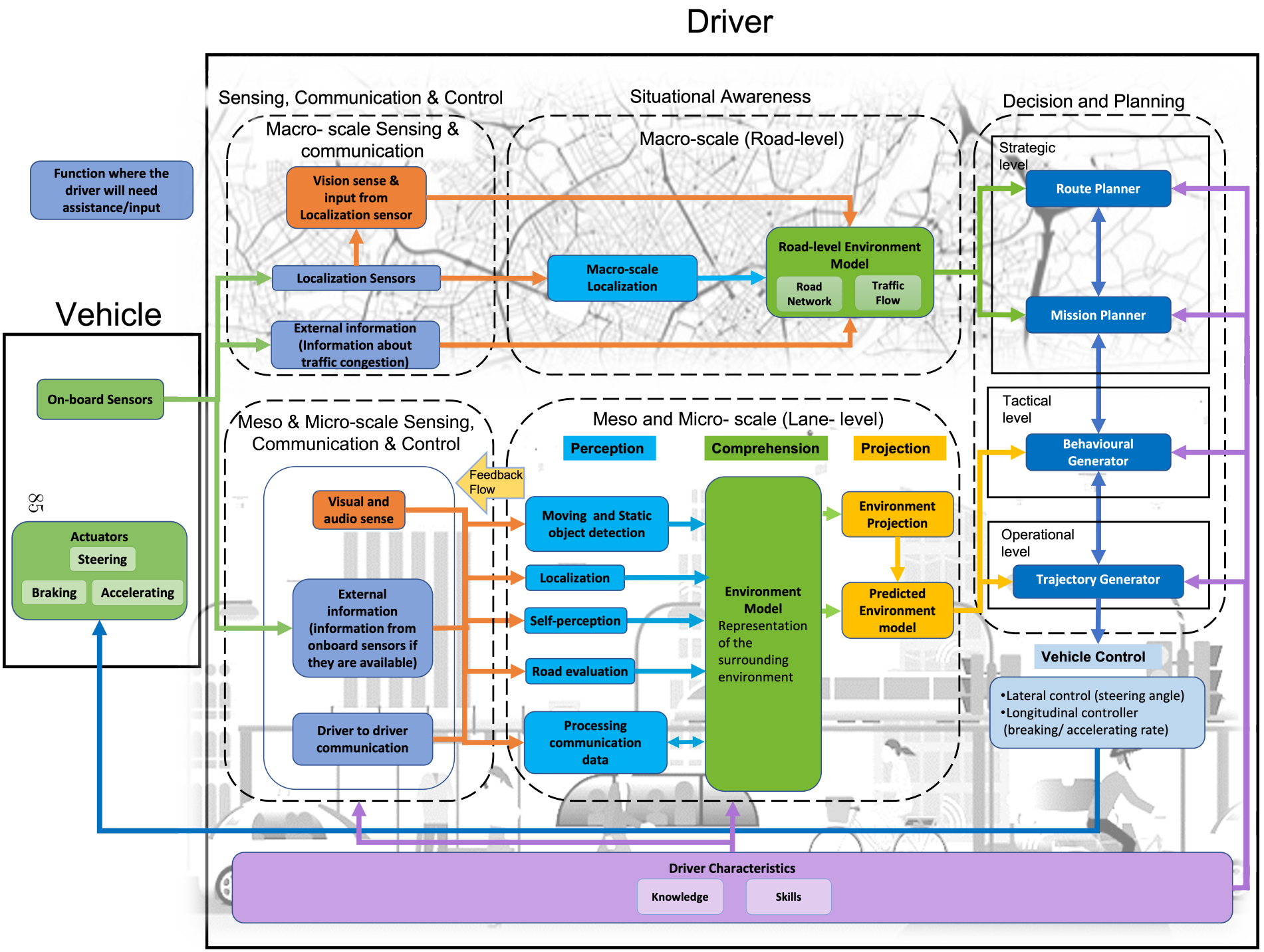


Figure 27: The functional system architecture for human driven vehicles at automation level 0.

of these factors will determine their reliance on the cane. Confidence refers to the user's experience and skills in walking within an environment using a cane; a confident user is likely to rely less on the cane while navigating. A user operating in a familiar environment is also likely to rely less on the cane. Additionally, environment factors such as how crowded the surrounding environment is are also likely to impact the user's reliance on the cane. Lastly, the classification of visual impairment (WHO 2023) can also have an impact on the user's reliance on the cane.

The VIP FSA is illustrated in Figure 28. The functions presented in purple, are those performed by the VIP, while those in grey, namely comprehension, projection, behavioural and trajectory generator, can be performed by either a smart cane, the VIP or a combination of the two. The execution of these depends on the VIP characteristics and reliance on the cane. The VIP might be able to carry out the comprehension and projection component, using information received from the device through the perception element. Alternatively, the VIP can use the predicted environment model together with the mission plan to generate the behavioural and trajectory plan. In this way the cane provides a provisional trajectory for the VIP to follow and it's up to the VIP's discretion as to whether to follow it.

The proposed FSA for VIPs presented in this section was implemented during a project focusing on VIP navigation and way finding. This illustrates the applicability of the proposed FSA for the navigation of a VIP as well as the transferability of the FSA developed in this chapter.

The proposed FSA includes a number of functions some of which require position information for their operation. In order to develop a positioning system able to support the operation of a cognitive agent, it is important to identify the functions that require position information, the LBFs. Having developed the FSA and illustrated its transferability across different agents, the following section selects the LBFs.

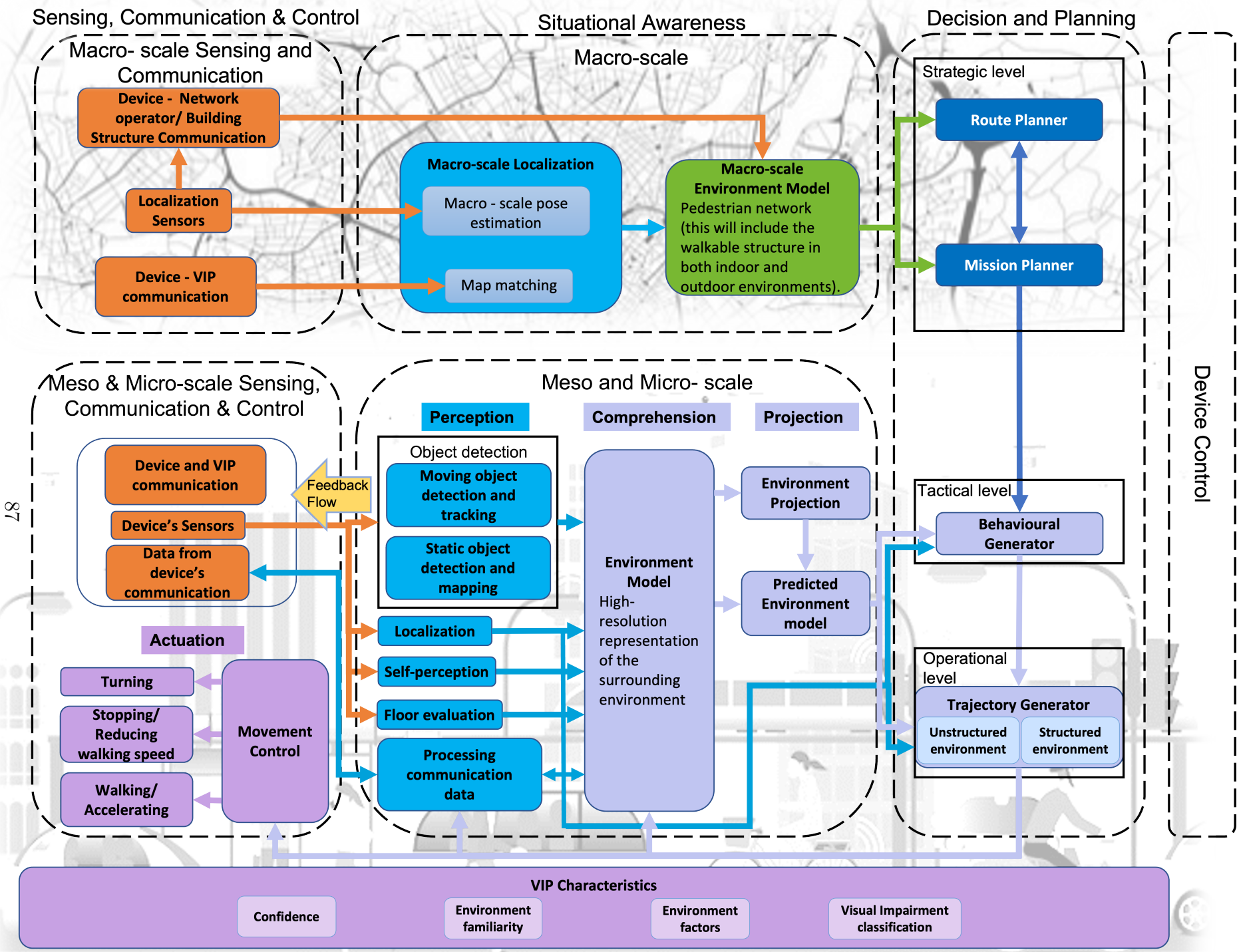


Figure 28: The functional system architecture for visually impaired person navigating using a smart cane.

4.4. Location based functions within the functional system architecture

The developed FSA in section 4.2 includes some functions that require position information in order to execute their tasks. These functions are known as LBF of the FSA. LBF might require position information either directly or indirectly. For the identification of LBFs, the functionality and required information of every function in the FSA is considered. The list of identified LBF is in Figure 29, while a visual representation of the LBF within the FSA is shown in Figure 30. The remainder of this section looks into the three main components of the developed AV FSA and selects the functions that are considered LBFs.

4.4.1. Sensing, Communication and Control

The sensing, communication & control component includes LBF within its sensing & communication as well as in the actuation elements.

The sensors of the vehicle collect data from the surrounding, which is processed by the SA element. The data collected from the sensors represent information relative to the sensor's state (i.e. position and orientation). An example are the data obtained through a LIDAR, which when processed, result in point cloud of the surrounding. However, it is important to remember that the position of each point is determined based on the distance between the point and the LIDAR itself (Pendleton et al. 2017). Knowing the exact position of the LIDAR helps in determining the exact position of the different objects that have been observed. The same principle applies for the use of RADAR, which has the capability of providing relative distance and velocity measurements. The measurements obtained are relative to the position of the sensor. Knowing the position and orientation of the signal transmission and reception point, are essential in mapping the detected elements in the surrounding. Positioning information of sensors is hence required to develop a good understanding of the surrounding as well as to localise the vehicle itself within that surrounding. An accurate mapping of the surrounding features is essential for the safe navigation of the vehicle in the network.

Vehicular communication is also considered a LBF. An important requirement of wire-

less communication is synchronisation (GOV 2018), which is achieved by the provision of accurate time. As a result wireless communication requires position information, which makes vehicular communication a LBF.

Finally, within the actuation element, the lateral and longitudinal controllers are also classified as LBF. As noted, earlier, the objective of the controllers is to convert the decision of the trajectory planner into actuator commands. The controllers should hence be aware of the precise position, velocity, acceleration and orientation of the vehicle in real time. This information allows the controllers to determine the relevant actions required to achieve the planned trajectory.

4.4.2. Situational awareness

The SA component includes LBF within object detection, localisation, self perception, vehicular communication, both scales of environment model, environment projection and predicted environment model.

Within the function of static object detection and mapping, there is the element of mapping. In order for the vehicle to map the objects within its immediate environment, it should be able to localize itself and more specifically to determine the position of the sensors which observe the environment. When referring to the processes of mapping in the field of robotics, there is the assumption that the position of the sensor is known (Stachniss 2014*a*). Hence, for the accurate mapping of static objects, the position and attitude of the sensor should be known with high accuracy at the point of data capture.

Similarly, for the detection and tracking of moving objects, there needs to be information about the position, velocity, acceleration and attitude of the sensor at the point of information capture. For example knowing the vehicle's (or more precisely the sensor's) velocity and determining the relative velocity between the vehicle and a moving agent (using a RADAR), allows for agent tracking.

The localisation function is responsible for localising the vehicle on the map, which can be expressed through a map-matching process (Quddus et al. 2007). This function is the one that provides the vehicle's position information to the rest of the functions within

the FSA. The vehicle's orientation, position and any of its derivatives such as velocity and acceleration should hence be known with very high accuracy within this function. Likewise, macro-scale localisation requires a different accuracy level of position information, to provide road-level pose estimation, that is used in the strategic level of decision making.

The self-perception function requires position information in order to monitor the state and motion of the vehicle, while vehicular communication function communicates the vehicle's location to surrounding agents, also making this function a LBF.

For the development of an accurate environment model, data synchronisation is an essential aspect, emphasising the requirement for PNT information. In addiction, the vehicle should be aware of where it is on the map as well as its orientation in order to map the surrounding objects as part of the environment model. Alternatively, for the environment projection module, additional information such as velocity and acceleration is also required. This is to realise where the vehicle is and where it is heading to predict the movement of the agents on its path. Finally, the predicted environment model requires information about the position, velocity, acceleration and orientation of the vehicle since it is the function that combines the current and future models together.

4.4.3. Decision and planning

The decision and planning component is heavily dependent on the position of the vehicle and all of its functions are LBF. Different levels of planning require different resolutions of position information and its derivatives. It is proposed that the route and mission planner use road level localisation, while the behavioural and trajectory planner use lane level localisation.

Both the route and mission planners consider the optimum path either between the origin and destination or in-between checkpoints. This can be determined using the vehicle's position, without necessarily requiring a lane-level accuracy. The route planner uses position information at the beginning of the journey to plan the entire route and throughout the journey to track the vehicle along the route. However, the route planner does not require information about the velocity or orientation of the vehicle in real time. Alter-

natively, the trajectory generator performs object avoidance, which requires knowledge of more accurate position, in addition to velocity, acceleration and orientation of the vehicle within the lane, in real-time. This illustrates that different decision and planning components required different resolution of position information as well as different position derivatives.

4.5. Summary

In this chapter the FSA of a cognitive agent based on the operation of AV has been developed. The FSA considers the existing FSAs in the literature and provides a more coherent and exhaustive structure based on the definition of a cognitive agent and human SA. In addition to the structure of cognitive agent and human SA, the proposed FSA is also layered, reflecting the different resolutions of decision and planning seen in the literature. The layered structure reflects the different resolution of information produced/ required by different functions, allowing for a more efficient distribution of information.

The transferability of the system was then illustrated by, adapting the developed AV FSA for the operation of other cognitive systems such as a human driven vehicle and a VIP navigating with the aid of a smart cane. An interesting point to note is that with the increasing involvement of a human within the system, an element referred to as human characteristics is added. This element was not presented in the case of AV, where the system is fully autonomous with no human input. However, in the case of humans, human characteristics, such us knowledge and familiarity with the environment or situation, impacts the decision process. The developed FSA for VIP operation was used as the basis of a project considering VIP navigation and way-finding, illustrating the applicability of the developed FSA.

The final section of this chapter identified the LBFs within the AV FSA. This step is essential when designing for the operation of a specific system. The identification of LBFs, aids in understanding the RNP system requirements, which are used to decide on a suitable system to support the specific application. Although defining the RNP requirements depends on the application, the layers give an idea of the functions that have

Vehicle				
Sensing and Communication			Actuation	
Sensors	Vehicular communication	Vehicle and user communication	Vehicle control (Lateral and Horizontal controller)	Actiators (Streering, Braking, Accelerating)
Data collected from the sensors represents information with respect ot the sensor's location.	The vehicle should be able to communicate its location to the object it has established communication with.		The precise location of the vehicle should be known in real time, for the controllers to determine the right actuator commands that will result in the trajectory defined by the trajectory planner.	

Situational Awareness								
Perception						Comprehension	Projection	
Moving object detection and tracking	Static Object Detection and mapping	Road Evaluation	Self-perception	Localization	Vehicular Communication	Environment Model	Environment Projection	Predicted Environment model
For the process of object tracking the location, velocity and acceleration of the sensor obtaining the data is assumed to be known.	For the process of object mapping the location of the sensor obtaining the data is assumed to be known.		PNT information required to monitor state and motion of the vehicle	This function is responsible for determining the position of the vehicle on the map.	The location of the vehicle needs to be known in order to be reported back to the surrounding agents.	The current and future environment model contain information about the vehicle's surrounding and develop models with respect to the vehicle's position. Data synchronyzation is also essential.		

Decision and Planning				System Management
Route Planner	Mission Planner	Behavioural Planner	Trajectory Planner	
These two functions will require road-level localization		These two functions will require lane-level localization		

Figure 29: The proposed LBF within the proposed FSA.

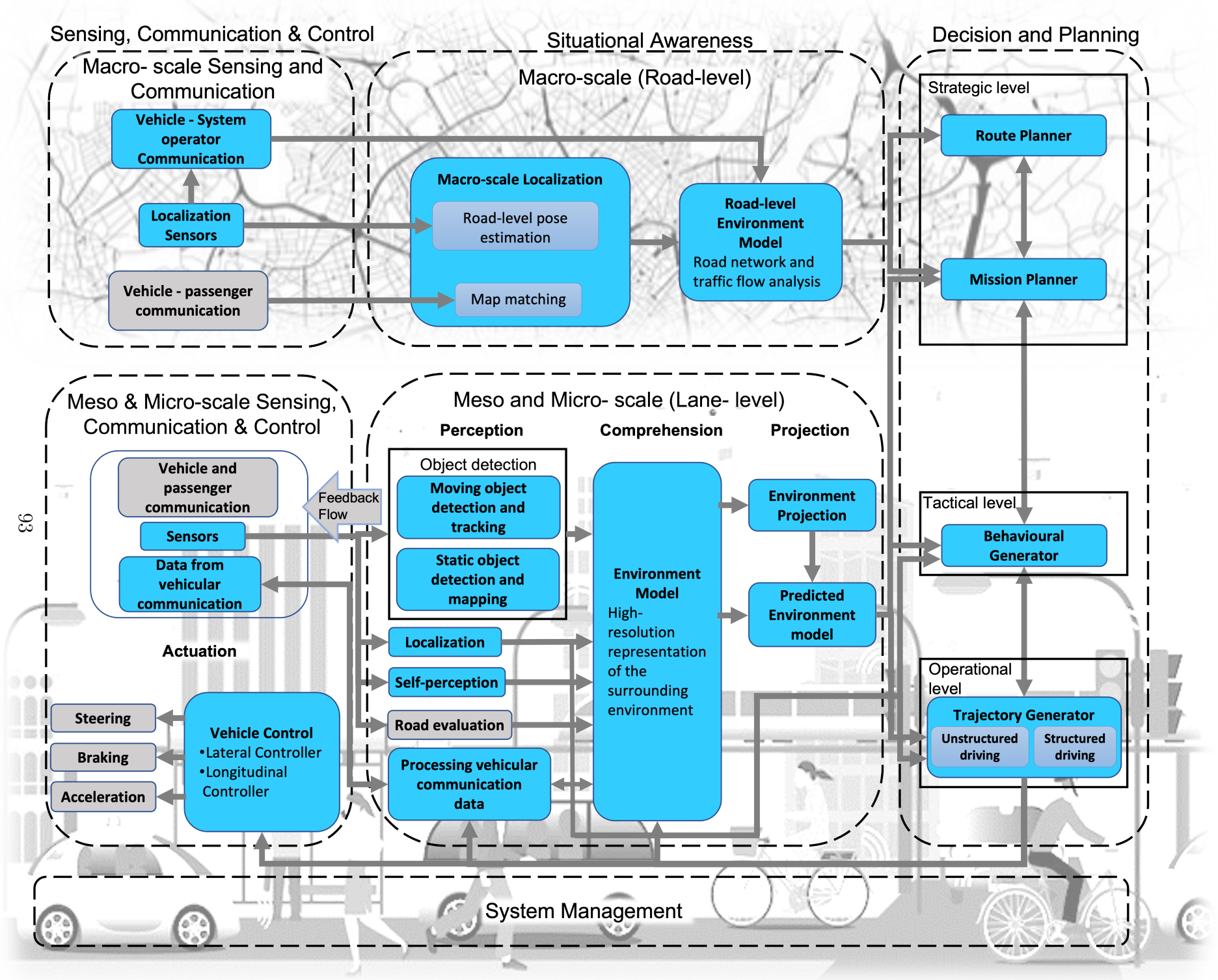


Figure 30: The location based functions within the proposed functional system architecture.

the most stringent requirements. The elements found in the Meso and Micro-scale have the most stringent requirements.

Even though the LBFs have been identified, their RNP requirements and targets have not been set, since as discussed in chapter 2, they are application dependent. Having developed the FSA and identified the LBF, the next step is the development of the physical architecture. The physical architecture is a representation of the technologies used to support the operation of the LBFs. As the focus of this thesis is the development of an IPS, the remaining chapters focus on the identification and performance evaluation of candidate positioning technologies. The next chapter reviews state-of-the-art technologies used in indoor positioning and selects candidate technologies for further investigation.

Chapter 5

State-of-the-art indoor positioning and selection of candidate methods

In chapter 4, the functional system architecture (FSA) of a cognitive agent was developed and the location based functions (LBF) were identified. As discussed in chapter 4, the LBF are functions that require position information for their operation. This chapter addresses the physical architecture that represents the technologies (or systems) that could be used to support the operation of LBF. The chapter presents an overview of indoor positioning technologies identified in the literature together with their accuracy performance. A comparison of their performance, results in the selection of high accuracy candidate systems for further investigation in the remaining of the thesis.

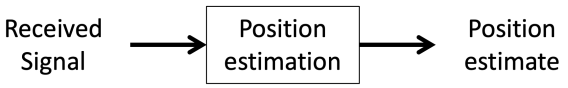
The chapter begins by explaining common methods and techniques used for position estimation. It then gives a brief description of state-of-the-art positioning systems and summarises their accuracy performance in section 5.3. The review of the literature for the selected candidate systems then follows in section 5.4, where the advantages and disadvantages of the selected systems as well as their potential for system integration are discussed.

5.1. Positioning Methods using radio waves and inertial systems

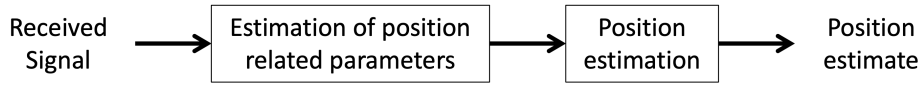
A variety of indoor positioning systems (IPS) exist which use different underlying technologies. Although there is no clear consensus in the literature on their classification, IPS are commonly characterised based on their underlying technology and position estimation process. This section provides an overview of the position estimation processes commonly used by IPSs and focuses on methods adopted by radio frequency technologies and inertial

systems.

When estimating a positioning solution there are usually two approaches, a direct and a two-step approach as shown in figure 31 (Sahinoglu et al. 2008). In the former, position is estimated directly from measured quantities. The more commonly used two-step approach, uses properties of the received signal to estimate position-related parameters which are in turn used to estimate the position.



(a) Direct approach



(b) Two step approach

Figure 31: Direct and Two step positioning approach (Sahinoglu et al. 2008, p. 63).

Position estimation approaches are classified into absolute and relative systems (Roumeliotis et al. 1999). A relative system calculates the position by integrating measurements from inertial sensors, once the system is initialised. These are systems that calculate position by tracking the motion of the dynamic agent, and hence the position estimated at every time-step is related to the last known position. Alternatively, absolute systems use measurements with respect to landmarks in the vicinity to estimate the position, without tracking the agent's motion. The classification of positioning methods and the way they relate to measurement techniques is shown in figure 32. As shown in figure 32, absolute positioning refers to geometric, scene analysis and proximity detection methods (Liu et al. 2007, Yassin et al. 2016), while relative positioning mainly refers to the method of dead reckoning (DR).

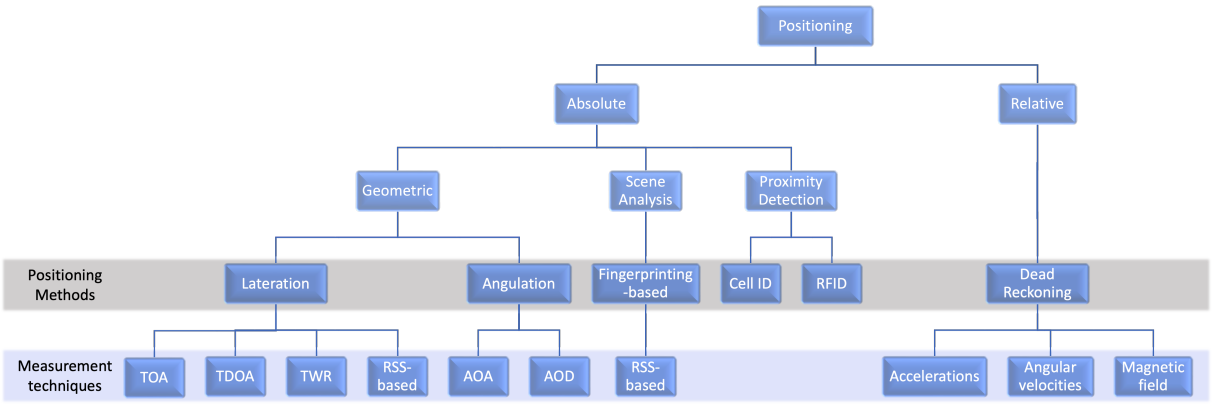


Figure 32: Classification of positioning methods.

The position-related parameters refer to the measurement techniques shown in figure 32. The parameters calculated by the measurement techniques can be classified into range, angle and signal strength measurements of the transmitted signal between an anchor node (AN) and the unlocated device (UD). These parameters are used by the positioning method to calculate the UD's position. The absolute positioning methods are split into three groups of geometric, scene analysis and proximity methods and use different parameters to calculate the position.

5.1.1. Geometric methods

Geometric methods, use geometric properties of triangles to estimate the position of UD and are split into lateration and angulation. The former, calculates the position of the UD using range measurements between the UD and multiple ANs in the vicinity. Alternatively, angulation uses angle measurements between the UD and ANs to calculate a position estimate.

Lateration methods use measurement techniques like time of arrival (TOA), two way ranging (TWR), time difference of arrival (TDOA) and received signal strength (RSS) to

estimate range measurements between UD and ANs. These techniques are either time based or signal strength based. Time-based techniques use the time-of-flight (ToF) information and signal propagation speed to estimate the range measurement. Alternatively signal strength based techniques use signal strength attenuation to estimate the range measurement.

TOA calculates the distance between AN and UD using ToF information and signal propagation speed. TOA-based systems use one-way propagation to estimate the ToF, where the signal is time-stamped at the transmitter and the receiver compares the time of reception with the transmitted time-stamp. This technique requires the transmitter and receiver clocks to be synchronised in order to avoid range errors due to clock asynchronisation.

TWR, also known as round-trip time of flight (RTOF) is a similar approach to TOA that eliminates the issue of clock synchronisation between the transmitter and the receiver (Che et al. 2023). In TWR the transmitter sends a signal at time T_{TS}, receiver receives the signal at time T_{RR}. The difference between T_{RR} and T_{TS} is the ToF or TOA. In the case of TWR, the receiver sends a reply back to the transmitter. The reply signal is sent by the receiver at time T_{RS} and arrives at the transmitter at time T_{TR} as shown in figure 33. The ToF is then calculated as shown in equation (2), with T_d corresponding to the time delay at the receiver and T_{Loop} is the time taken for the loop to be completed.

$$ToF_{TWR} = \frac{T_{Loop} - T_d}{2} = \frac{T_{TR} - T_{TS} - (T_{RS} - T_{RR})}{2} \tag{2}$$

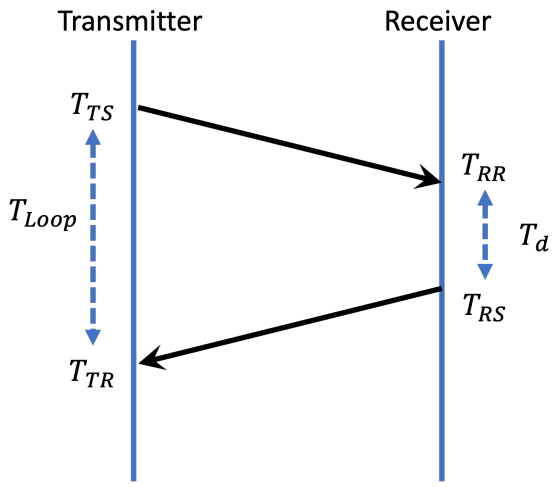


Figure 33: TWR range estimation.

In both TOA and TWR, the estimated ToF is used together with signal speed to estimate the range between AN and UD. An alternative approach for estimating range measurements is the use of RSS parameter of a signal. The distance travelled by the signal and hence the range between the transmitter and receiver is related to the attenuation of the signal (Che et al. 2023). Path loss models are used to calculate the distance between AN and UD based on RSS measurement. The complexity of the surrounding environment can impact the performance of the path loss model, resulting in erroneous RSS-based range measurements.

Once range measurements are calculated using the techniques above, the positioning method of lateration is used as shown in figure 34. The UD is on the surface of a sphere centered about the AN with radius equal to the range measurement. Each range measurement is related to the position of the UD using equation (3). For the process of position estimation, at least four range measurements need to be available. When more measurements are available, optimisation methods such as least-squares can be used to estimate the position.

$$d_i = \sqrt{(x_i - x)^2 + (y_i - y)^2 + (z_i - z)^2} \tag{3}$$

where d_i is the distance between UD and AN i, (x_i, y_i, z_i) are the coordinates of AN i and (x, y, z) are the coordinates of the UD.

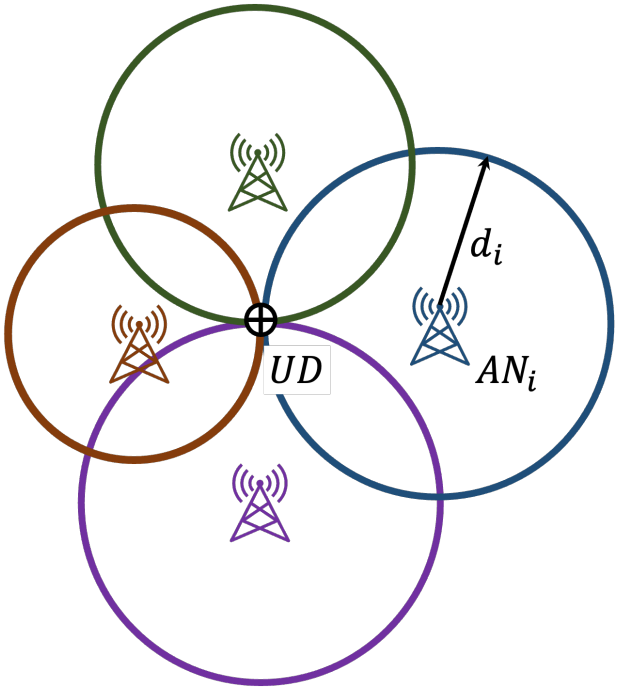


Figure 34: Using lateration to calculate the position of UD.

As opposed to TOA, TWR and RSS-based ranging techniques, TDOA uses the difference between range measurements from different ANs rather than absolute ranges. TDOA corresponds to the difference in time at which the signal arrives at different AN. As a result, TDOA is equivalent to a constant range difference between two ANs defining a hyperboloid at which the UD should lie. When at least two TDOA measurement are present, the position of UD can be calculated as shown in figure 35, which maintains a constant range difference $d_{i,j}^{TDOA}$ between the two AN. Equation (4) represents the range difference $d_{i,j}^{TDOA}$ between AN i and j, with (x_i, y_i, z_i) being the coordinates of AN i, (x_j, y_j, z_j) the coordinates of AN j and (x, y, z) the coordinates of the UD.

$$\begin{aligned} d_{i,j}^{TDOA} = &\sqrt{(x_i - x)^2 + (y_i - y)^2 + (z_i - z)^2} \\ &- \sqrt{(x_j - x)^2 + (y_j - y)^2 + (z_j - z)^2} \end{aligned} \tag{4}$$

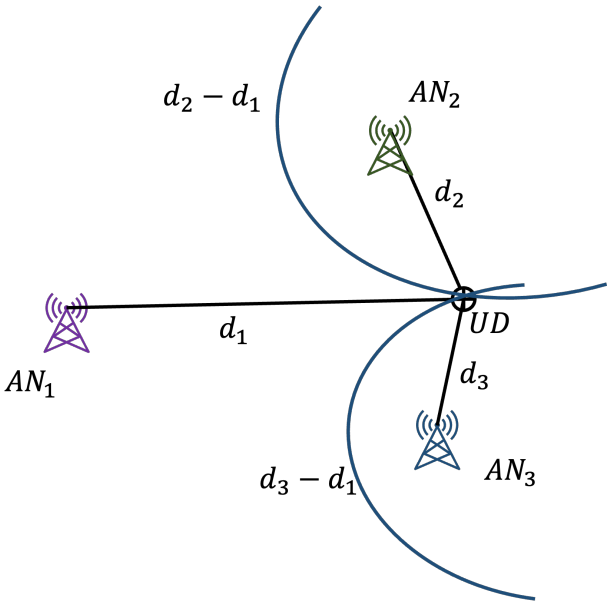


Figure 35: TDOA positioning method.

In contrast to lateration that uses range measurements for position estimation, angulation methods use angle measurements. Two commonly used angulation techniques are the angle-of-arrival (AOA) and angle-of-departure (AOD). Angulation methods calculate the signal's propagation direction which is then used to calculate the position of UD. UD position is calculated as the intersection of at least two direction lines, as shown in figure 36. The direction lines are defined by an angle measurement, that represents the direction of signal propagation. The angle measurement is obtained through either AOA or AOD techniques. AOA calculates the angle at which the signal is received by the receiver, while AOD calculates the angle at which the signal leaves the transmitter as shown in figure 37.

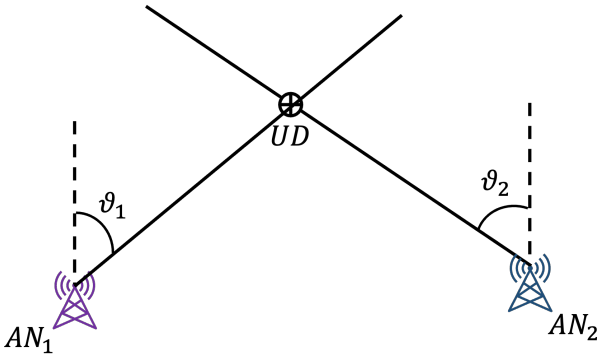


Figure 36: AOA positioning method.

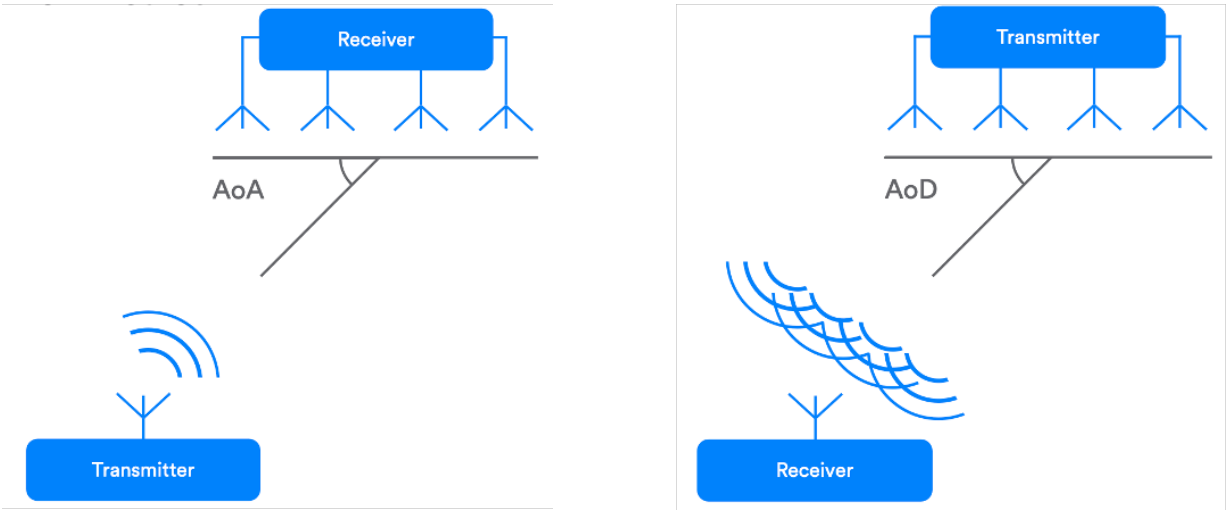


Figure 37: Angle calculation of angle of AOA and AOD (Woolley 2019, p. 10).

To determine the angle of either transmission or reception, an antenna array is used in integration with some fundamental properties of radio waves such as signal's phase measurements. The antenna array, calculates the phase difference between the antennae which relates to the angle at which the signal is received as shown in figure 38. The angle at which the signal arrives at the antenna array is calculated using equation (5), where ψ is the phase difference, d is the distance between adjacent antennae on the antenna array and λ is the signal's wavelength.

$$\vartheta = \arccos\left(\frac{\psi\lambda}{2\pi d}\right) \tag{5}$$

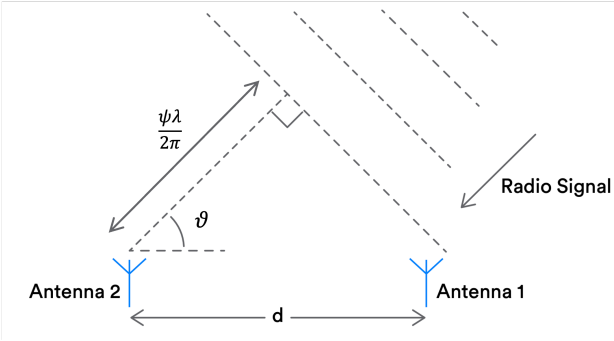


Figure 38: AOA angle calculation.

A variety of antenna array designs and layouts exist and their characteristics should be known before deciding on the appropriate type of antenna array to use. As an example, the linear array is used to calculate an angle in a single direction, while the more complex designs can calculate angles in two directions. The angle measurements calculated by the antenna arrays are used to calculate UD position.

5.1.2. Scene Analysis

The scene analysis method commonly used with radio frequency (RF) systems is fingerprinting, which positions UD in the environment, by analysing some parameters of the received signals and comparing them to pre-determined 'fingerprints' of that parameter in the environment. Fingerprinting is a two-stage method, with offline and online stages. In the offline stage the signal parameter, usually the RSS, is measured at several locations in the environment, for each detected AN. These measured quantities are stored to create a characterisation "map" called a radio map in the case of RF signals. The characterisation map is a collection of the fingerprints of the signal quantities in preparation for the online stage of the method. During the online stage, measurements of the signal quantity used to develop the characterisation map are recorded. The measurements are compared to the fingerprints of the characterisation map to estimate UD position.

A number of fingerprinting methods exist in the literature that use pattern recognition techniques. Some of these methods are probabilistic, k-nearest-neighbour (kNN), neural networks, support vector machines (SVM) and smallest M-vertex polygon (SMP), (Liu et al. 2007). Positioning methods based on RSS measurements are either lateration or fingerprinting and according to Huang et al. (2023), RSS fingerprinting is more accurate than RSS lateration.

5.1.3. Proximity

Proximity methods provide a more symbolic positioning approach, usually using a grid of antennas such as mobile cellular networks (Liu et al. 2007). This method uses the known position of ANs (or antennas) and their associated proximity area, where the proximity area is the grid area allocated to each AN. Once UD is detected by an AN, the location of the UD is considered to be within the proximity area of the detector AN (Gu et al. 2009). In the case of a mobile cellular network, cell identification (Cell-ID) can be used, where the mobile device is located within the cell of the antenna it receives a signal from (Che et al. 2023). The accuracy of such a method depends on the density of the networks. A dense network is characterised by a small distance between antennas and a small proximity area. The smaller the proximity area the higher the levels of accuracy achieved.

5.1.4. Dead Reckoning

A commonly used relative positioning method, that relies solely on measurements that track the motion of the agent is dead reckoning (DR). DR uses the distance traveled and change in direction to estimate the new position. A prerequisite for the operation of a DR system is the calculation of its initial position and direction. Once the system is initialised, an integration of the recorded changes in direction and distance travelled, results in the estimation of the new position as illustrated in Figure 39 (Groves 2013).

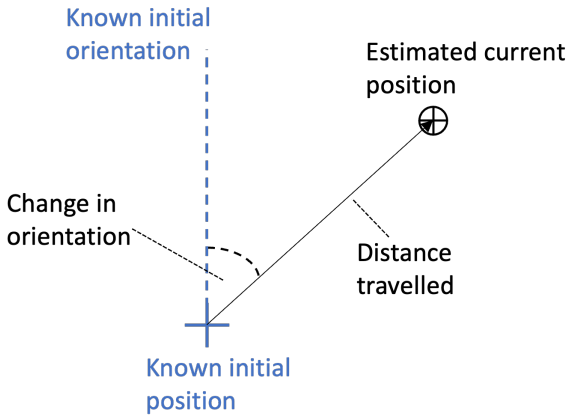


Figure 39: Principle of dead reckoning.

5.2. Signal propagation vulnerabilities

Time-based ranging techniques, are susceptible to errors cause by signal propagation. Figure 40 represents the three main propagation scenarios caused by the surrounding environment. Line-of-sight (LOS) path occurs when the signal is able to travel in a direct path (DP) between the transmitter and receiver, without any obstacles in between. A signal might reach the receiver through a non-DP, either through an obstacle or a reflected path. The former involves the propagation of the signal through different mediums, resulting in ToF measurements larger than the truth. Similarly in the case of the signal being reflected, the signal travels a longer distance before reaching the receiver, resulting again in a larger ToF corresponding to the reflected path. Both of these non-DP are know as Non-line-of-sight (NLOS) propagation and in both cases the measured range between the transmitter and the receiver are larger than the truth.

Another phenomenon related to signal propagation is multipath. Multipath occurs by signals arriving at the receiver from multiple paths (LOS and NLOS). Multipath fading occurs when signals arriving from different paths interfere in a constructive or destructive way. When a narrowband system is used, it can be hard to resolve the interference of signals arriving from different paths and identify the LOS signal. In complex environments, like indoor environments, multipath and NLOS propagation are very common phenomena due

to the complex nature of the environment.

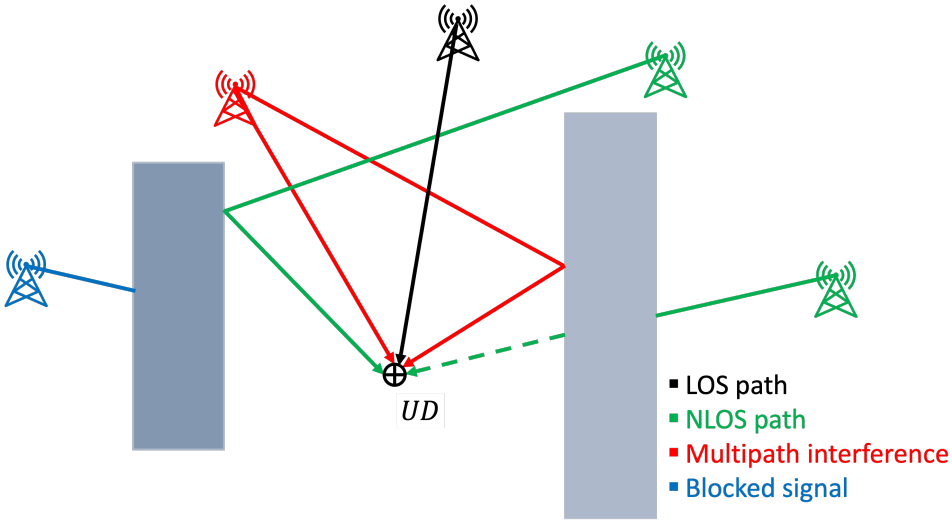


Figure 40: Signal propagation interference due to the surrounding environment.

In addition to signal propagation errors, the estimated position can further be affected by the geometry of AN around UD. The concept of signal geometry was developed in the field of GNSS positioning and is represented by a parameter known as the dilution of precision (DOP). The concept of DOP is introduced below, to illustrate how the geometry of AN/UD can affect the positioning accuracy.

5.2.1. Dilution of precision

An important factor that influences performance of GNSS, is the geometry of the satellites in view. The more spread the satellites are, within the receiver's plane of view, the better the positioning accuracy is, as illustrated in Figure 41. The positioning solution is calculated as the point of intersection of the two circles centred at each of the two satellites in view. Each of the circles is characterised by a radius equivalent to the range between the receiver and satellite. However, since the range estimates are susceptible to errors, there is an uncertainty region extending from either sides of the estimated distance. As a result the positioning solution is located within the uncertainty region formed at the intersection. Figure 41 shows that the shape of the uncertainty region varies with the geometry of the

satellites. The more spread-out the satellites are, the smaller the uncertainty region that contains the solution is and hence the better the positioning accuracy (El-Rabbany 2006).

Dilution of precision (DOP) is a dimensionless parameter representing the geometry of the satellites centered at a user antenna. The smaller the value of the DOP is, the better the geometry (El-Rabbany 2006). The different forms of DOP are:

- **Position dilution of precision(PDOP)**: represents the impact of satellite geometry on the three dimensional solution. This can include two components:
  - **Horizontal dilution of precision(HDOP)**: represents the impact of satellite geometry on the horizontal component of positioning.
  - **Vertical dilution of precision(VDOP)**: represents the impact of satellite geometry on the vertical component of positioning.
- **Time dilution of precision(TDOP)**: represents the impact of satellite geometry on the calculated time.
- **Geometric dilution of precision(GDOP)**: represents the combined effect of both PDOP and TDOP.

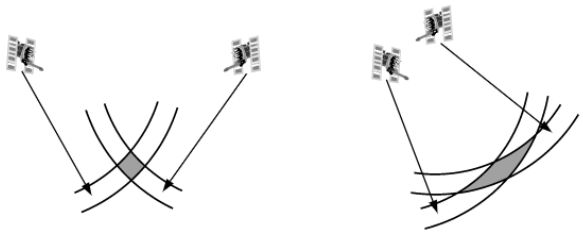


Figure 41: The impact of satellite geometry on the positioning solution (El-Rabbany 2006, p. 58).

The principle of DOP can be applied to any RF system using ranging information. AN/UD geometry has a similar impact on the positioning solution, as satellite geometry in the case of GNSS. Considering the geometric impact of the signals, is beneficial when setting up RF systems in indoor environments.

Having identified the different positioning methods, measurement techniques and RF signal vulnerabilities, the next section provides an overview of state-of-the-art systems used in indoor positioning.

5.3. State-of-the-art positioning systems

A reliable positioning system is needed to facilitate the operation of the identified LBSs. A variety of IPS and technologies exist, however there is not a general consensus for their classification. A few classifications seen in the literature classify the systems based on the technologies used, positioning method adopted, type of measurements or infrastructure requirements (Huang et al. 2023, Singh et al. 2024, Qi et al. 2024, Basiri et al. 2017, Yassin et al. 2016, Mendoza-Silva et al. 2019). In this thesis the positioning systems identified in the literature are classified in a similar way to Basiri et al. (2017), into beacon-based, on-board and device-free systems. Beacon-based systems have infrastructural requirements. They usually require the set up of anchor nodes at known locations, with which UDs interact. Alternatively, on-board systems do not require any infrastructure and are used to either observe the environment or track the motion of the agent. Device-free position systems do not require on-board sensors and location can be calculated by the infrastructure, such as CCTV camera system, and transmitted to the device if needed.

An additional classification in the literature, is the distinction between bespoke and overlaid systems (Nur et al. 2013). The former are developed for the purpose of positioning, while overlaid systems use existing infrastructure and networks (i.e signals of opportunity-SOP) for UD positioning (Nur et al. 2013). The advantages of using SOP, is the reduced set-up and infrastructure costs, however, SOP might require more complex algorithms to achieve comparable performance to bespoke systems (Liu et al. 2007). Alternatively, bespoke systems might result in a better positioning performance, as the designers have more control to develop a system with the positioning requirements in mind. However, the set-up and infrastructure costs are usually higher compared to SOP.

5.3.1. Beacon-based systems

Beacon-based systems are wireless systems that can be used for positioning in complex environments. A number of beacon-based systems make use of radio frequencies (RF), as radio waves have good propagation properties that allow for the coverage of larger areas with less hardware compared to other systems (Gu et al. 2009, Che et al. 2023). Figure 42 presents the radio spectrum used in navigation, broadcasting and telecommunications that varies from Low Frequency (LF) to Super high frequency (SHF).

Common RF systems used in the field of indoor positioning are ultra-wideband (UWB), Bluetooth low energy (BLE), WiFi and radio frequency identification (RFID) (Che et al. 2023, Huang et al. 2023, Singh et al. 2024, Qi et al. 2024, Tian et al. 2024, Naser et al. 2023). The following section presents an overview of these systems as well as Ultrasonic sensors.

5.3.1.1 Ultra Wideband

Ultra wideband (UWB) is a RF communication technology, which is also used in the field of positioning. UWB communication technology was developed in the 1990s and was approved for commercial applications in 2002, by the Federal Communication Commission (FCC). The technical aspects of UWB are described by the IEEE 802.15.4 standard. UWB operates in the 3.1GHz - 10.6GHz frequency spectrum and shares the spectrum with other RF narrowband signals used in communication (Dardari et al. 2009). The UWB signal is transmitted at low power spectral densities, in order to reduce the interference with signals occupying similar frequencies (Perakis & Gikas 2018). UWB operates in a wide bandwidth of 500 MHz, which allows for the signal to be transmitted in the form of short pulses. This is in contrast to other narrowband RF technologies usually used in the field of positioning. The pulse nature of the UWB signal, allows for the calculation of more accurate position compared to other narrowband systems, using techniques such as TOA and TWR (Che et al. 2023).

In complex environments, where multipath propagation is a usual phenomenon, narrow-

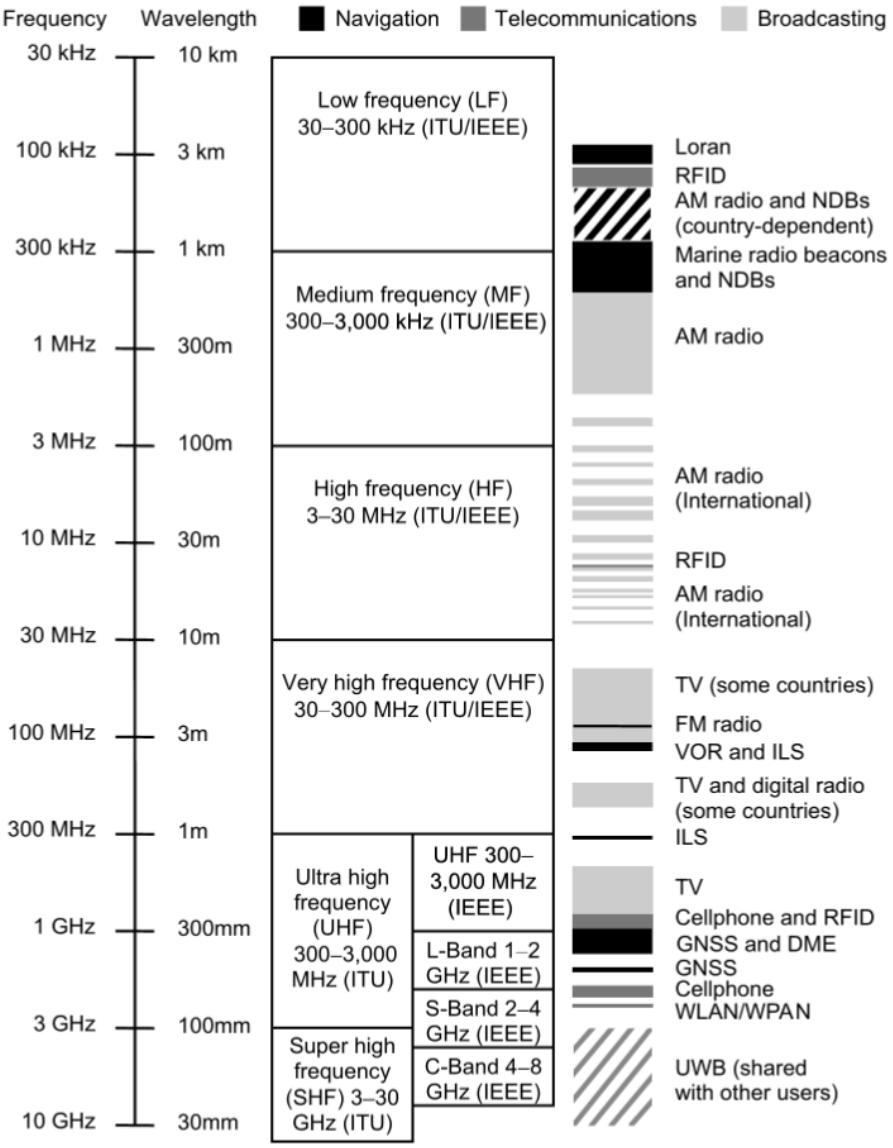
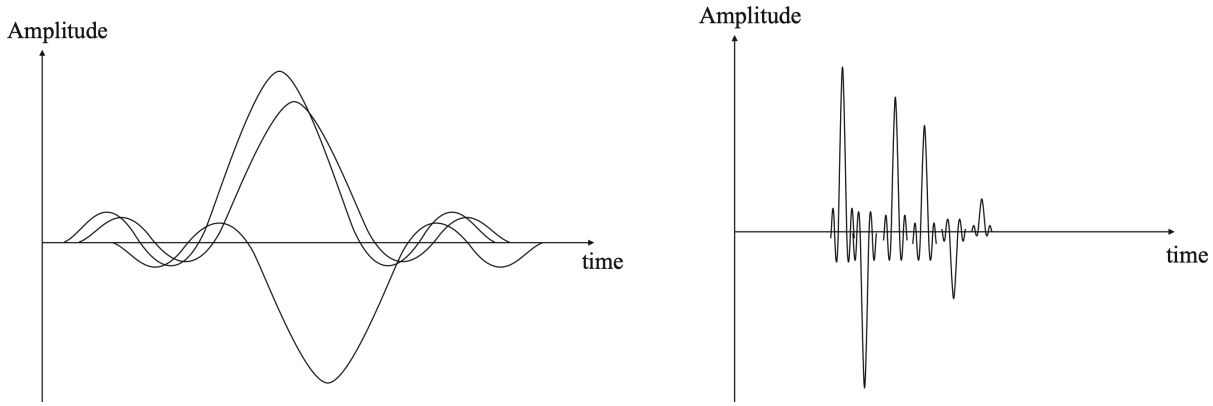


Figure 42: Radio spectrum and its used in navigation, broadcasting and telecommunications (Groves 2013, p. 278)

band systems have difficulty in resolving signals arriving from different paths. Multipath propagation can result in constructive or destructive interference of the signals, making the identification of the direct path (DP) signal a difficult task (Dardari et al. 2009). Alternatively, the pulse nature of UWB signal makes it easier to identify the DP signal, allowing for multipath resolution (Dardari et al. 2009, Elsanhoury et al. 2022, Mazhar et al. 2017, Che et al. 2023, Huang et al. 2023), as illustrated in figure 43. In addition to multipath resolution, the pulse nature of the signal allows the signal to penetrate through obstacles and reach the receiver through a NLOS path (Elsanhoury et al. 2022, Perakis & Gikas 2018, Che et al. 2023).

Even though, UWB allows for multipath resolution and NLOS signal propagation, complex environments like indoors, introduce complex geometries and varying conditions which can result in nontrivial multipath resolution (Elsanhoury et al. 2022, Perakis & Gikas 2018, Dardari et al. 2009), and the penetration through different material result in NLOS propagation errors (Perakis & Gikas 2018, Fan & Du 2022). The low transmission power, can further result in issues related to the scalability of the system, as the signal is able to propagate through shorter distances ($10m - 20m$) compared to other RF signals, depending on the complexity of the environment (Poulose et al. 2019).



(a) Narrowband signal (b) UWB signal

Figure 43: Multipath resolution for a narrowband and UWB signal (Mazhar et al. 2017, p. 4470).

Multilateration methods using techniques like TWR are usually adopted when calculating a UWB based positioning solution (Malajner et al. 2015, Elsanhoury et al. 2022). UWB results in high positioning accuracy, at the mm level (Alarifi et al. 2016). As mention earlier, its pulse nature results in UWB performing better than narrowband systems during multipath and NLOS propagation, both characteristics of complex environments.

5.3.1.2 Bluetooth low energy

Bluetooth is a wireless technology standard operating at the $2.4GHz$ frequency band, and was originally developed for the exchange of data over short distances. In addition to exchange of data, Bluetooth is also used as a positioning system. Bluetooth positioning methods used to rely mainly on RSS techniques, while recently the functionality of angulation methods have been added (Che et al. 2023, Pau et al. 2021).

RSS techniques are used in two ways, either for lateration or fingerprinting methods. In lateration methods RSS is used to estimate the ranges of the UD from surrounding ANs, which are then used to calculate the position of the UD. Alternatively, when using the method of fingerprinting, the RSS value is mapped against an existing map of fingerprints to calculate the position. According to Mendoza-Silva et al. (2019), the reported accuracy of Bluetooth RSS-based systems in the literature is between $1m$ and $3m$. This level of accuracy can be insufficient for navigation in complex environments, especially indoor environments.

Version 5.1 of Bluetooth core specification includes enhancements in the firmware that allow for direction finding capabilities (Pau et al. 2021). The enhanced version of Bluetooth introduces the functionalities of AOA and AOD with the corresponding hardware being equipped with antenna arrays, resulting in more expensive antennas. Bluetooth AOA-based positioning achieves sub-meter level accuracy, the highest accuracy of all Bluetooth-based methods (Zhuang et al. 2022).

5.3.1.3 WiFi

WiFi is a Wireless Local Area Networks (WLAN) that operates on the the IEEE 802.11 standard and in the $2.4GHz$ and $5GHz$ frequency bands (Makki et al. 2015, Che et al. 2023, Huang et al. 2023). Although WiFi networks were primarily developed for communication purposes, there is substantial research on the use of WiFi in positioning. WiFi-based positioning is an overlaid system using SOP with many existing access points (AP) that can provide good coverage (Che et al. 2023).

RSS techniques are more commonly used in WiFi systems, while TOA, TDOA and AOA are less common due to the complexity of the surrounding environment (Yassin et al. 2016). WiFi RSS-based lateration methods determine range measurements using propagation models that correlate path loss (PL) to distance travelled by the signal. Propagation models use PL coefficients to represent the type of environment the signal propagates through. The choice of PL coefficients determines the accuracy of the estimated range measurement. Due to the complexity of indoor environments, different PL coefficients may be required to represent the propagation of the signal in different directions (Khalajmehrabadi et al. 2017), introducing inaccuracies in RSS-based range estimation and consequently lateration methods.

RSS fingerprinting methods are more commonly used, as RSS measurements provide an accuracy between 5-10m in complex environments (Che et al. 2023, Basiri et al. 2017), while the performance is significantly worse when RSS ranging is used (Nur et al. 2013). Although RSS fingerprinting performs better than ranging, it is a method that requires building and maintaining an up-to-date map of RSS fingerprints, which can be challenging in indoor environments.

5.3.1.4 Radio frequency identification

Radio frequency identification (RFID) is a way of transmitting data through the use of RF. An RFID system is composed of tags, readers and communication links between them, where the RFID readers receive and read the data transmitted by the RFID tag (Huang

et al. 2023, Liu et al. 2007). RFID can operate in a passive or active mode. In the latter the reader is carried by the dynamic agent and scans for tags in the environments. Alternatively, in passive mode the tag is carried by the dynamic agent and its presence in detected by the readers in the environment (Basiri et al. 2017).

RFID systems can be used to replace the bar-code technology which is a proximity method that localises the user using the cell-id. RF-based IPS are commonly based in proximity or RSS-based lateration methods (Mendoza-Silva et al. 2019). The accuracy performance of the system varies from a centimeter to a meter level (Xia et al. 2022, Mendoza-Silva et al. 2019, Huang et al. 2023), while their detection range in passive mode is short (between 1-2m) (Liu et al. 2007, Basiri et al. 2017).

5.3.1.5 Ultrasonic sensor

An alternative type of signals to radio signals, used for positioning, are acoustic signals. Although acoustic positioning methods are widely used in underwater environments, a number of acoustic positioning systems have been developed for indoor environments. Some positioning systems using ultrasound signals in the literature are 'Active Bat', 'Sonitor', 'Cricket' and 'Dolphin' (Gu et al. 2009, Priyantha et al. 2000, Fukuju et al. 2003), which mainly use TOA and TDOA techniques. Moreover, it is common for ultrasound-based systems to be integrated with RF signals for purposes of system synchronization.

Although ultrasound-based positioning systems result in centimetre level accuracy, the signals suffer from reflections and interference. Moreover, the speed of the signal is affected by the medium of propagation and hence changes in temperature and humidity can impact the propagation speed and calculated ToF (Huang et al. 2023). An additional concern about the use of ultrasound-based systems is their usability and affordability. Although their deployment can be of relatively low cost for a single room, this might not be the case for large environments, while acoustic sensing capabilities of mobile devices are not enough to support such systems(Mendoza-Silva et al. 2019).

5.3.2. On-board systems

On-board sensors do not require the use of beacons in the environment. These systems provide a positioning solution, either by tracking the motion, or detecting landmarks in the surrounding environment that can be used for positioning a dynamic agent. The on-board sensors commonly used in the literature are inertial measurement units (IMU) and vision-based systems.

5.3.2.1 Inertial Navigation System

Inertial navigation systems (INS) are systems that provide position and navigation information using inertial measurements related to the motion of the sensor, like acceleration, angular velocity and some times measurements of the magnetic field. Sensors that collect inertial measurements are referred to as inertial measurement units (IMU).

INS employ the principle of DR, introduced in section 5.1.4, to estimate the position of the IMU sensor. IMU sensors have the advantage of being self-contained, as they measure changes in the motion of the system and do not need to interact with any infrastructure or surrounding environment. This makes inertial sensors able to operate in any location and environment without its measurements being compromised. Alternatively, a down-side of using INS is the accumulation of errors. Since such systems calculate a position estimation by integrating measurements of changes in motion over time, measurement errors accumulate over time as shown in Figure 44. Inertial sensors incorporate gyroscopes, accelerometers and some times magnetometers.

A gyroscope records angular velocities (ω), which represent changes in orientation of the inertial sensor over time. An accelerometer and a magnetometer calculate the acceleration and magnetic field experienced by the inertial sensor respectively. The integration of accelerometer and magnetometer data can be used to calculate absolute orientation. This is done using the accelerometer to indicate the direction of the gravitational force and the magnetometer to indicate the Earth's magnetic field. Inertial base sensory systems usually include tri-axis sensors, collecting measurements from three orthogonal directions. An

inertial based sensory system that uses tri-axial gyroscopes and accelerometers is known as an Inertial Measurement Unit (IMU) and is used to track changes in orientation and translational movements. Alternatively, a Magnetic, Angular Rate and Gravity (MARG) sensor incorporates tri-axial gyroscopes, accelerometers and magnetometers and can provide absolute orientation with respect to the Earth's gravitational and magnetic fields. In this thesis, magnetometer data are not considered, since electromagnetic interference is a known issue in indoor environments, that result in measurement inconsistencies (Marković et al. 2022).

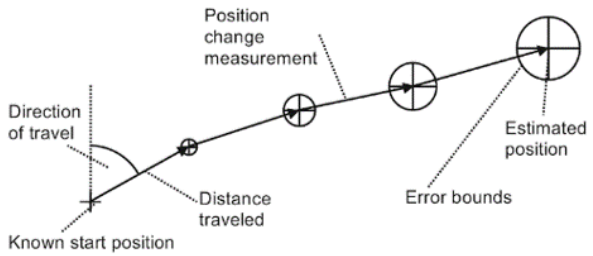


Figure 44: Error accumulating in a DR system (Groves 2013, p. 4).

An alternative way of using IMU and MARG sensor data, other than estimating the INS solution through the DR method, is their use in calculating the sensor's orientation (or attitude). An orientation filter, using data from either an IMU or a MARG sensor is the Madgwick filter (Madgwick et al. 2010). It is a filter that uses quaternions instead of Euler angles, to eliminate the issue of singularities that can arise with Euler angles. Euler angle singularities, also known as gimbal lock, can occur at specific values of roll, pitch and yaw angles, causing two of the three axis to align, reducing the degrees of freedom. Quaternion representation is an alternative way of representing orientations. A quaternion is a four-dimensional complex number (or vector), where the first element of the vector corresponds to the angle of rotation and the remaining three represent the vector of rotation. A detailed explanation of the Madgwick filter is presented in chapter 8, where empirical IMU data are used to estimate the orientation of the IMU sensor.

The performance of the INS depends on the performance of the IMU sensor used.

IMU measurements suffer from random and systematic errors in the measurements. The former considers the noise in the measurement, while systematic errors usually include bias and scale factor. The bias represents a constant shift in the measurements (El-Sheimy & Youssef 2020) and is evaluated when there is no input signal in the sensor, as the offset from zero of the recorded measurements (Lawrence 2001). Alternatively, the scale factor represents the ratio of the input value imposed on the sensor and output measurement and can vary with temperature (El-Sheimy & Youssef 2020).

The magnitude of random and systematic errors is dependent on the IMU technology used, and as a result IMU sensors are classified depending on their accuracy performance (El-Sheimy & Youssef 2020). Table 4, presents the four grades of IMU with a summary of their performance in terms of position drift rates, IMU sensors' biases and random walk as well as their approximate cost. Strategic grade IMUs are the most accurate and are commonly used in submarines, military ships and intercontinental ballistic missiles. Navigation grade are commonly used in commercial aviation and military aircrafts. Tactile grade IMUs are used in integration with alternative positioning systems in road vehicles, unmanned air vehicles (UAVs) and user guided weapons. Finally, consumer grade IMUs result in the lowest accuracy performance, and are typically integrated with an alternative positioning system in low cost navigation applications such as pedestrian navigation (El-Sheimy & Youssef 2020).

Table 4: Summary of IMUs classification and performance (El-Sheimy & Youssef 2020)

Specification	**Grade**			
	Strategic	**Navigation**	**Tactical**	**Consumer**
Positional Error (nmi/h)	0.016-0.054	0.5-2	10-20	Large variation
Gyroscope Bias (deg/h)	0.0001-0.001	0.005–0.01	0.1–10	$\geq$100
Gyroscope Random Walk ($deg/\sqrt{h}$)	-	$\leq$0.002	0.05-0.02	Several
Accelerometer Bias (mg)	0.01	0.03-0.1	1-10	$\geq$10
Cost ($)	1,000,000	70,000-100,000	2,000-50,000	$\leq$1000

nmi: nautical mile (1851 m).

5.3.2.2 Vision-based systems

Vision systems and in particular the camera is a sensor that can be installed on-board of a dynamic agent and used for positioning. Camera sensors are used for visual odometry (VO) systems and landmark-based positioning method (Huang et al. 2023). Landmark-base positioning uses a camera to detect landmarks in the surrounding, match them against geo-referenced images and locate the camera and hence the agent (Li et al. 2013, Basiri et al. 2017).

VO, like INS, tracks the motion of the dynamic agent, using a series of images captured by the on-board camera(s). The camera's motion and speed are determined using feature tracking, appearance-based methods or a combination of the two (Aqel et al. 2016). Appearance-based methods track changes in the appearance of the images and pixel information density such as displacement of brightness patterns (Aqel et al. 2016). Alternatively feature tracking methods identify and track features in the images. Feature tracking methods are more applicable for textured scenarios like complex environments, where a number of features can be tracked. Alternatively, appearance-based methods are more robust in low-textured scenarios. Even though VO is a relative process, like INS, and is characterised by accumulating drift, the rate of drift of a low-cost INS is higher than the drift of VO. However, the main issues of VO are that it is a computationally expensive method, while it is very sensitive to the quality of the image and light conditions (Aqel et al. 2016).

Another vision-based approach is Simultaneous localisation and Mapping (SLAM), which integrates localisation and mapping methods. The localisation component aims to calculate the position of the sensor observing the environment, while the mapping component maps the objects observed by the on-board sensors. These two components occur simultaneously and the agent knows its position as well as that of its surroundings' at the same time. SLAM is a method commonly used in the field of robotics (Stachniss 2014*a*). SLAM methods can be visual or LiDAR depending on the sensors used to observe the surrounding environment (Huang et al. 2023). Visual SLAM uses sensors such as cameras, while LiDAR SLAM uses a LiDAR sensor to create a point cloud of the surrounding

environment.

As localisation and mapping processes happen simultaneously, the two methods are interdependent and errors in one method can affect the other. An inadequate localisation performance will result in the development of a poor quality map and vice versa. Alternatively, these methods can benefit from one another, for example an agent operating in a known environment with well known landmarks is likely to return better mapping and localisation results. Although SLAM is a method commonly used for the navigation of autonomous agents (such as robots), it suffers from error accumulation and drift (Shunsuke et al. 2015).

5.3.3. Device-free

Device-free systems estimate UD position independently of the UD. An example of device-free system are tactile sensors that are used to detect the presence of a UD at a specific location (Basiri et al. 2017). Another example is the use of a camera network installed within the environment of interest. The cameras track features or markers that move within the environment. The stream of images (or video) recorded by the different cameras in the network are integrated together to estimate the position of the tracked UD in the environment (Huang et al. 2023). Even though such a system can result in high levels of accuracy, it is costly to set up. Moreover, as it is based on image processing, cameras need to have an unobstructed view of the UD, which can be difficult to achieve in a complex indoor environment. An additional implication of camera systems is their sensitivity to light and reflections, particularly for marker-based camera systems, where the reflective surfaces in the environment can interfere with the marker tracking process.

5.3.4. Accuracy performance summary of technologies

The currently known capabilities of different technologies used in indoor environments are addressed in the literature (Qi et al. 2024, Huang et al. 2023, Che et al. 2023, Naser et al. 2023, Xia et al. 2022, Basiri et al. 2017, Yassin et al. 2016, Mendoza-Silva et al. 2019, Liu et al. 2007, Gu et al. 2009, Mautz 2012). Table 5 is a summary table of the positioning accuracy achievable by different technologies. Evident from the literature is

that the performance of the technologies is dependent on the method and techniques used (Huang et al. 2023, Pascacio et al. 2021). Some reviews provide a typical estimate of the the accuracy of technologies (Che et al. 2023, Mendoza-Silva et al. 2019, Xia et al. 2022, Samama 2019, Basiri et al. 2017), while others provide accuracy values of specific IPS (Huang et al. 2023, Liu et al. 2007, Yassin et al. 2016, Gu et al. 2009). Samama (2019) presents a table of 41 technologies and their performance. The performance parameters were not given exact numeric values, as a system's performance depends on the set up, surrounding and technique used to calculate the positioning solution. This illustrates the difficulty in evaluating the performance of available systems and in choosing the right technology for the required application.

The most promising technologies in terms of accuracy, providing an absolute positioning solution are WiFi, Bluetooth-AOA and UWB. WiFi is a well-researched technology due to its already existing infrastructure availability, however, its accuracy is at the meter level. Bluetooth-AOA positioning, is more accurate compared to Bluetooth RSS and TOA/T-DOA methods. This is evident from the improvement in positioning accuracy, from 2m-5m with ranging and fingerprinting methods, to sub-meter level with angulation methods, as see in Table 5. Although this is the case, Bluetooth-AOA positioning is a relatively new technique, the performance and capabilities of which still need to be understood. Cominelli et al. (2019) through the use of empirical data conclude that Bluetooth's direction detection accuracy is limited to a constrained range, while although positioning errors are reduced, they are still not within a few centimeters. Moreover, Cominelli et al. (2019), note the possibility of BLE packet structure being tamper with by attackers, leading to misleading positioning solutions.

UWB is increasingly attracting interest as an indoor positioning system due to its centimeter-level positioning accuracy, robustness to multipath error, anti-interference characteristics and obstacle penetration properties (Tian et al. 2024, Singh et al. 2024, Huang et al. 2023, Santoro et al. 2023, Che et al. 2023). UWB also receives significant attention from the industry, as it is used for spatial awareness, positioning and movement tracking in smartphones and other mobile devices (Apple 2023, Samsung 2021). The potential of

UWB and the increasing interest of both research and industry in its operations, make UWB a strong candidate technology for indoor positioning system.

As the aim of this thesis is to develop a positioning system that satisfies the accuracy requirements of indoor LBSs, UWB is chosen as one of the systems to be further investigated. An additional system to investigate in this thesis is the use of an INS. Even though INS is a relative positioning system that suffers from error accumulation causing the drift of the solution, it is a system whose performance does not depend on the surrounding environmental conditions. It provides a solution in any environment and when integrated with an absolute system, such as UWB, provides a solution even when the absolute system is unavailable. A more detailed review of the literature for UWB, INS and their integration is presented in the following sections.

Table 5: Summary Table of accuracy performance of different technologies, with advantages and disadvantages obtained from Xia et al. (2022).

Tech	Accuracy	Advantages	Disadvantages	Refs
BLE	Lateration/ Fingerprinting methods: 2-5m	• Range <50m • Easy to deploy • Low power • Directional	• Weak anti-interference ability • Path loss	Huang et al. (2023), Mendoza-Silva et al. (2019), Xia et al. (2022), Liu et al. (2007), Yassin et al. (2016), Basiri et al. (2017), Cominelli et al. (2019)
	Angulation: sub-meter (85cm)			Cominelli et al. (2019)
Computer Vision	0.09m	• Strong adaptability • No need for environmental transformation	• High requirements for algorithms and computing performance • Performance affected by light • Suffer in weak feature environments	Huang et al. (2023), Mendoza-Silva et al. (2019)

Table 5 continued from previous page

Cellular	5m-hundreds of meters	• Range <300m • High speed	• High cost • High equipment requirements • High power consumption	Liu et al. (2007), Basiri et al. (2017), Xia et al. (2022), Mendoza-Silva et al. (2019)
Image	centimeter level	• Strong anti-interference ability	• NLOS is unavailable • Range <10m	Xia et al. (2022)
Infrared	centimetric level (10cm-80cm)	• Mature technology	• Vulnerable to high temperature • Complex deployment • Range <5m • High cost • High power consumption	Xia et al. (2022), Yassin et al. (2016), Basiri et al. (2017), Huang et al. (2023)
RFID	0.5m - 3m	• Low Power consumption • Strong signal penetrating power • Small size	• Range <5m • Weak anti-interference ability	Mendoza-Silva et al. (2019), Xia et al. (2022), Basiri et al. (2017), Yassin et al. (2016), Liu et al. (2007)

Table 5 continued from previous page

Sound	about 10cm	• Mature technology • High positioning accuracy	• Range <10m • High power consumption • Signal attenuation • High cost	Huang et al. (2023), Yassin et al. (2016), Basiri et al. (2017), Mendoza-Silva et al. (2019), Xia et al. (2022)
UWB	10cm - 1m	• Range High positioning accuracy • Strong anti-interference ability • High multipath resolution • High penetration capability	• Medium popularity • Medium cost • Medium power consumption	Tian et al. (2024), Santoro et al. (2023), Singh et al. (2024), Che et al. (2023), Huang et al. (2023), Yassin et al. (2016), Mendoza-Silva et al. (2019), Xia et al. (2022), Basiri et al. (2017)
WiFi	2m - 5m	• Range <100m • Medium cost • High popularity • Easy to deploy	• Medium power consumption • Weak anti-interference ability	Huang et al. (2023), Liu et al. (2007), Mendoza-Silva et al. (2019), Xia et al. (2022), Basiri et al. (2017), Yassin et al. (2016)

5.4. Candidate systems for evaluation

From table 5, there are a number of different systems that can be used to provide a positioning solution. However, the nature of complex environments, like indoors, result in challenging conditions that compromise system performance. The integration of different systems is a common practice to improve performance compared to individual systems. The systems selected for further investigation are UWB and INS employing IMUs. This section provides a further review of the two systems, explaining their advantages and disadvantages, and concludes with a review of the literature on their integration.

5.4.1. Ultra-wideband

Section 5.3.1.1, introduced the advantages of the pulse nature of the UWB signal, which include the capability of multipath resolution, ability of the signal to propagate through different materials allowing for NLOS signal reception and large time resolution resulting in accurate ranging capabilities. However, the complexity of indoor environments introduces challenges in the performance of the UWB, making UWB susceptible to unresolved multipath propagation and signal propagation delays due to NLOS signal receptions (Perakis & Gikas 2018, Fan & Du 2022, Hol et al. 2009, Zhong et al. 2018). Multipath and NLOS propagation result in a larger range measurement than LOS. There are currently on going attempts in the literature to identify and correct errors in the UWB measurements.

Perakis & Gikas (2018), used empirical data to develop calibration models used to reduce the impact of errors in the range measurements, resulting in an improvement of 86% in the accuracy of the range measurements, providing a more accurate position estimation at the cm level.

Approaches used to mitigate NLOS propagation include using signal-to-noise ratio (SNR) as an indication of NLOS propagation (Schroeder et al. 2007), residuals resulting from the range measurements in a least squares estimation (Chen 1999) and more recntly the use of machine learning (ML) models for NLOS identification (Che et al. 2023). Fan & Du (2022) used a Kalman filter to predict the distance measurement and identify NLOS range measurements using the Mahalanobis distance between the predicted and observed

range measurement.

The development of calibration models and NLOS mitigation methods, show the need for understanding the UWB range measurement errors and developing methods for their correction. Empirical data, are used in chapter 7 to analyse errors in the range measurements, and develop possible approaches for corrections that could be applied during their integration with IMU measurements.

UWB measurements are used in geometric methods for position calculation using measurements such as TOA/TWR ,TDOA, RSS and AOA. The performance of the AOA method depends on the distance between the antenna array and the tag, as the error in the positioning solution increases with increasing distances (Che et al. 2023, Reddy & Sujatha 2011). Alternatively, when using RSS measurements, the attenuation of the signal caused by complex environments can result in inadequate range estimation, introducing errors in position estimation (Che et al. 2023, Dardari et al. 2009).

Although TOA/TWR, TDOA and AOA benefit from the pulse nature of the signal, the implementation of AOA has high complexity, making it less practical to use (Che et al. 2023, Alarifi et al. 2016). TOA/TWR measurements are used in the method of multilateration, while TDOA measurements are implemented in hyperbolic methods. The difference in accuracy performance, between multilateration and hyperbolic method, is investigated in chapter 7, using empirical UWB data.

5.4.2. Inertial Navigation System

INS use IMU measurements to calculate the position and orientation of the dynamic agent and provide a navigation solution. IMU measurements include systematic and random errors in the measurements. When integrating the raw data of acceleration and angular velocities to derive the position and orientation respectively, the errors accumulate, drifting the estimate away from the truth. The long-term growth in position and orientation errors is known as random walk (Lawrence 2001). Random walk parameters related to the gyroscope and the accelerometer of the IMU, are know as angle random walk and velocity random walk respectively.

IMU acceleration and angular velocity measurements are numerically integrated over time to calculate displacement and direction, in order to calculate position using the method of DR. The displacement is calculated by a two step integration of acceleration, while orientation by one step integration of angular velocities. The two step integration for displacement estimation, in comparison to one step integration for orientation estimation, results in position errors increasing at a faster rate compared to orientation errors (Won et al. 2008). In addition, the two step integration result in the position errors increasing over time, in a quadratic form. The linear component corresponds to the drift rate of the solution, while the quadratic term represents the rate of change of drift over time. In chapter 8 it is demonstrated that the position error varies in time with polynomial order two.

IMUs are systems with high sampling rates and once initialised can calculate positioning solutions by only tracking the motion of the IMU without relying on installed infrastructure in the surrounding environment. The estimated position is characterised by short term accuracy, which however deteriorates by error accumulation in the solution over time, causing long-term instabilities in the solution. The IMUs solution drifts due to the integration of the acceleration and angular velocity measurements, which cause the accumulation of errors (Zhao & Wang 2011, Randeniya et al. 2010, Kelly & Sukhatme 2009, Won et al. 2008, 2009, Li et al. 2008). The drift of the solution is the limiting factor in using an INS (Randeniya et al. 2010). A common way of correcting the drift and eliminating the long term instability, is its integration with alternative systems that are characterised by long term stability. The integration of the INS with UWB is presented in chapter 9, where a novel approach, using UWB measurement to reduce the impact of IMU accelerations, is also presented.

5.5. Integration of candidate systems

The UWB and INS are two systems that compliment one another. When integrating systems together the method of integration can impact performance. This section provides an overview of the three types of integration, loose, tight and ultra-tight integration, the

variations of the Kalman filter (KF) typically seen in the literature and a few examples where the integration of INS and UWB is seen in the literature.

5.5.1. Integration types

When integrating systems, there are mainly three types of integration methods that can be used, loose, tight and ultra-tight integration. The difference between them is the way data from different sensors are used.

In a loose integration, the measurements of the UWB are used to calculate a positioning solution. The UWB positioning solution is then used in the integration with INS. This type of integration is also sometimes known as position domain integration in the field of navigation, as the integration happens once the positioning solution is calculated from the individual systems prior to their integration (Groves 2013). In a tight integration the UWB measurements are used directly in the integration, without a UWB position solution being calculated. This is sometimes referred to as measurement domain integration, as the measurements are directly used in the integration (Groves 2013). In an ultra tight integration, processing begins at the signal level of the different sensors, before determining measurements for integration (Groves 2013). Ultra-tight integration is the most complex to use.

Although a loose integration is easier to implement, in cases where UWB measurements are not enough to calculate a position, the UWB data cannot be used in the integration. Alternatively, in a tight integration, all available measurements are used even when they are not enough to calculate the UWB position. Although tight integration is more complex than loose, it is expected to use data from the UWB system more frequently, calibrating the INS solution and reducing the duration of the time intervals in between which the INS drifts.

5.5.2. Methods of integration

Kalman Filter (KF) a is a method commonly used for sensor fusion. KF is a realisation of a Bayes filter and is used for recursive state estimation. The Bayes filter is a framework that estimates the current state of the dynamic agent, using the previous state estimation

and observations (such as sensor measurements). The KF is constituted by two main stages: state prediction and measurement update.

State prediction uses kinematics to predict the current state given the previous state estimate. At the stage of measurement update, observation equations are used to relate the new measurements to the state and update the prediction. The estimation of the current state is a combination between the prediction and measurement update stages. The exact form of the combination is determined by the Kalman gain matrix. This is a weight that determines the relative contributions of the two stages in the final state estimation. It is dependent on the covariance matrices of the predicted and measurement update stages.

Both the prediction and measurement update stages have their associated noise components, which correspond to uncertainties in the process (the way the prediction is being estimated) and the observations (or the measurements). In the KF the noise components of the prediction and measurement stages are assumed to follow a Gaussian distribution.

An important assumption in a KF is that a linear model is used to characterise the state at both the prediction and update stages. In reality, this is rarely the case, and as a result the Extended Kalman Filter (EKF) is commonly used for non-linear cases. The EKF has the same two stages as the KF, with the additional step of linearising the kinematic and observation equations. Linearisation is done using the first order term in a Taylor expansion, evaluated at the previous state estimate for the prediction stage, and at the predicted state for the measurement update stage.

In every cycle of KF and EKF, the previous estimate of the state, predicted state and measurement update are assumed to be normally distributed. In the case of a EKF, by linearising the nonlinear problem and evaluating at either the previous estimate or prediction, the mean is tranformed through the linearised model to realise a new distribution with the newly calculated mean.

However, when a Gaussian distribution is transformed through a non-linear function, the result is not necessarily Gaussian (Chebrolu 2019). EKF linearises the model about the mean of the state estimate and transforms the distribution about that point. However, it is not guaranteed that the transformation will be representative as it depends on the

point about which linearisation occurs (Chebrolu 2019).

An alternative to the EKF is the Error State KF (ESKF), where the filter solves for the error in the estimates instead of the actual state. In the ESKF the true state is composed of the nominal and error states (Sola 2017). The nominal state is the prediction of the state, calculated using equations of motion, and the error state considers the errors in the prediction. In the measurement update step of the ESKF, the measurements are used to correct the error state. The corrected error state is then added to the nominal state to obtain an estimate of the true state. As noted by Madyastha et al. (2011), Marković et al. (2022), the error state dynamics can be assumed to be linear, satisfying the linearity assumption of the KF and resulting in a more optimal prediction and update of the error state, causing the ESKF to outperform the EKF (Madyastha et al. 2011, Marković et al. 2022).

Yet another variant of the KF, the Unscented KF (UKF), eliminates the linearisation step altogether. Rather than linearising the model about the mean, the UKF applies the non-linear transformation to a number of points (known as sigma points), thereby creating a new distribution (Stachniss 2014*b*). The mean of the newly transformed distribution is then considered as the new state. UKF overcomes linearisation errors using a deterministic sampling approach and calculates the state distribution as a set of sample points (Brigadnov et al. 2023).

The different variations of the KF exist in the literature to calculate a position estimate by integrating measurements from different systems. EKF is the most popular version of the KF (Huang et al. 2023, Brigadnov et al. 2023, Yao et al. 2017, Marković et al. 2022). However, the linearisation of the non-linear model does not always result in a good representation of the system (Brigadnov et al. 2023).

It may be expected that the UKF outperforms the EKF due to linearisation errors in the latter (Brigadnov et al. 2023, Marković et al. 2022). Feng et al. (2020) integrate IMU and UWB, in the measurement domain, using an EKF and an UKF to compare their performance. Simulation results concluded that UKF outperformed EKF in terms of accuracy, resulting in a 17% improvement in the average position error when three anchors

were used and 28% improvement when one anchor was used. In contrast, D'Alfonso et al. (2015), compared the performance of UKF and EKF using empirical data and found the performance of the two filters to be comparable.

The use of ESKF is not as common as EKF for indoor environments, even though its implementation has significant potential (Brigadnov et al. 2023). Performance comparisons of the ESKF and the EKF presented by Madyastha et al. (2011), Liu et al. (2019), Guo et al. (2019), Youn & Gadsden (2019) show the robustness in performance of the ESKF. The use of ESKF is also found in Yao et al. (2017), Zhong et al. (2018), where the measurements from an IMU and a UWB are integrated. The integrated solution was shown to be more stable than the UWB standalone position and performed about two times better compared to the UWB standalone solution in terms of accuracy. An ESKF using quaternion kinematics was developed by Sola (2017) and was used by Brigadnov et al. (2023) and Marković et al. (2022), which according to Marković et al. (2022) allows for the quaternions to be treated with their minimum representation. The integration results in Brigadnov et al. (2023) showed that the accuracy performance of ESKF is better than EKF, while the accuracy performance of ESKF and UKF is comparable. This thesis implements an ESKF in chapter 9, to integrate UWB and IMU measurements in a tight integration.

5.5.3. IMU/UWB integration

As mention earlier UWB is a ranging based system that can achieve high accuracy range measurements due to the pulse nature of the signal resulting in high time resolution. Moreover, it has high penetration capabilities, that allow for NLOS signals while the pulse nature of the signal enables multipath resolution. However, in complex environment where NLOS conditions and multipath effects are significant, UWB performance is compromised (Zhong et al. 2018). This can either create gaps in the UWB positioning solution or result in large errors in position. This makes UWB a long term stable system, but short-term unstable.

Alternatively, an INS is a system with high sample rate, of which the performance is

not dependent on the surrounding environment. However, a known disadvantage of an INS is the drift and aging of the positioning solution with time. INS is a system that has short term stability with long term instability due to the drift and aging of the solution. Since UWB and INS systems are complementary, in this thesis they are integrated in an attempt to improve position availability and accuracy performance of the individual systems. A number of INS and UWB integrations are presented in the literature that show that the integration of the two results in a better accuracy performance that the UWB standalone solution (Zhong et al. 2018, Feng et al. 2020, Marković et al. 2022, Hol et al. 2009, Yao et al. 2017, Huang et al. 2023).

The majority of the reviewed studies, integrate the two systems using tight integration (Feng et al. 2020, Marković et al. 2022, Hol et al. 2009, Yao et al. 2017), while Zhong et al. (2018) presents loose integration. The performance of loose integration is dependent on the number of anchors and hence measurements available. If UWB measurements are available, but not enough to calculate a UWB solution, the predicted state is not corrected even though some measurements are available. This might result in larger intervals between UWB measurement updates, allowing the IMU solution to drift further. In contrast, when using tight integration, even if the number of UWB measurements are not enough to calculate the UWB position, they are still used, reducing the possibility of the IMU drifting the solution away. In this thesis, the integration between the two systems is done using tight integration.

Filtering of the UWB measurements prior to the integration is also seen in a number of publications (Marković et al. 2022, Hol et al. 2009, Yao et al. 2017), in an attempt to mitigate the integration of erroneous UWB measurements. Measurement filtering is usually applied by comparing the innovation of the filter to a filtering threshold. Different filtering thresholds are seen in the literature, related to either, the standard deviation of the measurements (Yao et al. 2017), error covariance matrices of the prediction and update stages of the filter (Hol et al. 2009) or being empirically tuned (Marković et al. 2022). The ESKF implemented in this thesis, also uses a measurement filtering stage, however the threshold used is defined in a different way and will be explain in chapter 9.

The accuracy performance of the presented IMU and UWB integrations vary. Marković et al. (2022) considers the operation of Unmanned Aerial Vehicle (UAV) and integrates data from IMU, LiDAR, cameras and UWB resulting in the root mean square error of the 3D solution to be 0.34m. Alternatively, Yao et al. (2017), integrates data from an IMU and UWB resulting in mean 2D position errors of 0.16m when moving in a straight line and 0.21m when in circular motion.

An in depth description of the ESKF used in this thesis as well as the implementation of the proposed integration is given in chapter 9. Empirical data are used to evaluate the performance of the integration.

5.6. Summary

This chapter presented the state-of-the-art systems and methods used in the literature. An overview of positioning methods and measurement techniques was presented in section 5.1. The state-of-the-art systems presented in section 5.3, were compared based on their accuracy performance, which resulted in the selection of UWB and INS for further investigation. Section 5.5 presented common integration methods with a focus on UWB and INS integration.

The accuracy performance of the selected systems are investigated further in a standalone and integrated modes. The performance of the UWB system is investigated in chapter 7, where empirical UWB data are used to estimate UWB-based solutions using the positioning method of multilateration as well as a hyperbolic method. In addition, the performance of the INS system and impact of acceleration measurements on the positioning solution is analysed in chapter 8. The integration of the two systems is performed in chapter 9, where an ESKF is implemented. The proposed integration uses a novel method of reducing the impact of IMU accelerations in the positioning solution, using the UWB measurements.

In order to evaluate the performance of UWB, IMU and their integration, empirical data are collected and used. The next chapter describes the data capture scheme design and execution for the collection of the empirical data used in the thesis.

Chapter 6

Data capture scheme design and execution

In chapter 5 an absolute and relative positioning systems, UWB and INS respectively, were chosen for further investigation. UWB is a system that can result in high positioning accuracy, however, its performance can be challenged in complex environments. Alternatively, performance of the INS is not dependent on the surrounding environment but the calculated positioning solution drifts over time due to error accumulation. The long term stability of the UWB and short term stability of an INS, indicate that these two systems complement one another, which led to their choice as candidate systems for further investigation. This chapter describes the set-up of an experiment to investigate the performance of the two systems separately and when integrated. Section 6.1 presents the room configuration, set-up of the systems, coordinate frames used, as well as the scenarios and configurations for which data were collected. Section 6.2 explains the data collection process and section 6.3 outlines the synchronisation of the different systems used, that allows for their integration.

6.1. Experiment set-up

The aim of the data collection is to evaluate the performance of UWB and INS in determining the position of a moving agent in an indoor environment. The performance is evaluated by considering the accuracy of the positioning solution, which requires information on the 'true' or reference position of the moving agent. The 'true' position of the moving agent is provided by a camera system, able to act as a tracking system. The equipment used throughout the data collection process includes:

- The camera motion capture system OptiTrack Primex 13, used as the 'truthing'

system.

- The Pozyx creator kit, used for collecting UWB ranging data.
- The NGIMU, used as the INS system.
- The NVIDIA JetRacer, a mini autonomous AI race-car used as the dynamic platform. The JetRacer is equipped with a Jetson Nano which is a small computer used for the operation of the JetRacer.

The experimental set-up considers room configuration, process of data collection, relationship between the coordinate frames of the systems used and configuration of the data collection scenarios.

6.1.1. Room configuration

The room configuration, showing the distribution of the sensors around the room, is illustrated in Figures 45 and 46. The cameras of the truthing system are attached on the tripods surrounding the space of motion, while the 8 UWB anchors are evenly distributed around the room.

The UWB anchors are evenly distributed around the room and are placed at at different elevations to ensure strong anchor geometry. This layout was decided based on the principle of Dilution of precision (DOP) explained in section 5.2.1. Even though DOP is a concept developed in the field of GNSS, its consideration is essential when setting up the UWB system. This is due to the similarity of the basic principles of UWB and GNSS, where in both cases range measurements are used to calculate a positioning solution and a strong anchor (or satellite) geometry reduces the uncertainty area in the calculated position. Figure 47 illustrates the distribution of the UWB anchors around the room, where 4 of the anchors are found at the corners of the room, close to the ceiling, and the remaining 4 anchors are found mid-span along each side of the room, close to the floor.

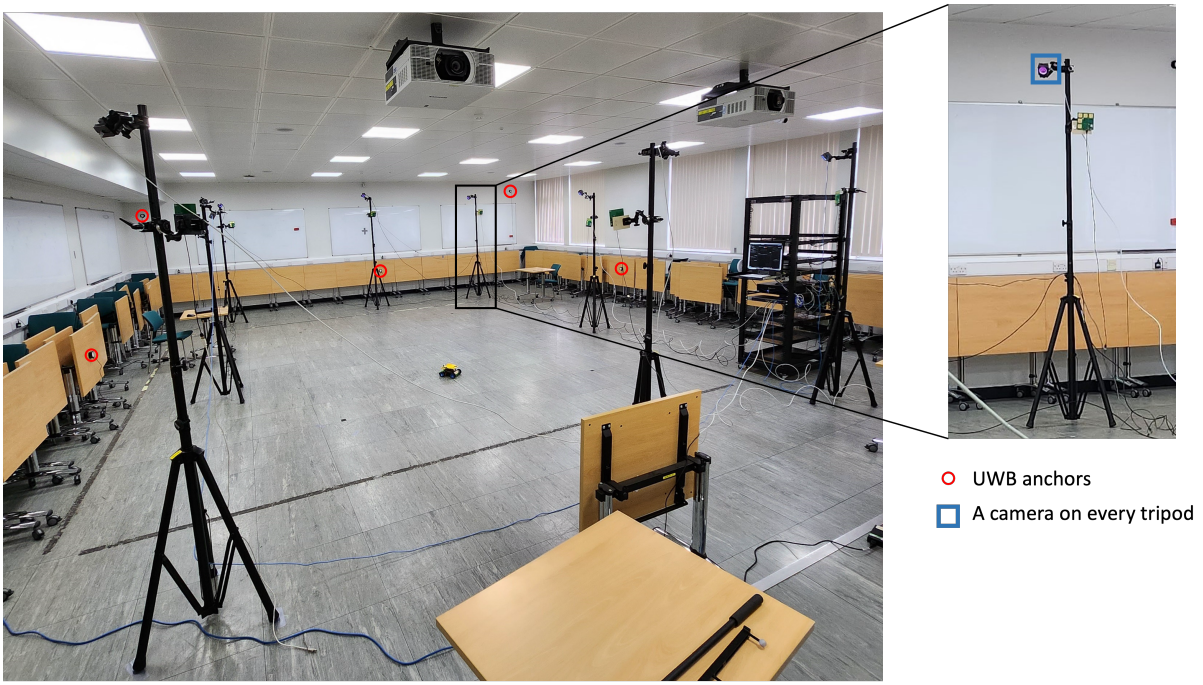


Figure 45: Room configuration.

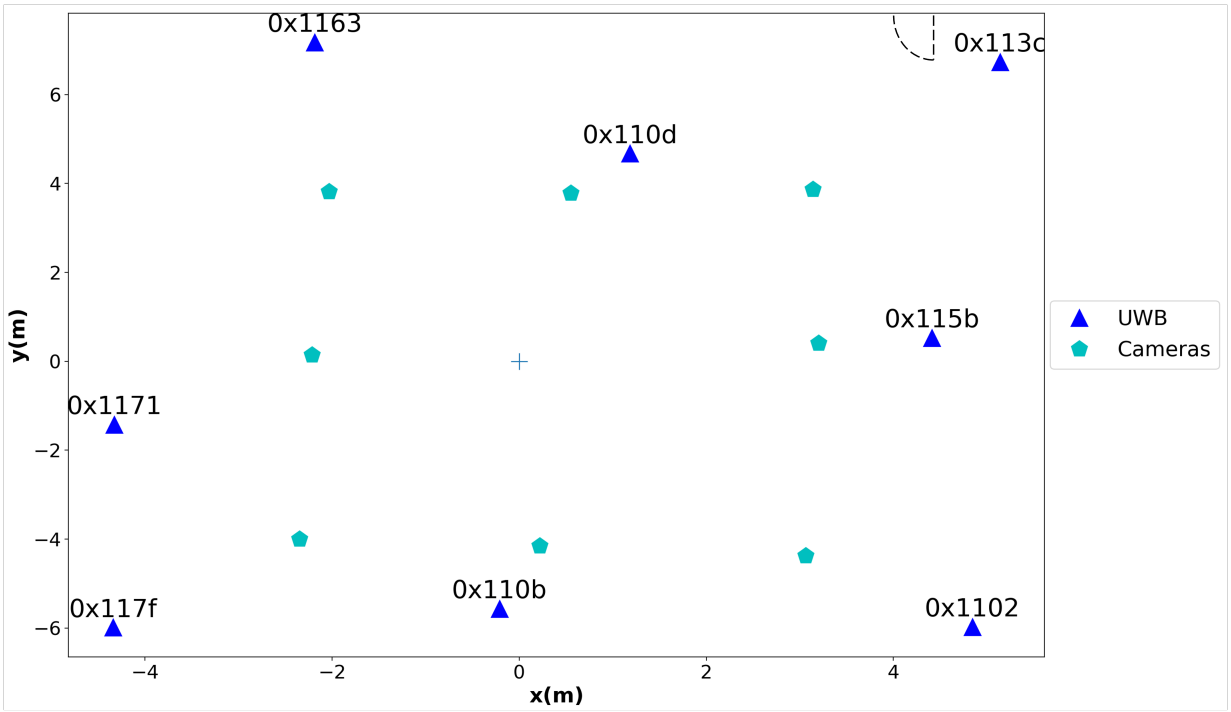


Figure 46: Plan view of the UWB and Bluetooth configuration.

An additional consideration when setting up the UWB anchors, is their proximity to metal surfaces. According to Pozyx (2022), the anchors should be placed about 20cm away from metal surfaces to avoid the degradation of antenna's performance. The antenna used is a metal conductor transmitting UWB signals. Its proximity to metal surfaces will reduce its efficiency and can tamper with its omnidirectionality in an unpredictable way (Pozyx 2022). The anchors close to the ceiling are hence attached to the walls, while the anchors close to the floor are attached to the folded desks, as shown in Figure 45 to avoid being close to the metal desk supports.

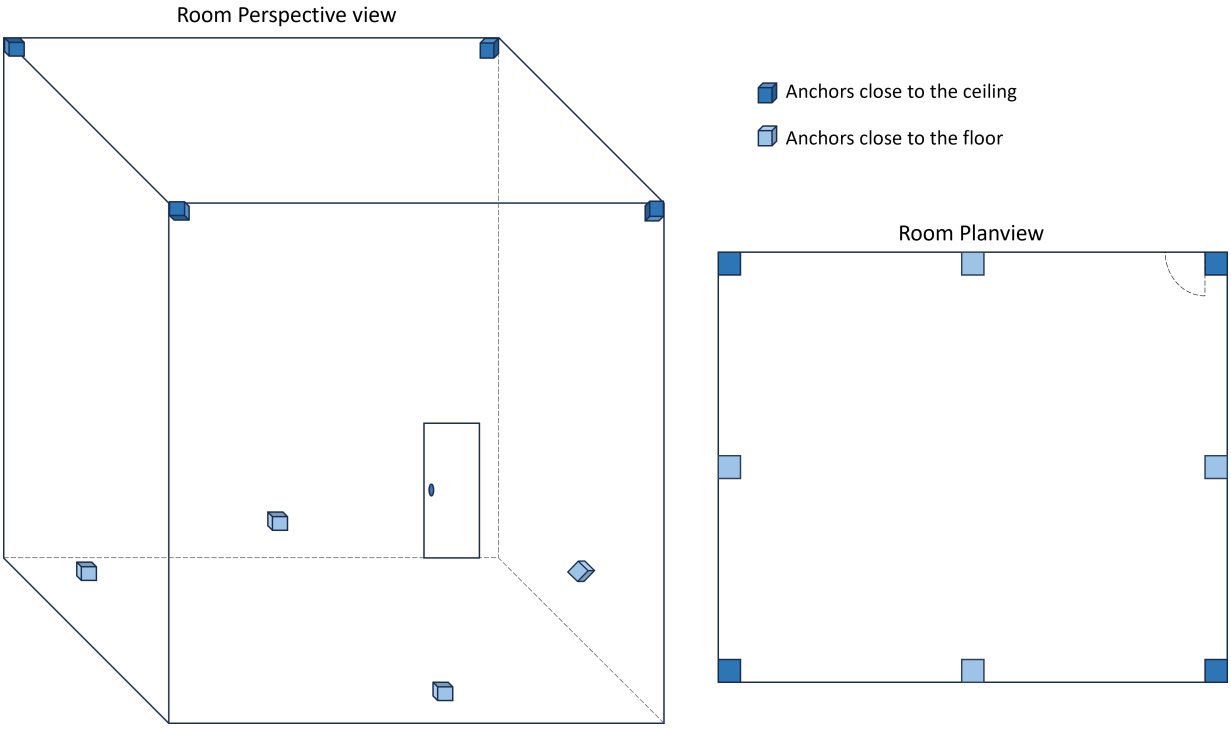


Figure 47: The distribution of the UWB anchors around the room.

6.1.2. Systems configuration

The configuration of the systems involves the set-up of the systems used (UWB and cameras) as well as setting up the dynamic robot platform.

The robot platform is equipped with a 3D-printed plate, which includes pockets for the different sensors, as shown in Figure 48. The plate includes, a UWB tag communicating

with the UWB anchors, the four markers tracked by the camera system and an IMU. The configuration of the sensors on the plate is shown in Figure 48.

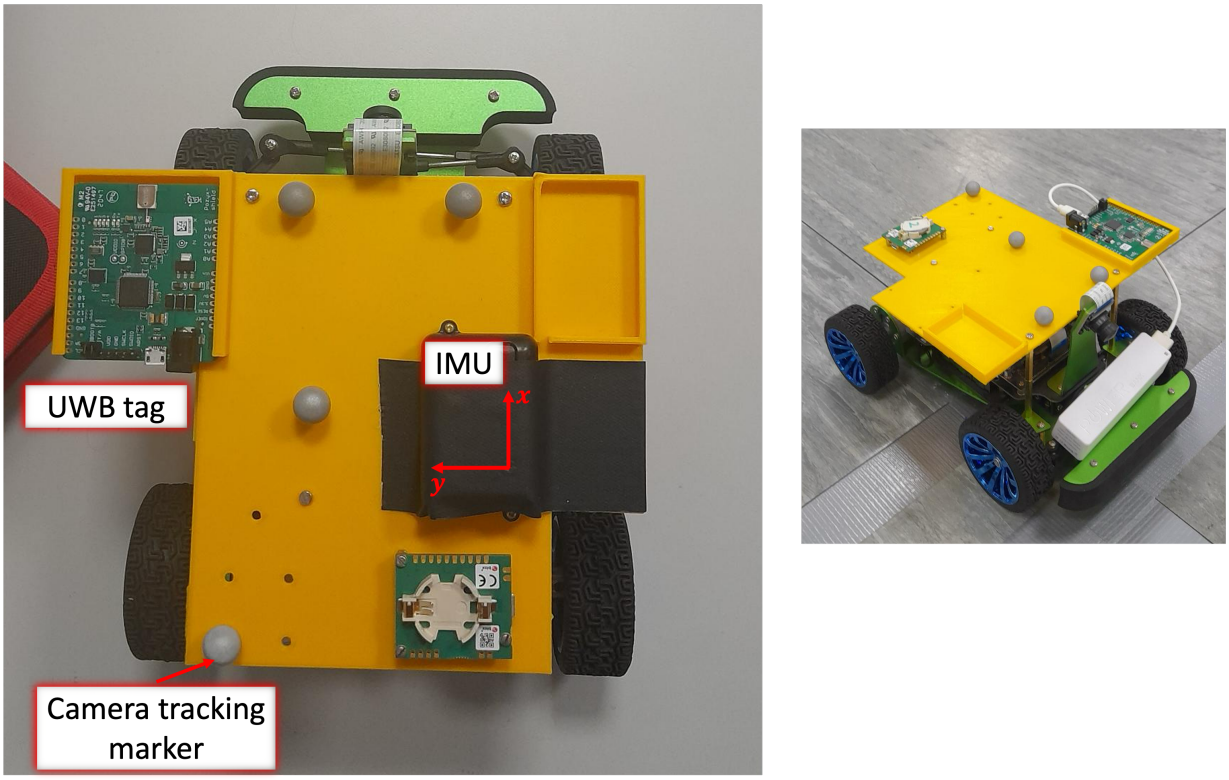


Figure 48: The dynamic platform equipped with the UWB tag, IMU and the markers tracked by the camera system.

Other than the dynamic platform, the UWB and camera system need to be set-up around the room. As can be seen from Figure 46, the camera system has a good coverage of the area of motion. However, the UWB anchors are not in the view of the cameras and hence their coordinates cannot be determined using the camera truthing system. As a result the UWB anchors are set-up using a Total Station (Leica TS 10) to determine their fixed coordinates. The local coordinate frame for the total station is aligned with that of the camera system to minimise the necessity for coordinate conversion. This is because the cameras are used as the truthing system and consequently the fixed position of the anchors need to be in the same coordinate frame as the cameras. The origin and orientation of

the camera coordinate frame are set-up using a calibration square. The camera calibration square and orientation of the total station are aligned, with both frames using the same origin, to ensure that the coordinate frames of the two systems match.

The camera motion capture system, used as the truthing system, can track a moving agent down to mm level of accuracy ($+/-0.2mm$ according to OptiTrack (2023b)). The eight cameras used are mounted on tripods and strategically arranged in a layout designed to guarantee comprehensive coverage of the robot's operational area. A good camera coverage is achieved through overlapping cameras' fields of view, as shown in Figure 46. The cameras calculate the 3D position of the tracking markers attached on the moving object, using triangulation from multiple cameras (OptiTrack 2023a). After configuring the cameras to capture the area of motion, calibration takes place by establishing the origin and coordinate frame using a calibration square. The camera's coordinate frame is aligned with the origin and orientation of the total station's frame.

6.1.3. Coordinate frames

The coordinate frames considered for this data collection are the body and local coordinate frames. The body coordinate frame is represented by the IMU coordinate frame, with the x-axis pointing towards the forward direction of motion as shown in Figure 48. The local coordinate frame is chosen to be the same as the frame of the cameras and the total station. The two coordinate frames are illustrated in Figure 49.

The transformation between the IMU and local coordinate frames is not constant as the IMU moves in space and hence its orientation keeps changing. As the IMU is a relative system, it needs to be initialised in order to calculate a positioning solution. The IMU is initialised using data from the cameras, and IMU accelerations and changes in orientations are used to update position estimate.

As the dynamic platform and hence the IMU move on a flat ground, it is assumed that initially, while the IMU is static, there is neither roll nor pitch misalignment between the IMU and local coordinate frame. The initial orientation of the IMU is consequently determined by calculating the yaw angle between the initial direction of motion and room's

positive x-axis. The initial direction of motion is determined by calculating the tangent of the true motion recorded by the cameras in the x-y plane, as shown in Figure 50.

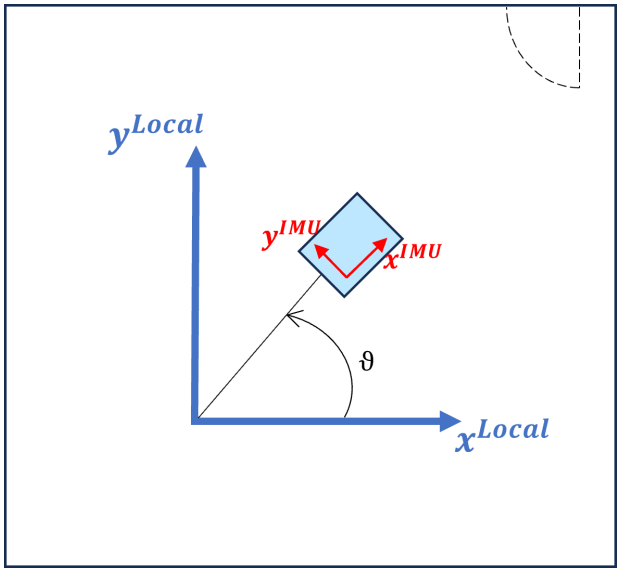


Figure 49: The conversion between the IMU and the local coordinate frame.

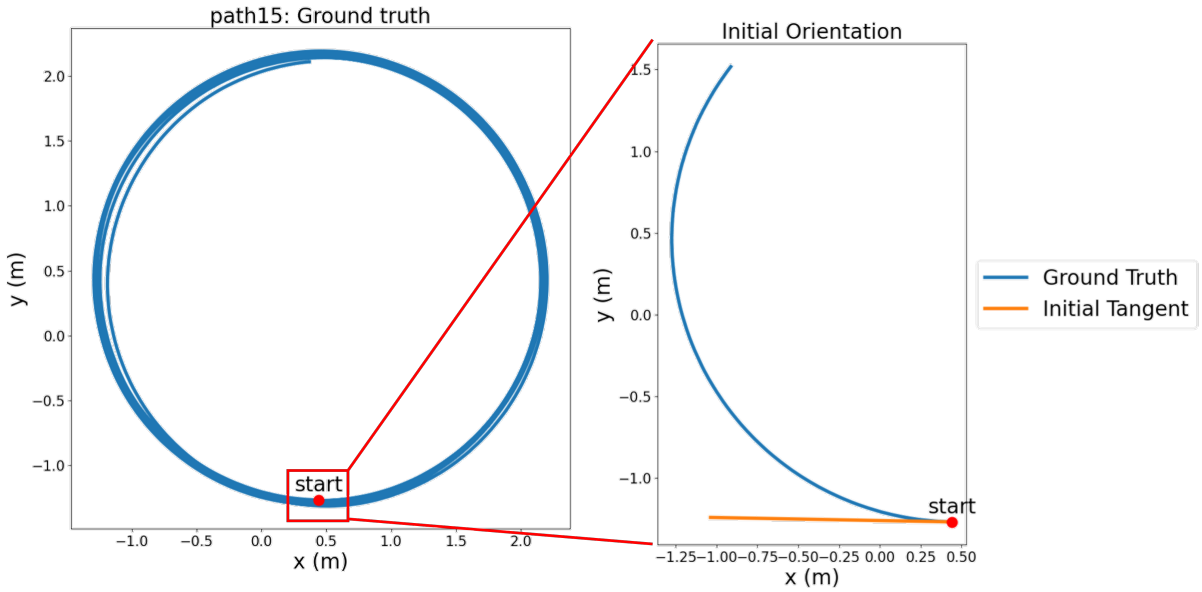


Figure 50: The initial direction of motion, determined as the tangent of the true motion recorded by the cameras in the x-y plane.

6.1.4. Data collection scenarios/configurations

During the data acquisition phase, both static and dynamic modes were considered. For the static modes, only UWB data were collected, while for dynamic modes both UWB and IMU data were collected. The distribution of the eight static positions around the room is illustrated in Figure 51.

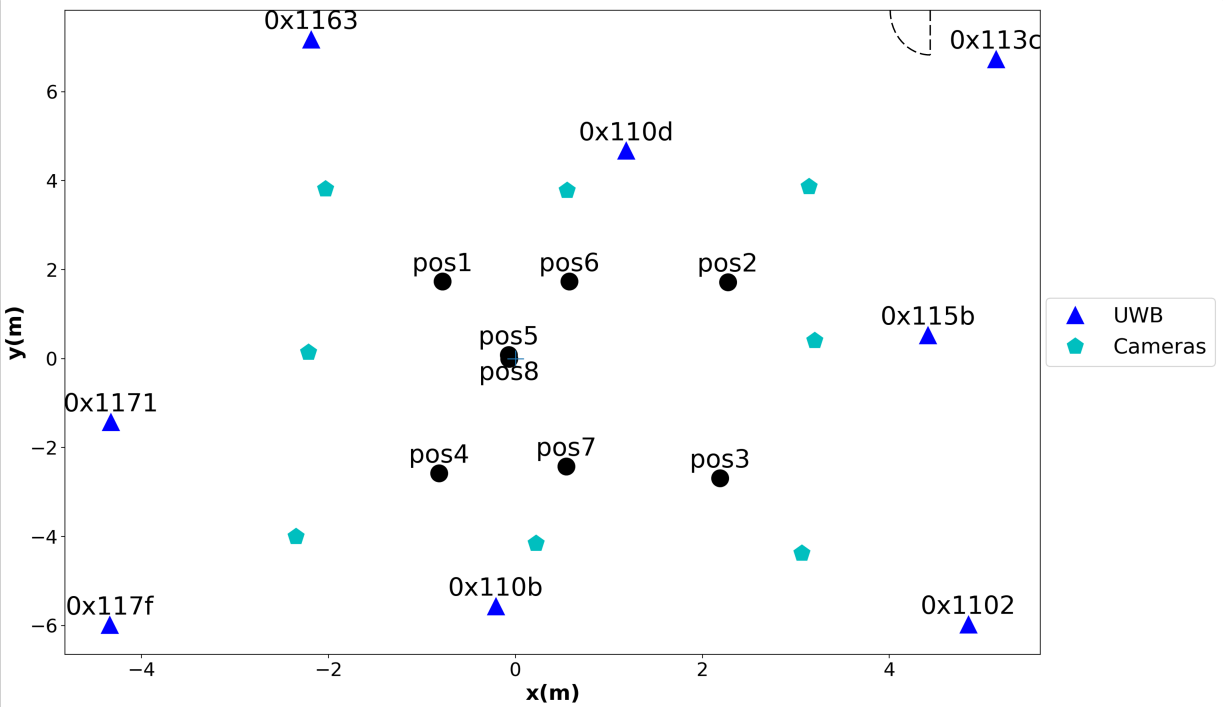


Figure 51: Configuration of the collected static points.

For the dynamic modes, data were collected from both UWB and IMU. These data are split in three main categories based on the motion of the dynamic platform. These correspond to straight diagonal lines, circular paths and random motion within the test area. A representation of the three types of dynamic paths is shown in Figure 52. Moreover, some of the circular paths were collected with a human acting as an obstacle standing in the middle of the test area. A summary of the dynamic data is presented in table 6. As shown in table 'diag6' and 'path4' do not have information of average speed. This is because the true position was not recorded for the whole duration of these two trajectories and as a result, measurements from these two paths will not be processed.

Table 6: Data collection process of dynamic paths

Dynamic Paths	Average speed (m/s)	Description
diag1	0.183	straight line
diag2	0.181	
diag3	0.180	
diag4	0.386	
diag5	0.386	
diag6	-	
diag7	0.571	
diag8	0.565	
diag9	0.568	
path1	0.366	random path
path2	0.366	
path3	0.214	
path4	-	
path5	0.207	
path6	0.431	
path7	0.434	
path8	0.441	
path9	0.660	
path10	0.665	
path11	0.657	
path12	0.741	circular
path13	0.729	
path14	0.721	circular with obstacle
path15	0.483	at the centre of the circle

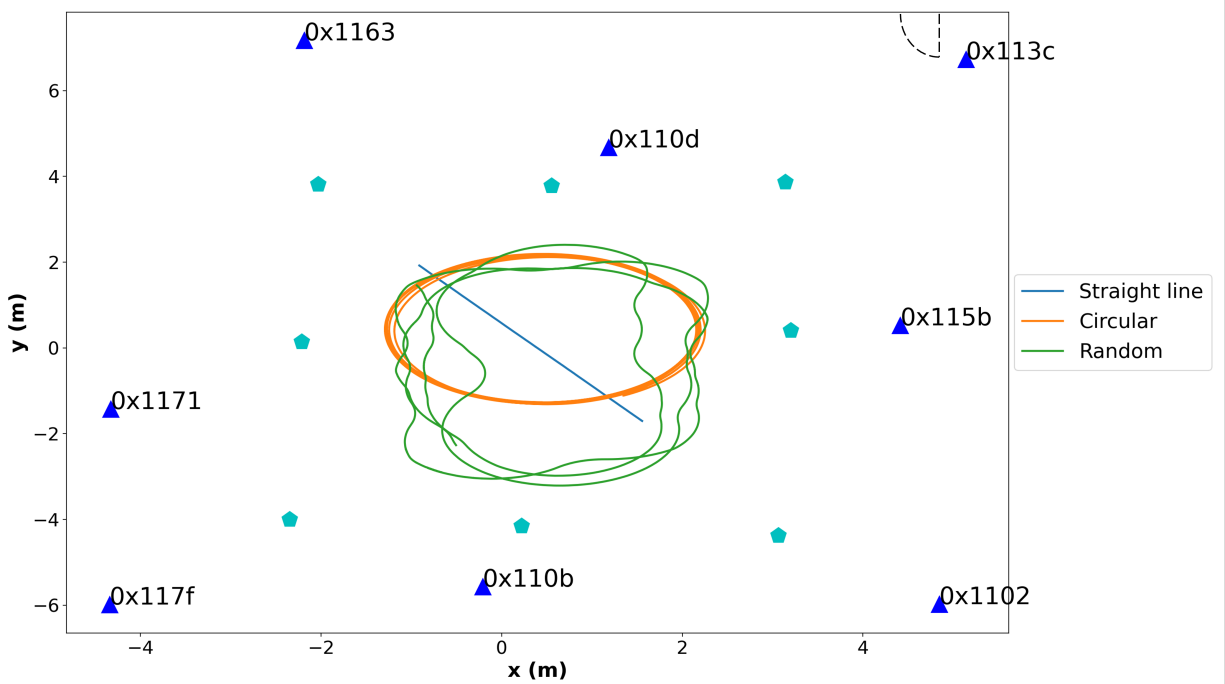


Figure 52: Types of dynamic data.

This section explained the way the experiment was set-up, through room and system configuration, explaining the coordinate frames used, and describing the static and dynamic data collection scenarios. The following section explains the data collection method used for each system and the type of collected measurements.

6.2. Data collection process

This section provides an overview of the type of data collected from each system as well as the data collection procedure used.

6.2.1. IMU

Data collected from the IMU include readings from the accelerometer, gyroscope and magnetometer in the three axes of the IMU, as shown in Table 7. The IMU sensor attached on the dynamic platform, is remotely connected to a computer and sends timestamped collected data at a rate of $50Hz$.

Table 7: IMU collected data (x-io Technologies 2022, p. 17).

Measurement	Units
Gyroscope x axis	°/s
Gyroscope y axis	°/s
Gyroscope z axis	°/s
Accelerometer x axis	g
Accelerometer y axis	g
Accelerometer z axis	g
Magnetometer x axis	µT
Magnetometer y axis	µT
Magnetometer z axis	µT
Barometer	hPa

6.2.2. Cameras

The camera system employs triangulation, with overlapping perspectives from multiple cameras, to deduce the 3D position of a dynamic object. The configuration formed by the reflective markers on the dynamic platform (sown in Figure 48), establishes a unique rigid body for tracking purposes. The marker highlighted in Figure 48 is manually designated as the center of the rigid body. This ensures that the position deduced by the cameras aligns with a clearly defined point on the plate.

The camera system records data at a frequency of $120Hz$. The recorded data used in post-processing include the capture start time and for each epoch the temporal information of the duration elapsed since the capture start time, and the 3D coordinates of the tracked marker.

6.2.3. UWB

The data collected from the UWB system comprises time information, ID of the anchor from which data are collected, range data between the anchor and moving tag, as well as the received signal strength. The range information provided is determined using two-way ranging (TWR).

The method of UWB data collection follows a local-remote configuration. The local device is connected to a computer, which then communicates with all the remote devices and allows for specific functionalities to be performed on remote devices. In this specific scenario, the local device communicates with the remote devices (the different anchors around the room) and obtains range information between each anchor and the tag. The local device systematically accumulates ranging data by sequentially cycling through the eight anchors.

This section explained the data collection methods and types of measurements from each system. Once the data collection process was complete a preliminary visualisation of the UWB, camera and IMU data illustrated some timing inconsistencies between the three systems. The following section explains the observed inconsistencies and the way in which synchronisation issues are resolved.

6.3. Time Synchronisation between the different sensors

Time synchronisation is pivotal for seamless integration of the systems, as it guarantees that the gathered data correspond to a uniform time instance amongst the systems. Of the systems used for data collection (UWB, IMU and camera), all but the UWB system provide timestamps synchronised with UTC time. Alternatively, each UWB anchor provides timestamps as the time elapsed since setup. Even though IMU and cameras are both using UTC-timed timestamps, the raw data show the first instance of motion (FIM) of the dynamic platform occurring at slightly different times for the two systems. The FIM corresponds to the time that a system first shows sign of motion. The deviation between the FIM of the cameras and IMU systems is illustrated in Figure 55, where the red line corresponds to the FIM of the camera system and black line to the FIM of the IMU.

As mentioned earlier, synchronising the systems is essential in ensuring that measurements from different systems refer to the same time and hence space. Since the FIM of the cameras and IMU was used to examine the synchronisation between the two systems, it is also used to synchronise the systems. The FIM is calculated using different parameters depending on the system examined. These parameters are either the raw measurements or parameters derived from the raw measurements of the different systems. As the raw measurements differ for the different systems, the parameters used to determine the FIM will be dependent on the system used. A more detail explanation of the method used to calculate the FIM for every system is provided in the following sections.

Once the FIM of every system is determined, all the systems are synchronised with the IMU timestamps. The timestamps of the IMU and camera system, are initially converted from UTC time to GPS time of the week, that corresponds to the elapsed number of seconds within the specific week. The following sections give a detailed explanation of the way the FIM are determined, and the way synchronisation between the three systems is achieved.

6.3.1. IMU

As discussed in section 6.1.3, the IMU's x axis aligns with the forward direction of the dynamic platform. The FIM of the IMU system is hence determined by analysing the accelerations along the IMU's x axis. Raw acceleration measurements in the three orthogonal directions are shown in Figure 53. Figure 53 shows a large variation in accelerations, while the measurements in all three directions have two stages. A stage where the variations in the measurements is relatively small and a stage during which there is large variation. These stages represent the static and dynamic stage of the trajectories, as shown in Figure 53. The distinction between these two stages allows for the calculation of the FIM. FIM is defined as the time where the variation in acceleration suddenly increases. The FIM of the IMU sensor (T_{motSt}^{IMU}) is calculated using the variation in accelerations and corresponds to the time indicated by the black line in Figure 53

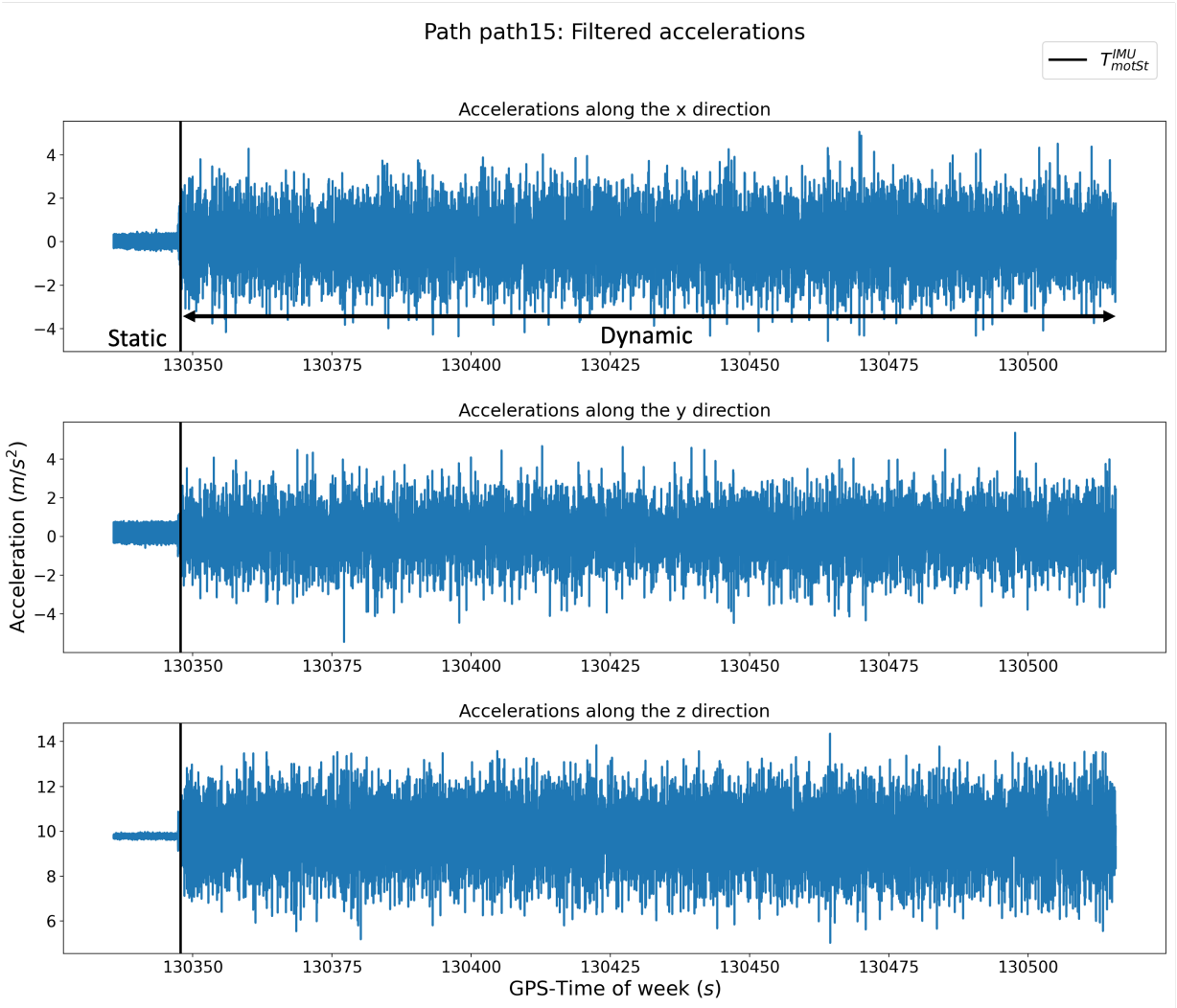


Figure 53: Identification of FIM using IMU raw accelerations.

6.3.2. Cameras

When considering the camera system, the FIM is determined using the speeds derived from camera's position measurements. The camera derived speeds are shown in Figure 54 for path15. Figure 54, shows the static and dynamic stages discussed in the case of IMU. The distinction between the two stages in the camera system, is the value of speed. Estimated speeds close to $0m/s$ correspond to the static part of the trajectory, while speeds that significantly deviate from zero correspond to the dynamic part of the trajectory. The switch between the two stages is a sudden increase in the value of speed, and corresponds

to the FIM of the camera system (T_{motSt}^{CAM}). This is shown by the red line in Figure 54.

The speed and T_{motSt}^{CAM} calculation is done by

1. Calculating the distance moved between consecutive epoches (ds)
2. Calculating time elapsed between consecutive epoches (dt)
3. Calculating velocity between consecutive epoches $v = ds/dt$ in m/s
4. The first instance when epoch velocity significantly deviates from $0m/s$, is considered as the FIM of the camera system.
5. The GPS time of week of this instance is set to be the T_{motSt}^{CAM}

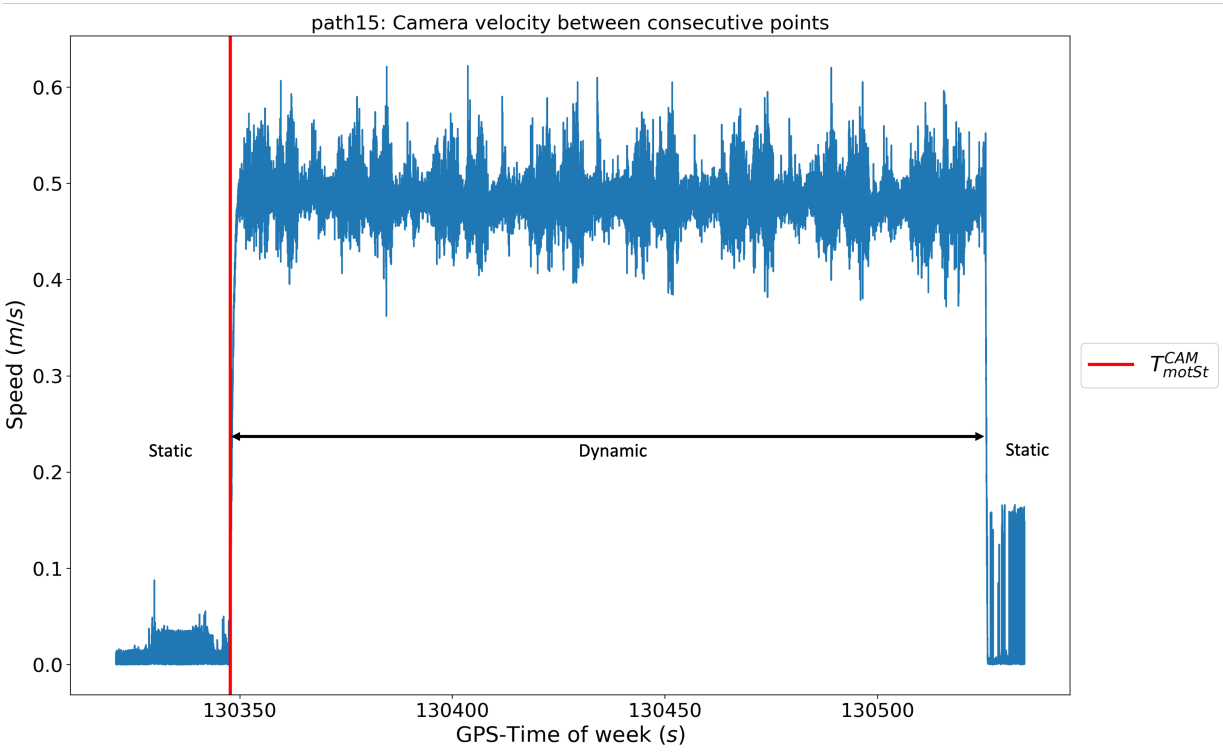


Figure 54: Identification of first instance of motion using epoch-to-epoch velocity calculated by the camera system.

The time difference between the IMU and the camera system, T_{diff}^{IMUcam}, is calculated in equation (6). This time difference is added to the timestamps of the camera epoches to

synchronise the camera system to the IMU. A visualisation of the difference between the FIMs calculated for the two systems (T_{motSt}^{CAM} and T_{motSt}^{IMU}) is presented in Figure 55. Figure 55 presents the IMU acceleration measurements with T_{motSt}^{CAM} represented by the red line, T_{motSt}^{IMU} by the black line and T_{diff}^{IMUcam} being the difference between the two.

$$T_{diff}^{IMUcam} = T_{motSt}^{IMU} - T_{motSt}^{CAM} \tag{6}$$

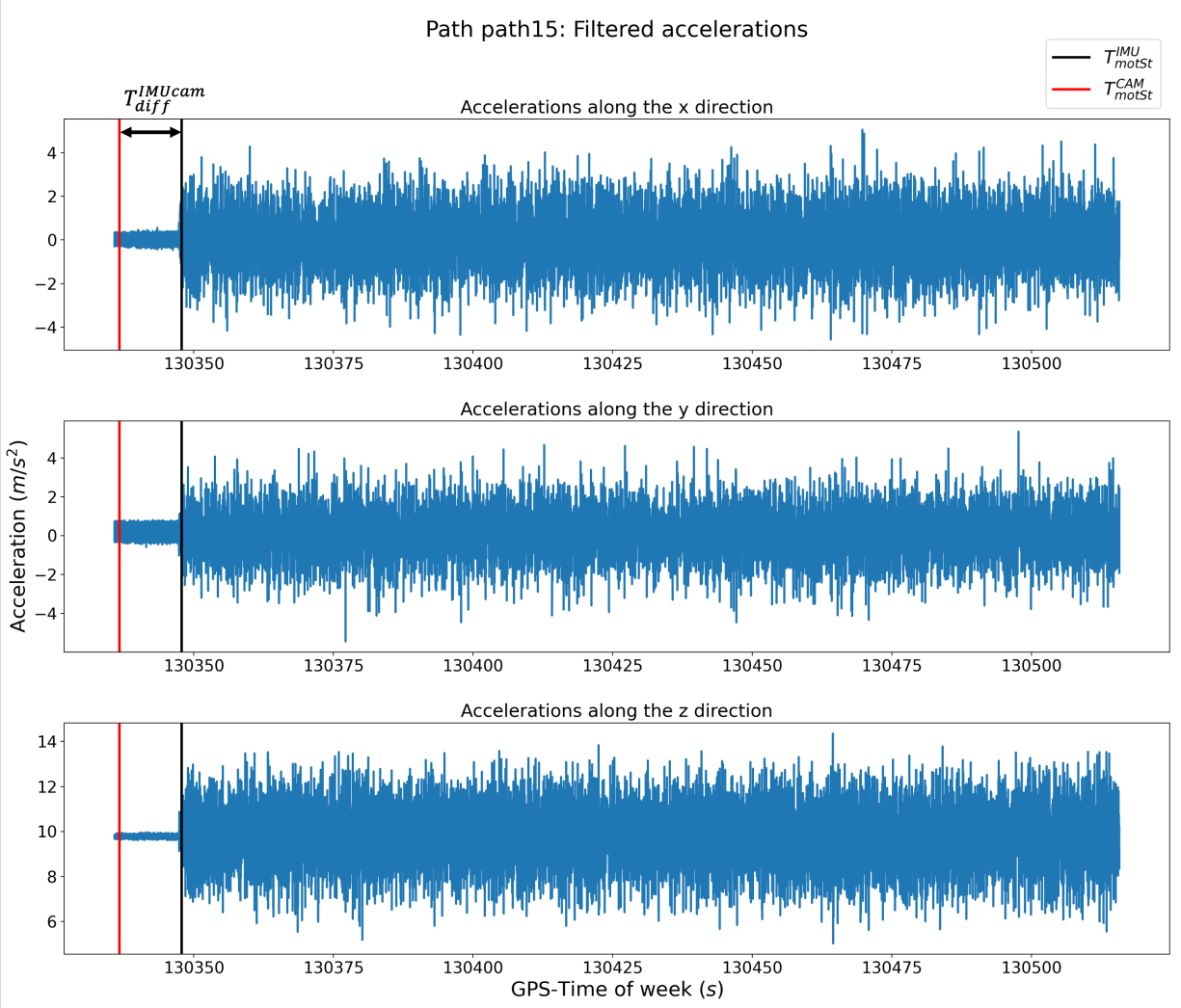


Figure 55: Difference between the IMU and camera's first instance of motion.

6.3.3. UWB

In the case of UWB, the time synchronization process includes the identification of the FIM, and allocation of timestamps according to the specific epoch. As explained in section 6.2.3, the local device loops through 8 anchors to obtain ranging data. For this reason, an epoch in the data is considered as a block of 8 measurements from the anchors. In the process of correcting the UWB timestamps, the initial goal is to identify the epoch that corresponds to the FIM. The FIM is allocated the timestamp of T_{motSt}^{IMU}, and all other epoches are then time-stamped accordingly.

In an attempt to analyse the UWB data for the purpose of timestamp assignment, collected distance measurements are plotted against time for each anchor and every dynamic path. An example of the distance data collected from all anchors, for two different dynamic cases is shown in Figures 56 and 57.

Figure 56 shows range measurements obtained during a circular path, while Figure 57 shows measurements from a straight line path. In Figure 56 measurements from all the anchors illustrate a very similar pattern, characterised again by two stages. During one stage, the range measurements are stable at a constant point, while after some point the recorded range measurements begin to oscillate (second stage). These two stages represent the static and dynamic phases of the trajectory. While the robot platform is static, the range measurements stay at a constant value. Alternatively, once the robot platform moves in a circular patter, range measurements oscillate as the robot platform moves towards and away from the anchor.

In Figure 57, a different pattern is observed. In this case there is the case of constant range measurements, however, in the second stage, range measurements seem to have an either increasing or decreasing trend depending on the anchor. The constant range measurements correspond to the static phase of the trajectory, while the trends correspond to the dynamic phase. Increasing trend correspond to cases where the robot platform moves away from the anchor and decreasing trends to cases where the robot platform moves towards the anchor. In all path types, the UWB FIM is defined as the transition

between static and dynamic phases.

As the initial goal is to identify the epoch corresponding to the FIM and each epoch has 8 measurements from the 8 anchors, a reference anchor is chosen. The distance profile of the reference anchor is then monitored to identify the FIM epoch. Figures 56 and 57 show that anchors 0x110d and 0x110b, provide a smoother distance profile compared to the rest of the anchors, making it easier to identify the FIM. The selection between 0x110d and 0x110b as reference anchor, is path dependent as for some paths 0x110d provides a more representative FIM than 0x110b and vice versa.

Once the FIM epoch is identified, the timestamps are corrected using the specific epoch as the FIM. The FIM recorded by the UWB systems is identified by:

1. Calculating the average initial distance to the reference anchors, using the first 10 measurements, when the robot platform is static. The initial average distance of the reference anchor is defined by $d_{static}^{refAnch}$

2. Calculating the absolute deviation ($d_{absDev}^{refAnch}$) of every epoch measurement ($d_{meas}^{refAnch}$) from $d_{static}^{refAnch}$ by
$$d_{absDev}^{refAnch} = |d_{meas}^{refAnch} - d_{static}^{refAnch}|$$

3. Identifying the FIM as the instance when $d_{absDev}^{refAnch}$ significantly deviates from $0m$. This instance is set to be $T_{motSt}^{refAnch}$.

4. Splitting the recorded data into epoches, where every epoch is considered as a block of 8 measurements from the 8 different anchors.

5. Identifying the epoch that includes $T_{motSt}^{refAnch}$. This is the epoch of FIM, defined as $Epoch_{motSt}^{UWB}$.

6. Setting the FIM of every anchor T_{motSt}^{anchor}, as the time recorded in the $Epoch_{motSt}^{UWB}$ epoch.

Having identified the FIM for every anchor (T_{motSt}^{anchor}), the collected data are split by anchor. The timestamp of every epoch is assigned by equation (7) to align with the IMU time.

$$T^{UWB} = T_{meas}^{anchor} - T_{motSt}^{anchor} + T_{motSt}^{IMU} \tag{7}$$

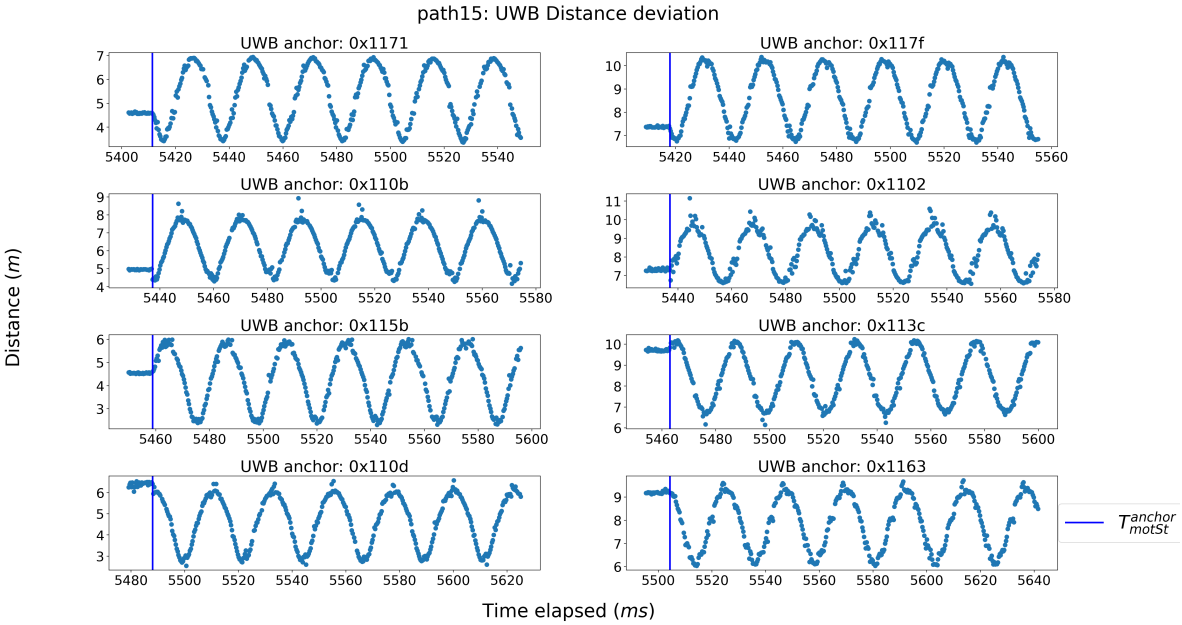


Figure 56: Identification of first instance of motion of a circular path using UWB.

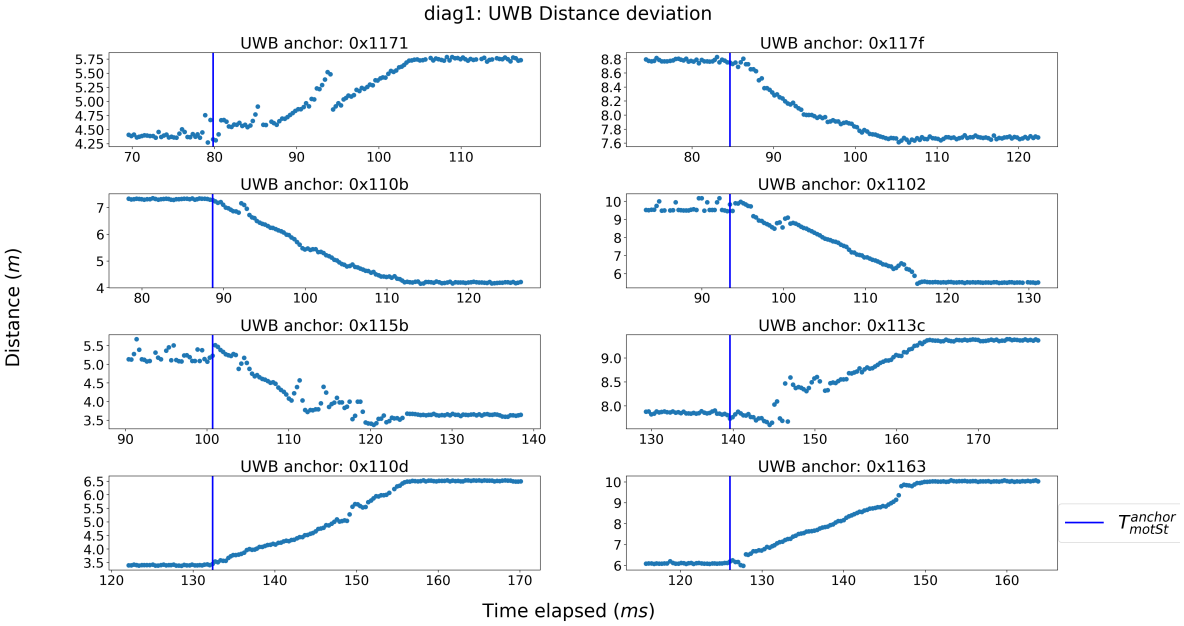


Figure 57: Identification of first instance of motion of a straight path using UWB.

6.4. Summary

This chapter provided a detailed explanation of the scheme used to collect data for further analysis in the thesis. It provided an overview of the experimental set-up including the room and system configuration as well as the convention between coordinate frames and the scenarios for which data have been collected. It then explained the data collection process of the different system, while it also gave a detailed explanation of the synchronisation process between the systems. The data collected through this data capture scheme will be used in the thesis to evaluated the performance of the UWB and IMU systems.

Chapter 7

Ultra-Wideband measurement and position domain performance characterisation

In this chapter, raw UWB range measurements are used to calculate the position of a UWB tag. The chapter has two main sections: analysis of raw UWB measurements and computation of position. For the former, measurement errors are calculated for both static and dynamic cases and their distributions are assessed. Moreover, measurement error variations of dynamic cases are analysed to determine whether there are any dependencies of measurement errors on the motion of the dynamic platform.

Once the raw measurements are analysed, they are used to estimate the position of the dynamic UWB tag using the hyperbolic and multilateration models to estimate the position using the least squares method. These methods are used to compute the position for both static and dynamic cases, and their associated position errors are used in an attempt to compare their performance with respect to accuracy.

7.1. Methodology

Least-squares based hyperbolic and lateration methods are employed for position computation. This is followed by performance assessment in terms of the metric of accuracy.

7.1.1. Least Squares based position estimation

Least squares (LS) technique is commonly used as a position fixing technique that uses the relationship between the observations (or measurements) and the unknown state, to calculate a state (or position) estimate. The position estimation of LS is calculated based on the objective of minimising a quadratic form of the residuals (Cross 1983). The position

is calculated based on the observations which can be quantities like distance measurements. A vector of observations, in this case range observations (ℓ), is related to the true value of observations ($\bar{\ell}$) by

$$\bar{\ell} = \ell + \nu \tag{8}$$

where ν is a vector of the residuals. The mathematical relationship between the true observation and the true state estimation parameters ($\bar{x}$), is defined as:

$$F(\bar{x}, \bar{\ell}) = 0 \tag{9}$$

In the case of position estimation, relationship (9) is usually a non-linear model. Linearisation using the first order terms of a Taylor series expansion is then performed to linearise the model. The linearisation process requires the use of approximations for the true state parameters and observations, $\bar{x}$ and $\bar{\ell}$. The approximations of $\bar{x}$ and $\bar{\ell}$ are x^0 and l respectively, and the relationships between true and approximated values are presented by equations (10) and (8).

$$\bar{x} = x^0 + x \tag{10}$$

Replacing the true values with equations (10) and (8) in $F(\bar{x}, \bar{\ell})$ results in:

$$F(\bar{x}, \bar{\ell}) = F\left(x^0 + x, \ell + \nu\right) = 0 \tag{11}$$

allowing for the Taylor series expansion to be applied, and the first order terms to be obtained as:

$$F(\bar{x}, \bar{\ell}) = F\left(x^0, \ell\right) + \left.\frac{\partial F}{\partial \bar{x}}\right|_{x^0} x + \left.\frac{\partial F}{\partial \bar{\ell}}\right|_{\ell} v = 0 \tag{12}$$

which can also be expressed as:

$$Ax + Cv - b = 0 \tag{13}$$

where A corresponds to $\left.\frac{\partial F}{\partial \bar{x}}\right|_{x^0}$ and is known as the design matrix, C corresponds to $\left.\frac{\partial F}{\partial \bar{\ell}}\right|_{\ell}$ and $-b$ to $F\left(x^0, \ell\right)$

When considering the method of lateration, the observation equations are:

$$\begin{aligned} d_1 &= \sqrt{(x_1 - x)^2 + (y_1 - y)^2 + (z_1 - z)^2} \\ d_2 &= \sqrt{(x_2 - x)^2 + (y_2 - y)^2 + (z_2 - z)^2} \\ &\vdots \\ d_n &= \sqrt{(x_n - x)^2 + (y_n - y)^2 + (z_n - z)^2} \end{aligned} \tag{14}$$

where d_i is distance observation from anchor node i and corresponds to $\bar{\ell}$, (x_i, y_i, z_i) are the known coordinates of anchor node i and (x, y, z) are the unknown coordinates of the dynamic platform and correspond to the state estimate $\bar{x}$. Equation (9) is then expressed as:

$$F(\bar{x}) - \bar{\ell} = 0 \tag{15}$$

As a result:

$$C = \frac{\partial F}{\partial \bar{\ell}} = -I \tag{16}$$

and equation (13) is re-written as:

$$Ax = b + \nu \tag{17}$$

Equation (17) is the linear model for observation equations.

LS estimates the state by minimising the square of the residuals. The minimisation objective is to find x and ν that minimise $\nu^T \nu$ subject to the constraints $Ax - \nu - b = 0$. The minimisation problem is solved using the Lagrange's method and the estimate of x is:

$$\hat{x} = \left(A^\top A\right)^{-1} A^\top b \tag{18}$$

The LS solution for x in (18) assumes all the observations have the same weight. One variation of the LS is the weighted LS (WLS), where the weights of the observations are taken into consideration when estimating x, such that:

$$\hat{x} = \left(A^\top W A\right)^{-1} A^\top W b \tag{19}$$

where W is the weight matrix, calculated as the inverse of the covariance matrix of observations (Cross 1983).

In this thesis LS is used instead of WLS, as the characteristics of the measurement distributions are not fully understood, leading to an unrealistic covariance and weight matrix in the case of WLS.

Two methods are used to calculate the position estimation using the UWB range measurements. The first method is the method of multilateration illustrated in figure 34, while the second method is a hyperbolic method. These two methods have different observation equations. The method of multilateration is characterised by non-linear observation equations while the hyperbolic method results in a linear system of equations. As a result, the method of multilateration requires the linearisation of the system and an initial state approximation. The two methods are presented in detail.

7.1.2. Multilateration method

The basic observation equation of the method of multilateration is equation (20). As this is a non-linear relationship the system needs to be linearised as described in section 7.1.1 with the use of an initial state approximation. As shown in equation (10), the x estimate of the LS corresponds to the deviation of the initial approximation from the truth. The implemented method uses an iterative process, that monitors the magnitude of x, to estimate the position at every epoch $\bar{x}$. An epoch is considered as a collection of eight range measurements form the eight different anchors. The iterative process monitors the estimate of x and the iterations stop when the value of x becomes smaller than a pre-defined threshold, or when the number of iterations exceeds a pre-set maximum number of iterations.

$$d_i = \sqrt{(x_i - x)^2 + (y_i - y)^2 + (z_i - z)^2} \tag{20}$$

As every epoch is made up of eight range measurements, the mathematical model $F(\bar{x}) - \bar{\ell} = 0$ of every epoch is expressed as:

$$\begin{aligned}
\sqrt{(x_1-x)^2+(y_1-y)^2+(z_1-z)^2}-d_1&=0\\
\sqrt{(x_2-x)^2+(y_2-y)^2+(z_2-z)^2}-d_2&=0\\
&\vdots\\
\sqrt{(x_8-x)^2+(y_8-y)^2+(z_8-z)^2}-d_8&=0
\end{aligned} \tag{21}$$

An initial state approximation for every iteration is provided as $x^0_{iter}=(x^{iter}_0, y^{iter}_0, z^{iter}_0)$ and the design matrix A is calculated for every iteration as the Jacobian of the system of equations (21) as:

$$A=\left.\frac{\partial F}{\partial \bar{x}}\right|_{x^0}=\begin{bmatrix}
\frac{x^{iter}_0-x_1}{d^0_1} & \frac{y^{iter}_0-y_1}{d^0_1} & \frac{z^{iter}_0-z_1}{d^0_1}\\
\frac{x^{iter}_0-x_2}{d^0_2} & \frac{y^{iter}_0-y_2}{d^0_2} & \frac{z^{iter}_0-z_2}{d^0_2}\\
. & \vdots & \vdots\\
. & . & \vdots\\
\frac{x^{iter}_0-x_8}{d^0_8} & \frac{y^{iter}_0-y_8}{d^0_8} & \frac{z^{iter}_0-z_8}{d^0_8}
\end{bmatrix} \tag{22}$$

where $d^0_i=\sqrt{(x_i-x^{iter}_0)^2+(y_i-y^{iter}_0)^2+(z_i-z^{iter}_0)^2}$ corresponding to the range measurement of anchor node i. The vector b in (17), is defined as:

$$b=\begin{bmatrix}
d_1-d^0_1\\
d_2-d^0_2\\
.\\
.\\
d_8-d^0_8
\end{bmatrix} \tag{23}$$

The LS is then used to estimate x^{iter} using equation (18). The state estimation of the specific iteration is calculated as:

$$x^{iter}=\left(A^\top A\right)^{-1}A^\top b \tag{24a}$$

$$\bar{x}^{iter}=x^0_{iter}+x^{iter} \tag{24b}$$

The initial approximation for the next iteration is replaced by the state estimation of the current iteration $\bar{x}^{iter}$ calculated by (24b). The iterations stop either when x^{iter} has a magnitude lower than the pre-defined threshold of $1mm$ or when the number of iterations reach the pre-set value of 200. A threshold of $1mm$ is used because this is an insignificant change in position compared to the target level of accuracy, which is at the decimeter level. Alternatively, the maximum number of iterations is set by observing that the majority of epoches require a maximum of about 15 iterations to calculate the LS position. In order to allow for the calculation of as many positioning epoches as possible, without the calculation getting stuck in an infinite loop, a maximum of 200 iterations is used, which is a significantly larger threshold compared to 15.

Once the iterations stop, the last state estimation $\bar{x}^{iter}$ represents the state of the moving tag for the specific epoch only if the maximum number of iterations has not been reached. If the maximum number of iterations is reached, the specific epoch does not have a position estimate. The initial state approximation for the next epoch is the latest position estimate from previous epoches. Alternatively, when using the multilateration method to estimate the position of a static point, every epoch has the same initial state approximation of $x_0^0 = (0, 0, 0)$. As the moving space is about 5.5m x 8m, an initial state approximation of $x_0^0 = (0, 0, 0)$ is close to any position within that space.

7.1.3. Hyperbolic method

The hyperbolic method used, is similar to the method of time difference of arrival (TDOA), represented in section 5.1.1 by figure 35, where one of the anchors acts as the reference anchor. The difference between the hyperbolic method used and TDOA, is that instead of using the difference between range measurements of two anchors like in TDOA, the difference of the square of the range measurements is used. The basic observation equation of the hyperbolic method used (Feng et al. 2020) is shown in equation (25), where $(x_{ref}, y_{ref}, z_{ref})$ refer to the coordinates of the reference anchor and (x_i, y_i, z_i) are the coordinates of any of the remaining seven anchors. The simplification of this equation

results in the elimination of the quadratic state parameters, illustrating that this method is characterised by a linear system of equations. As opposed to the method of multilateration, since the observation equations of the hyperbolic method are linear, the process of solving for the state of the tag is not iterative and an initial state approximation is not required.

$$\begin{aligned} d_i^2 - d_{ref}^2 = &(x_i - x)^2 + (y_i - y)^2 + (z_i - z)^2 \\ &- \left[(x_{ref} - x)^2 + (y_{ref} - y)^2 + (z_{ref} - z)^2\right] \end{aligned} \tag{25}$$

The design matrix (A) and vector b of the hyperbolic method, are presented by (26) and (27) respectively.

$$A = \begin{bmatrix} x_2 - x_{ref} & y_2 - y_{ref} \\ x_3 - x_{ref} & y_3 - y_{ref} \\ \vdots & \vdots \\ x_n - x_{ref} & y_n - y_{ref} \end{bmatrix} \tag{26}$$

$$b = \frac{1}{2} \begin{bmatrix} x_2^2 + y_2^2 - d_2^2 - \left(x_{ref}^2 + y_{ref}^2 - d_{ref}^2\right) \\ x_3^2 + y_3^2 - d_3^2 - \left(x_{ref}^2 + y_{ref}^2 - d_{ref}^2\right) \\ \vdots \\ x_n^2 + y_n^2 - d_n^2 - \left(x_{ref}^2 + y_{ref}^2 - d_{ref}^2\right) \end{bmatrix} \tag{27}$$

A and b are used in the LS technique to calculate the position estimate of the tag as:

$$\bar{x} = \left(A^\top A\right)^{-1} A^\top b \tag{28}$$

The hyperbolic method requires the use of one anchor as the reference anchor. As there is no specific anchor to be used as the reference, the position of every epoch is estimated eight times and every time a different anchor acts as the reference anchor. Once the eight positions are estimated, the position of the specific epoch is calculated as the average of the eight estimations.

7.1.4. Accuracy performance evaluation

One common way of evaluating the performance of a positioning system in terms of accuracy, is the error in the computed position at the 95^{th} percentile. Another way used in this thesis is the root mean square error (RMSE). RMSE is an alternative way of representing the errors in position, and is calculated as shown in equation (29) for the case of a 2D positioning solution. The accuracy at 95% and the RMSE are both parameters used to evaluate and compare the accuracy performance of the two methods presented in this chapter.

$$\begin{aligned} RMSE =& \sqrt{\sum \frac{(\Delta x^2 + \Delta y^2)}{n}} \\ =& \sqrt{\sum \frac{(x_{est} - x_{true})^2 + (y_{est} - y_{true})^2}{n}} \end{aligned} \tag{29}$$

7.2. Measurement errors

The collected UWB raw data are range measurements between the anchor and the UWB tag. Before calculating the positioning solution, errors of the raw range measurements are analysed, from data collected in both static and dynamic modes. Range measurement errors (d_{er}) are calculated using equation (30), where d_{meas} is the range measurement and d_{true} is the true range. The true range values are calculated using the known coordinates of the anchor and the true position of the UWB tag provided by the camera system at the time of information capture.

$$d_{er} = d_{meas} - d_{true} \tag{30}$$

7.2.1. Error distribution

The collected data are divided into static and dynamic . Dynamic data are collected for paths described in table 6. Static data are collected from the eight positions shown in figure 51, in addition to the static components of the dynamic paths. An initial step in analysing the errors in static and dynamic data, is to visualise the measurement error

distributions derived by the data. The error distribution is approximated using histograms of the measurement errors. Every anchor node has their own error distribution as illustrated in figures 58 to 65. As shown in these figures, distributions of static and dynamic data generally seem to be characterised by multiple and two peaks, respectively.

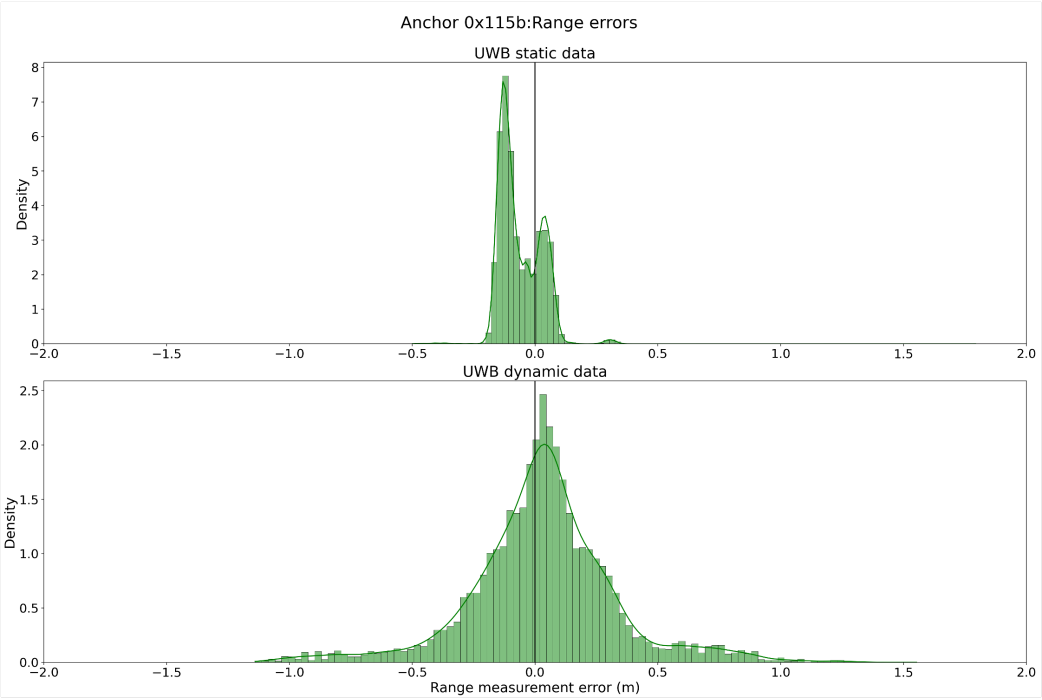


Figure 58: Measurement error distributions from static and dynamic data of anchor 0x115b.

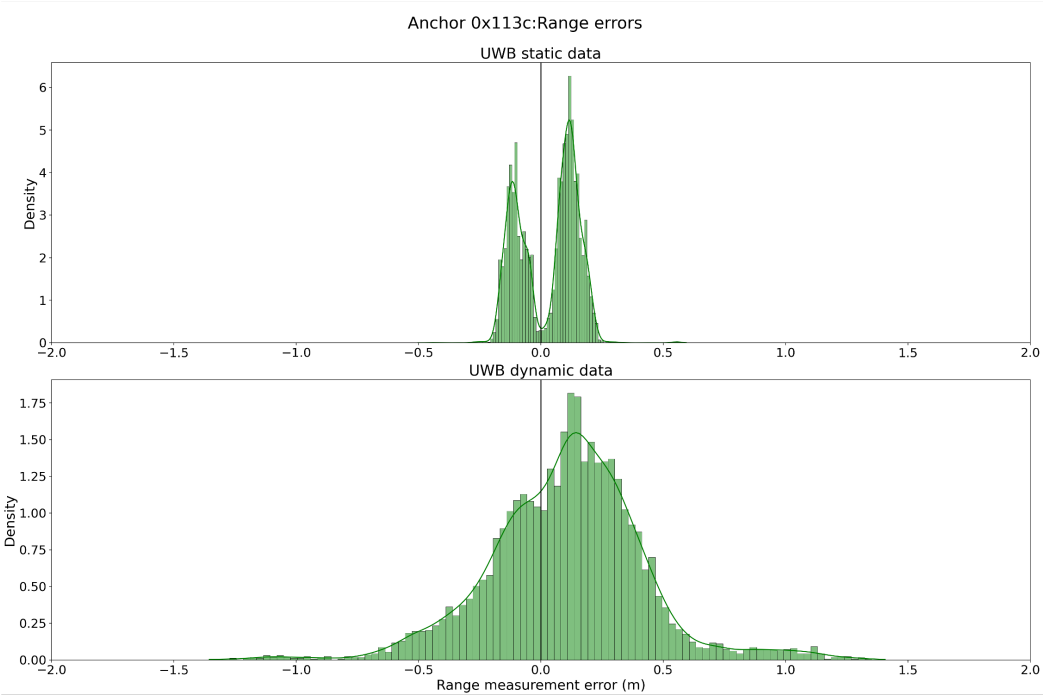


Figure 59: Measurement error distributions from static and dynamic data of anchor 0x113c.

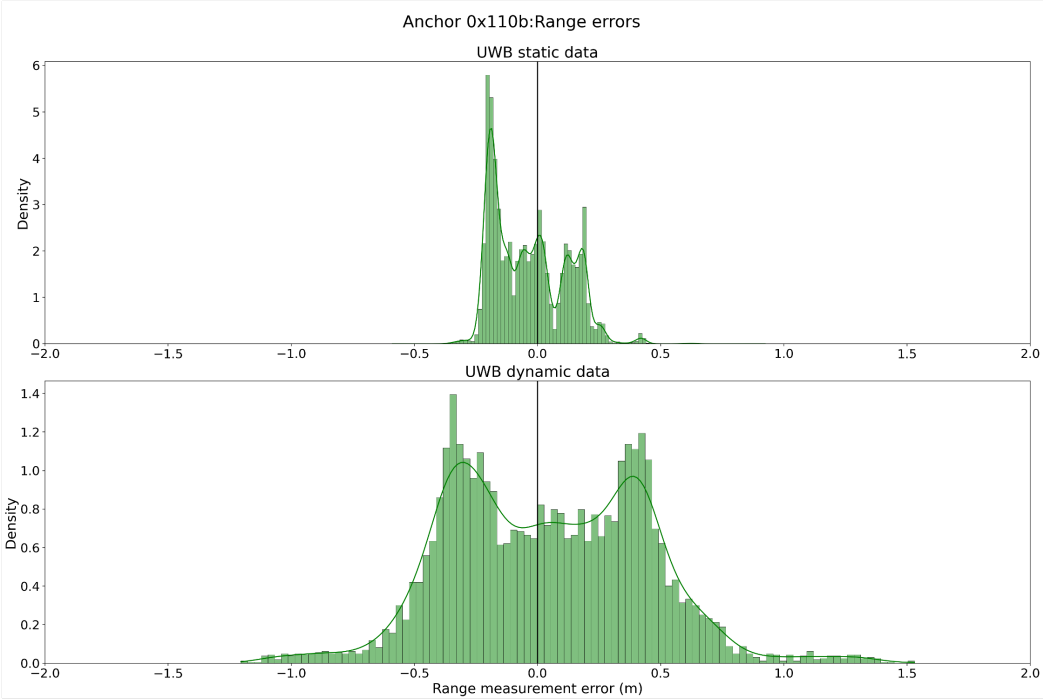


Figure 60: Measurement error distributions from static and dynamic data of anchor 0x110b.

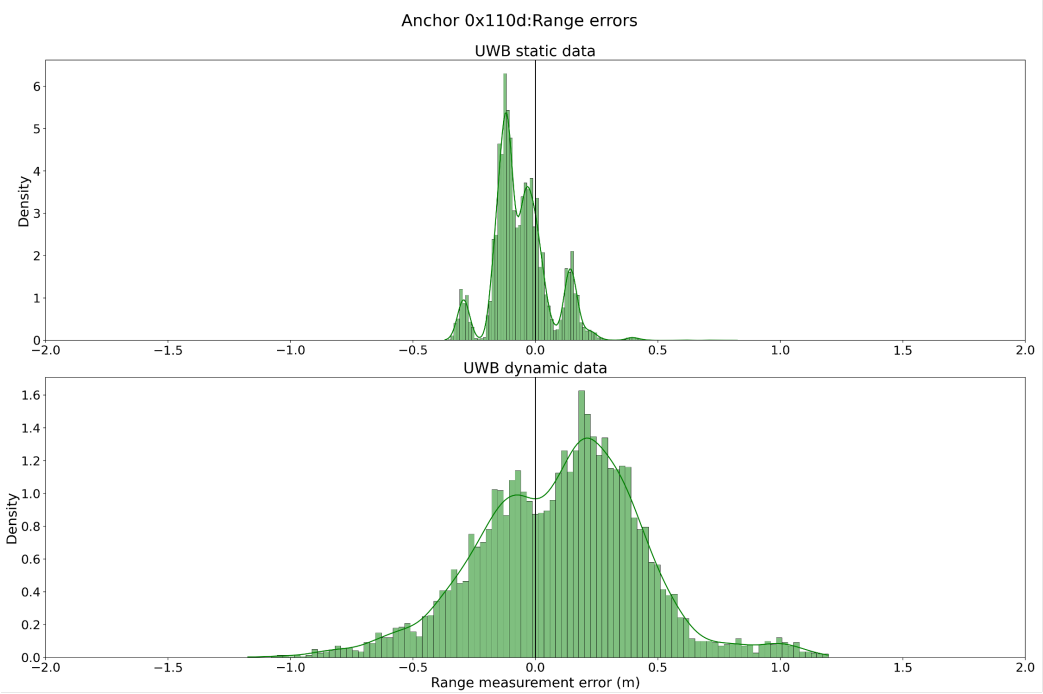


Figure 61: Measurement error distributions from static and dynamic data of anchor 0x110d.

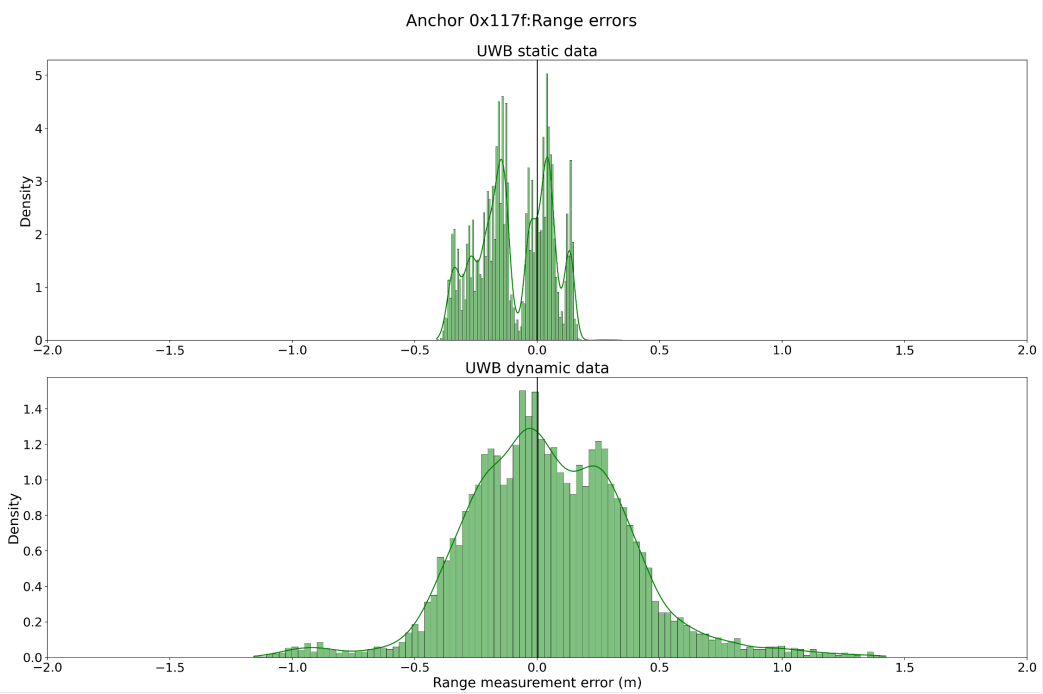


Figure 62: Measurement error distributions from static and dynamic data of anchor 0x117f.

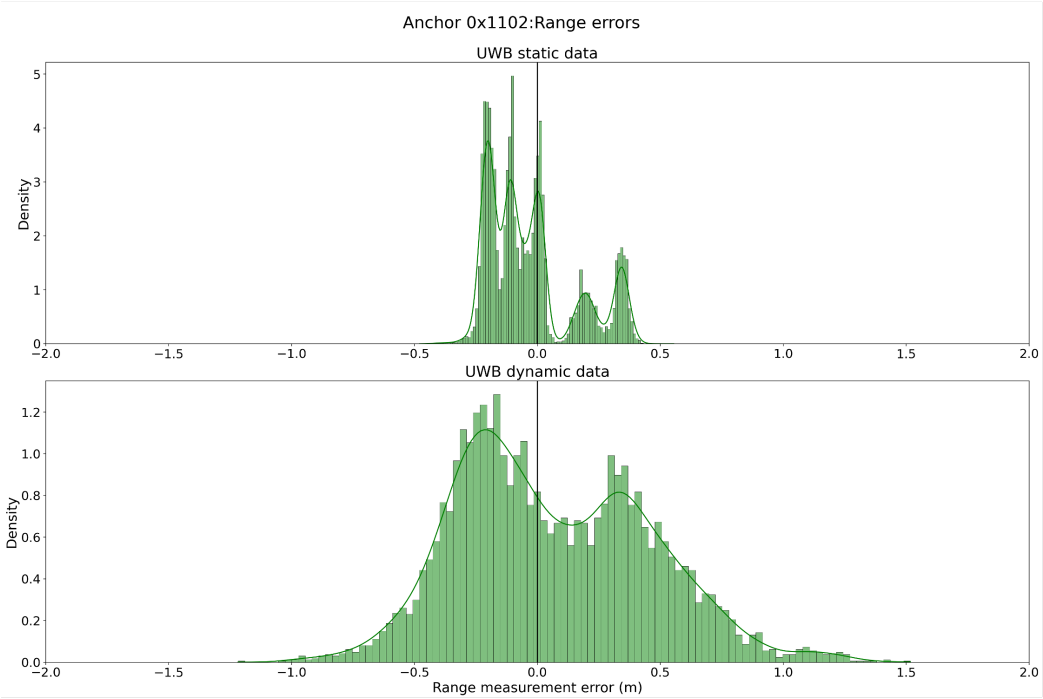


Figure 63: Measurement error distributions from static and dynamic data of anchor 0x1102.

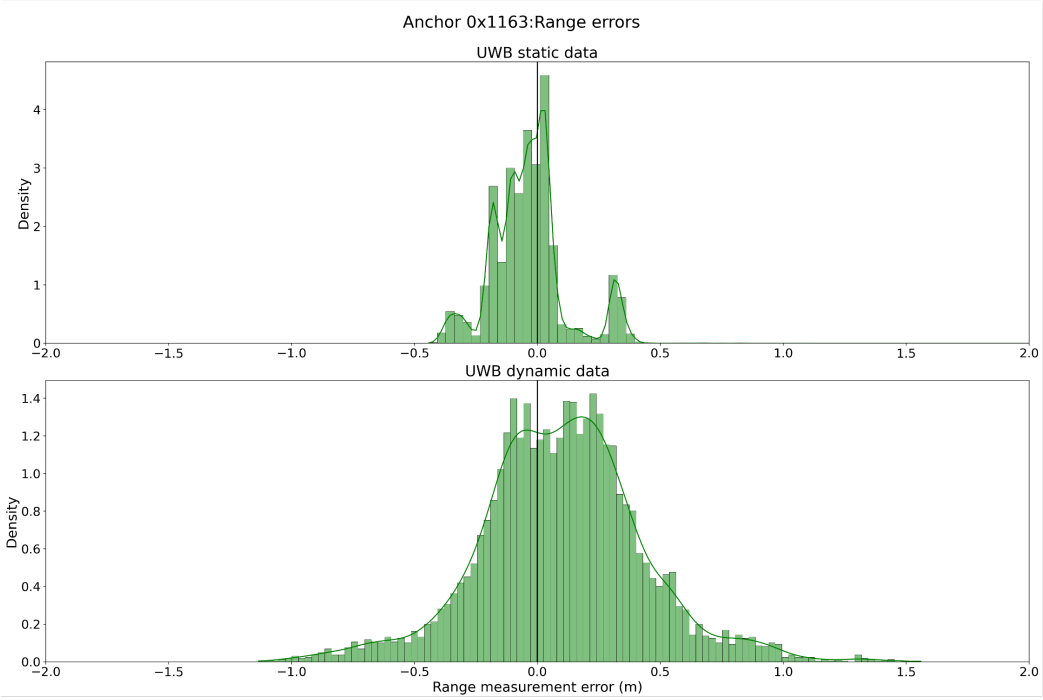


Figure 64: Measurement error distributions from static and dynamic data of anchor 0x1163.

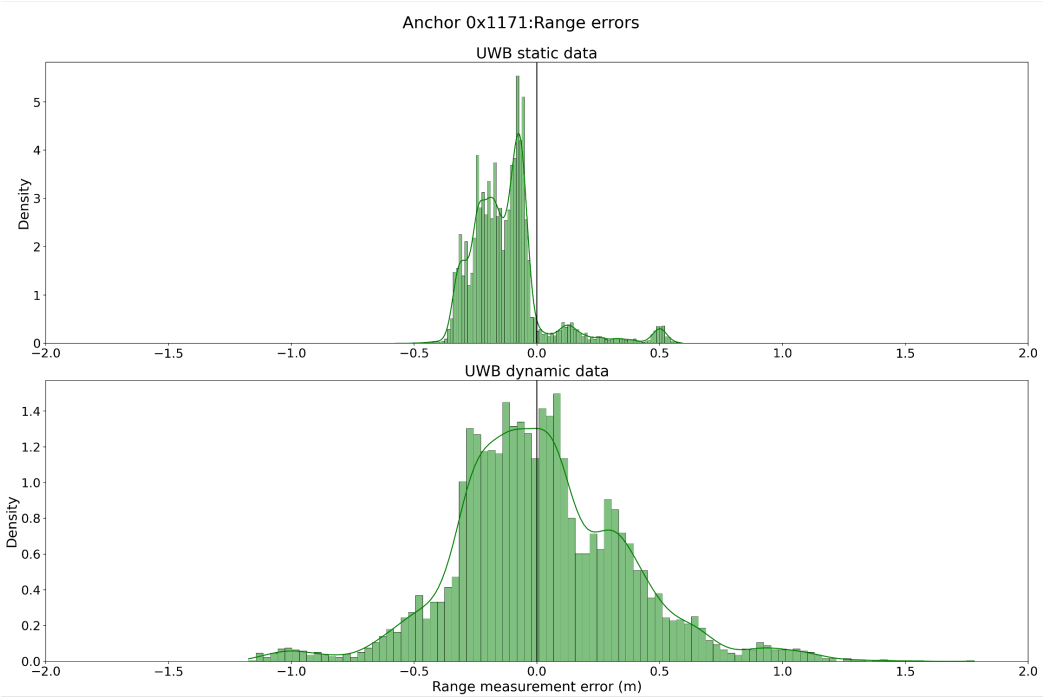


Figure 65: Measurement error distributions from static and dynamic data of anchor 0x1171.

Measurement error distributions in static data illustrate peaks of both positive and negative errors. To further investigate measurement errors in static data, distributions from the eight static positions are illustrated in figures 66 to 73 for every anchor node. These distributions illustrate that for all the anchors, some positions result in biased normally distributed measurement errors, while other positions result in multiple-peak error distributions. Another observation, is that it is not always the same position that causes the non-normal error distribution amongst the anchors. This illustrates measurement errors are not necessarily associated with specific positions around the room. However, there seems to be a dependency of measurement errors on the geometry between the tag and the anchor, signal propagation, or a combination of the two.

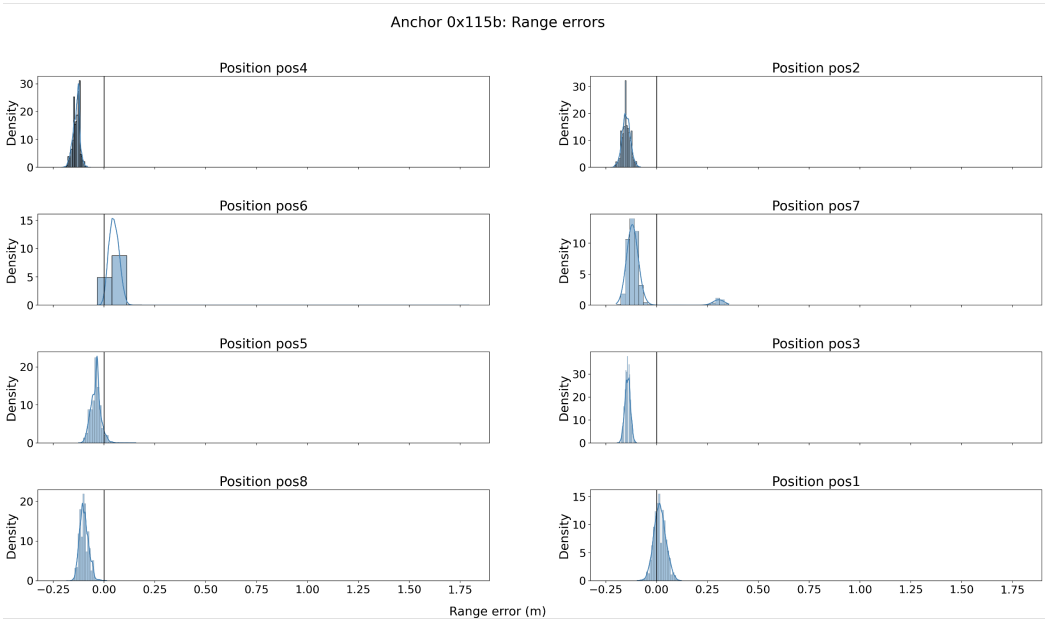


Figure 66: Measurement error distributions from static data of anchor 0x115b.

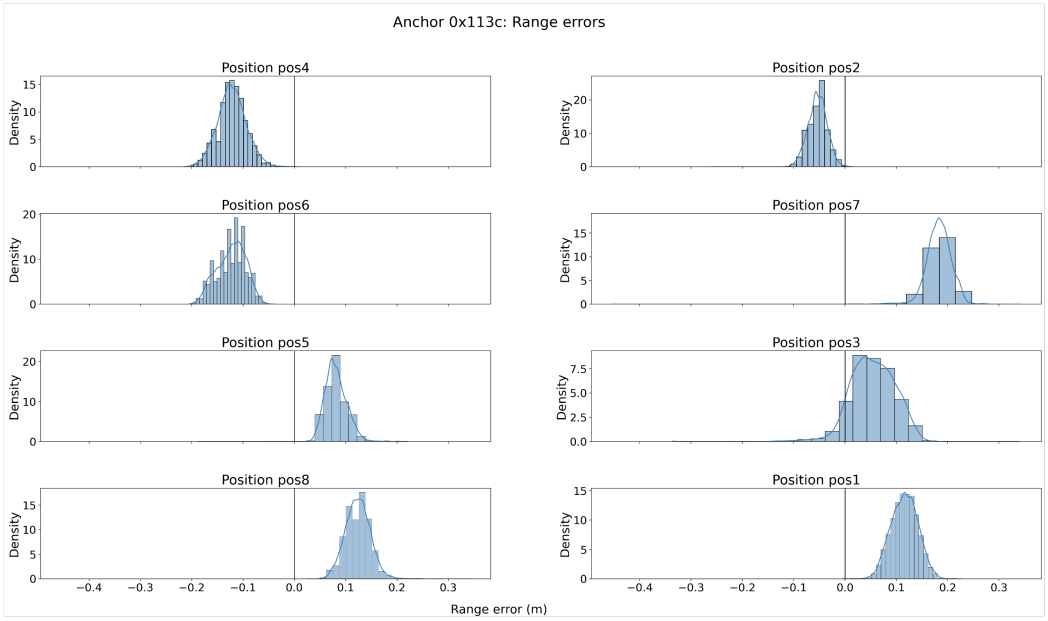


Figure 67: Measurement error distributions from static data of anchor 0x113c.

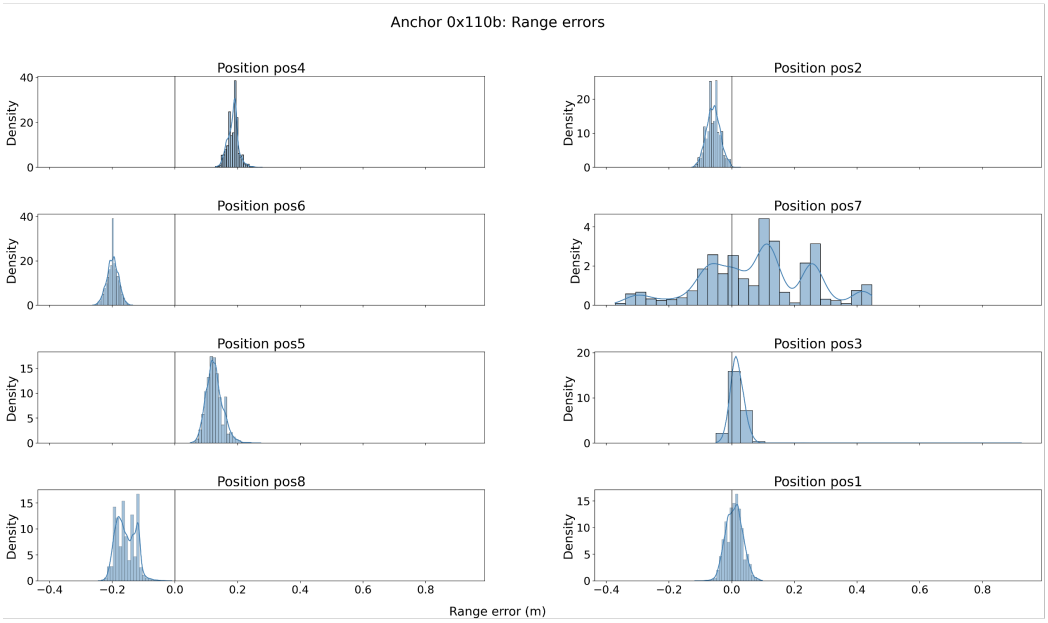


Figure 68: Measurement error distributions from static data of anchor 0x110b.

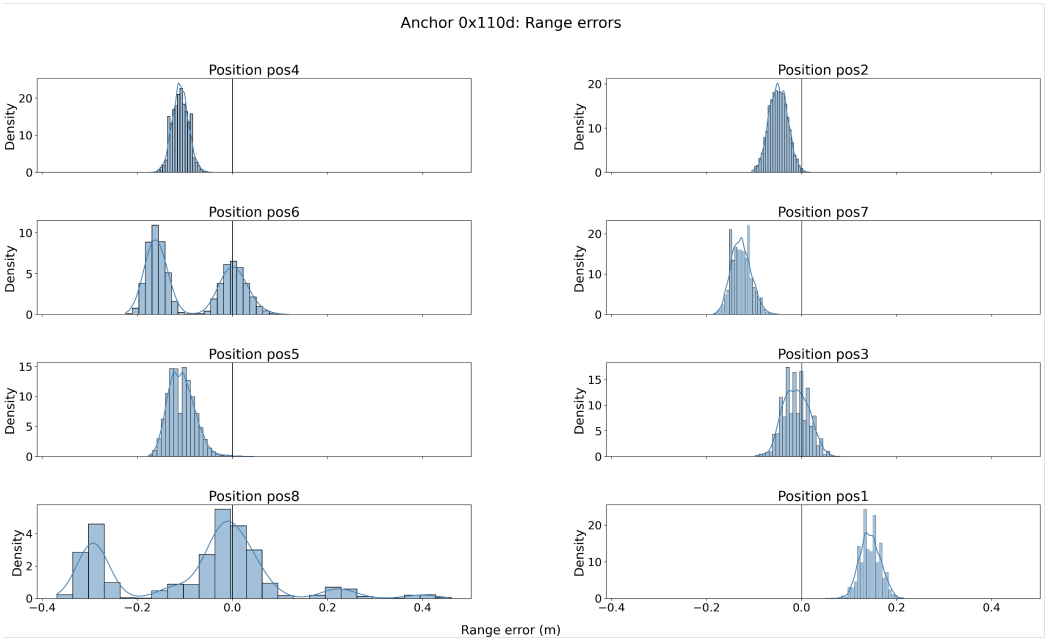


Figure 69: Measurement error distributions from static data of anchor 0x110d.

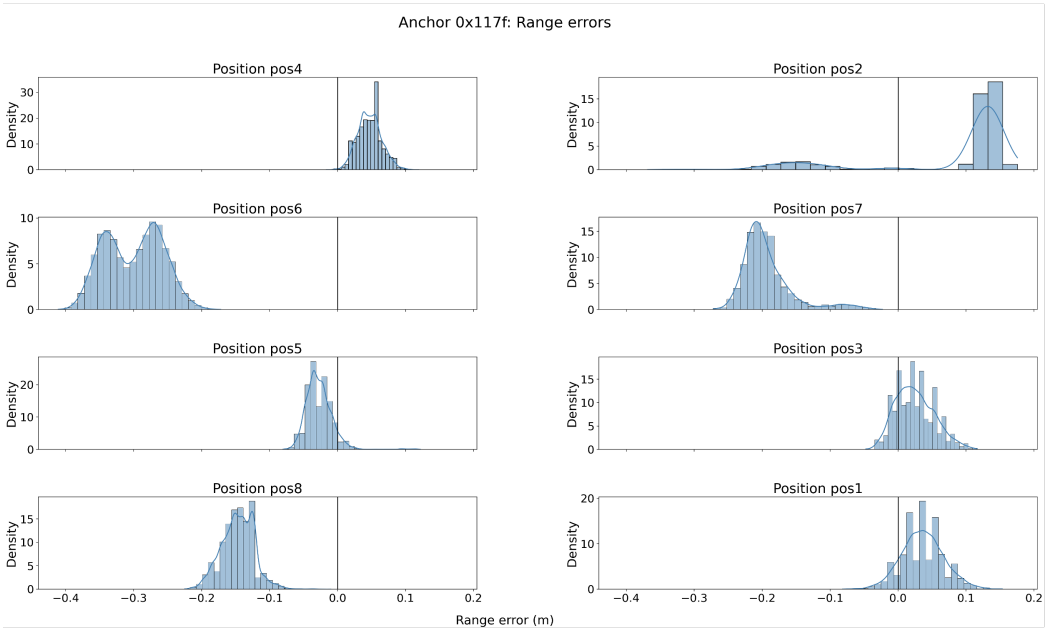


Figure 70: Measurement error distributions from static data of anchor 0x117f.

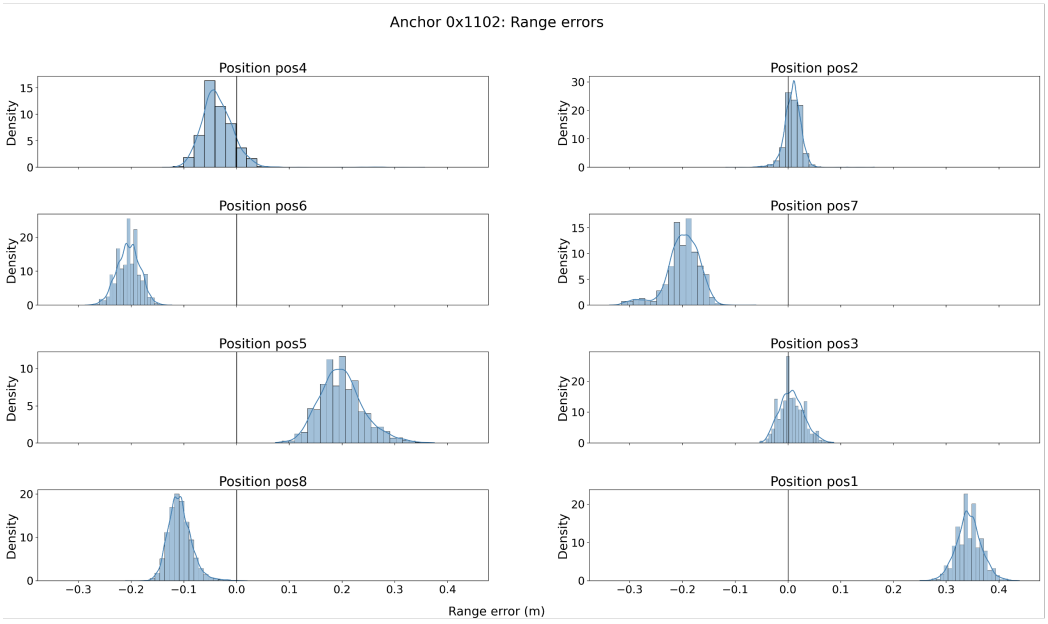


Figure 71: Measurement error distributions from static data of anchor 0x1102.

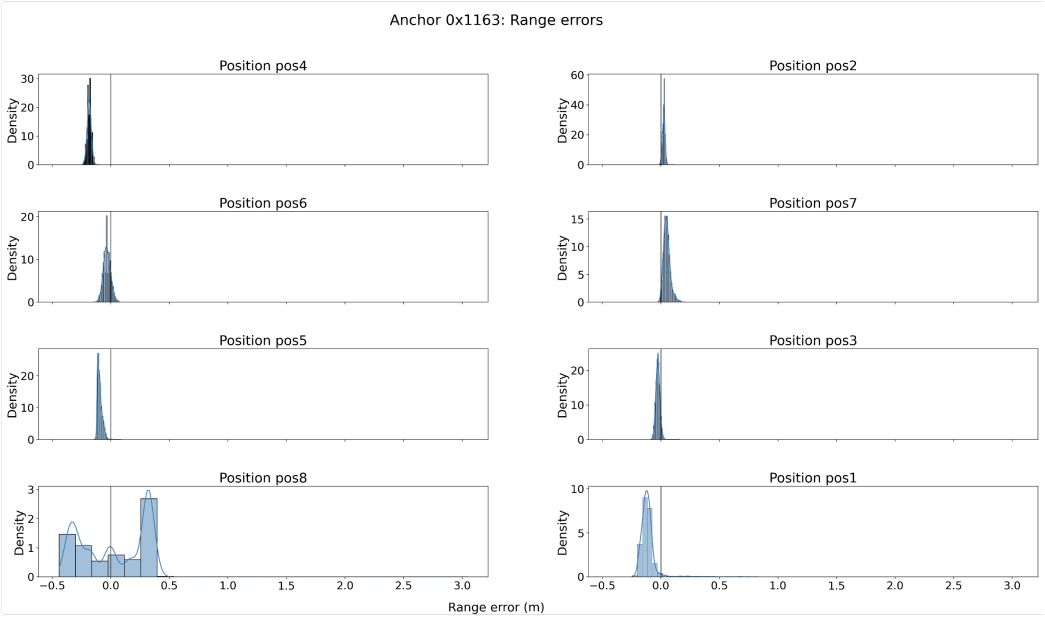


Figure 72: Measurement error distributions from static data of anchor 0x1163.

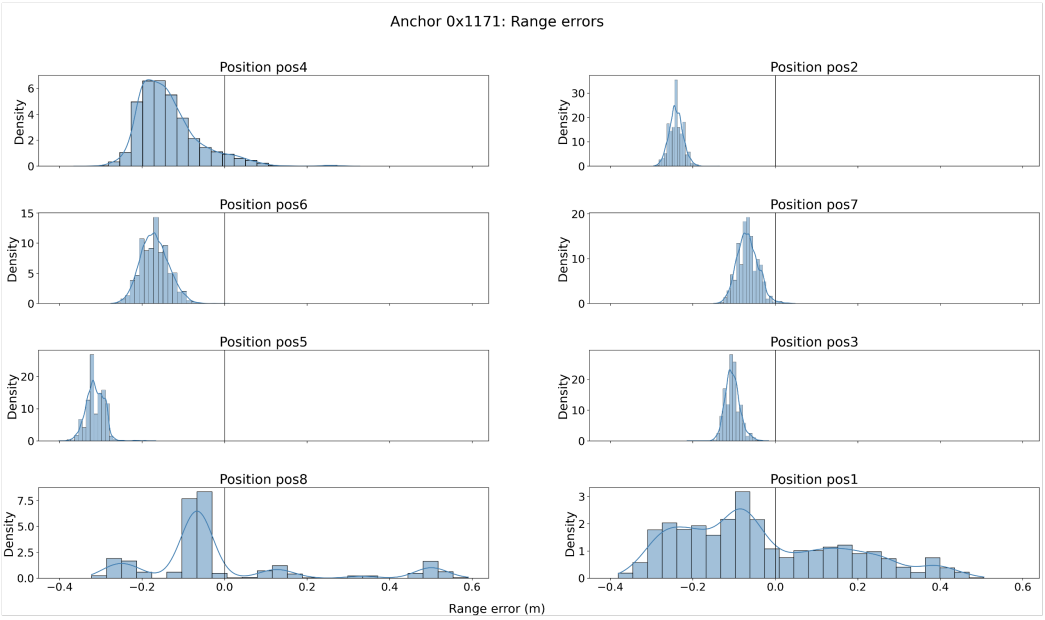


Figure 73: Measurement error distributions from static data of anchor 0x1171.

An additional interesting observation is the existence of negative measurement errors. Even though positive measurement errors can be related to NLOS signal propagation or

multipath effects, negative errors do not have a clear source in a static mode. The next section investigates measurement errors in dynamic mode.

7.2.2. Measurement errors of dynamic mode

As shown in figures 58 to 65, measurement error distributions in dynamic modes are generally characterised by two peaks. To investigate the reason of these two peaks, colour-coded maps of measurement errors along the dynamic paths are visualised for every anchor and every path. These are used as heat-maps to identify potential reasons for the two-peaked distributions of measurement errors. Figure 74 presents the heat-maps of measurement errors for three distinct paths and two anchors, for purposes of illustration. The three paths are selected to illustrate the consistency of the observations amongst different path types, namely, straight lines, random paths and circular paths. Similarly, the two anchors illustrate the consistency of observations across the anchors.

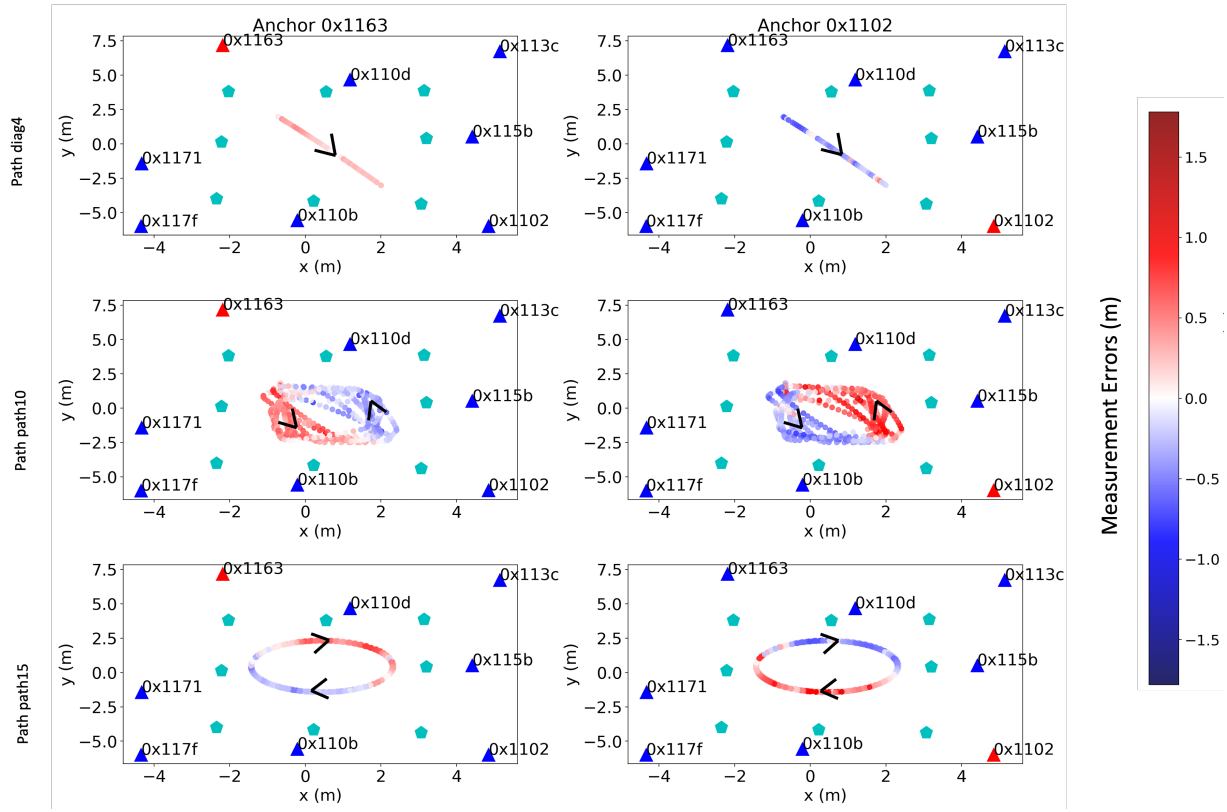


Figure 74: Colour-coded measurement errors of two different anchors for three paths. The three paths are a straight line (diag4), a random path (path10) and a circular path (path15). In every figure, the relevant anchor is highlighted in red and the arrows illustrate the direction of motion.

Observing all the figures, a clear pattern can be seen in all the path types and across all anchors. For the part of the trajectory where the dynamic platform is moving towards the anchor, the measurement errors are negative, alternatively, as the dynamic platform is moving away from the anchor the measurement errors become positive. This is evident from figure 74, where as the dynamic platform moves in the direction illustrated by the arrows, the measurement errors turn from negative, as the robot approaches the anchor, to positive, as the robot is moving away from the anchor.

The dependency of measurement errors on the direction of motion with respect to the anchor leads to further investigation of measurement error distributions. The dynamic

data are classified into two groups, according to the direction of motion with respect to the anchor. One of the groups include measurement when the dynamic platform is moving towards the anchor, and the other group includes measurements when the motion is away from from the anchor. Figures 75 to 82 illustrate the distribution of measurement errors as the dynamic platform is moving towards and away from the anchor. The distributions of the two directions when combined together, result in the two-peak distributions of measurement errors. This can be seen in figures 75 to 82, where the distributions of the errors when the platform is moving towards the anchor (the blue distribution) and when the platform is moving away from the anchor (the red distribution), are characterised by distinct peaks corresponding to the two peaks of all the dynamic data (the grey distribution).

These figures illustrate that depending on the direction of motion with respect to the anchor, the measurement error is likely to be biased by a different amount, corresponding to the two distinct peaks. The bias of the measurements is calculated as the mean of the measurement errors for each direction of motion. The calculated measurement bias, which corresponds to the mean of the errors (μ_{twrds}, μ_{away}), and standard deviation ($\sigma_{twrds}, \sigma_{away}$) are shown in table 8 and have a different value depending on the direction of motion with respect to the anchor. Figures 75 to 82 show the mean of both both distributions using the solid blue and red line, as well as the cut-off at $\mu + / - 3\sigma$ which seems to include the majority of the distribution.

An interesting thing to note is that even though the standard deviations of both directions of motion are similar, the magnitude of the mean values tend to be larger when the platform is moving away from the anchor, compared to when the platform is moving towards the anchor. The calculated biases and standard deviations are used later on in the integration of the INS and UWB system to correct the UWB measurements according to the direction of motion.

Table 8: The mean and standard deviation of measurement errors when the robot is moving towards and away from the anchor.

Anchor	Moving towards anchor		Moving away from anchor	
	$\mu_{twrds}(m)$	$\sigma_{twrds}(m)$	$\mu_{away}(m)$	$\sigma_{away}(m)$
0×115b	-0.050	0.281	0.126	0.298
0×113c	-0.005	0.312	0.197	0.294
0×110d	-0.045	0.314	0.226	0.298
0×1163	-0.052	0.300	0.225	0.307
0×117f	-0.116	0.273	0.230	0.303
0×1102	-0.133	0.299	0.298	0.364
0×1171	-0.136	0.299	0.159	0.341
0×110b	-0.176	0.320	0.288	0.336

This dependency of measurement errors on the direction of motion with respect to the anchor, might be caused by the way range measurements are obtained. Range measurements are calculated using TWR, which as mentioned in section 5.1.1 uses the ToF information of the round trip of the signal, which includes a slight delay at the receiver for sending a reply. A potential reason for this dependency on the direction of motion, might be that during the round trip of the signal, if the robot is moving either towards or away from the anchor, this might cause respectively shorter or longer ToF calculations than the truth. A shorter or longer ToF calculation than the truth, will result in a respectively negative and positive error. A dependency of measurement errors on direction of motion might allude on another dependency of measurement errors on the speed of motion. The next section investigates the correlation between measurement errors and speed of motion.

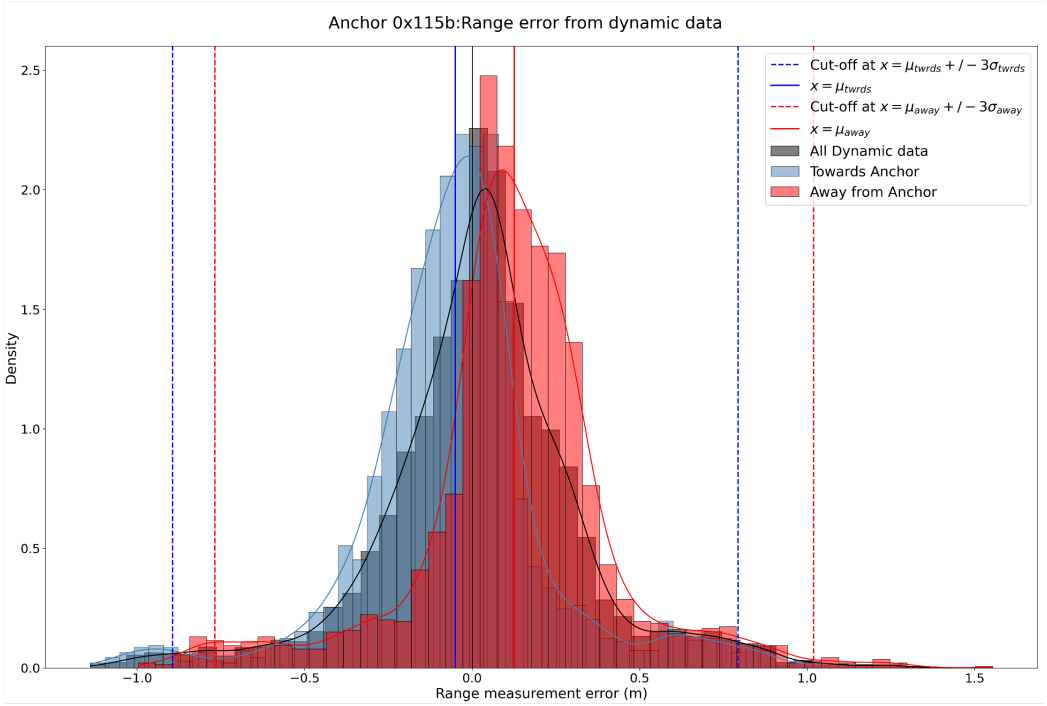


Figure 75: Measurement error distributions of dynamic data for anchor 0x115b.

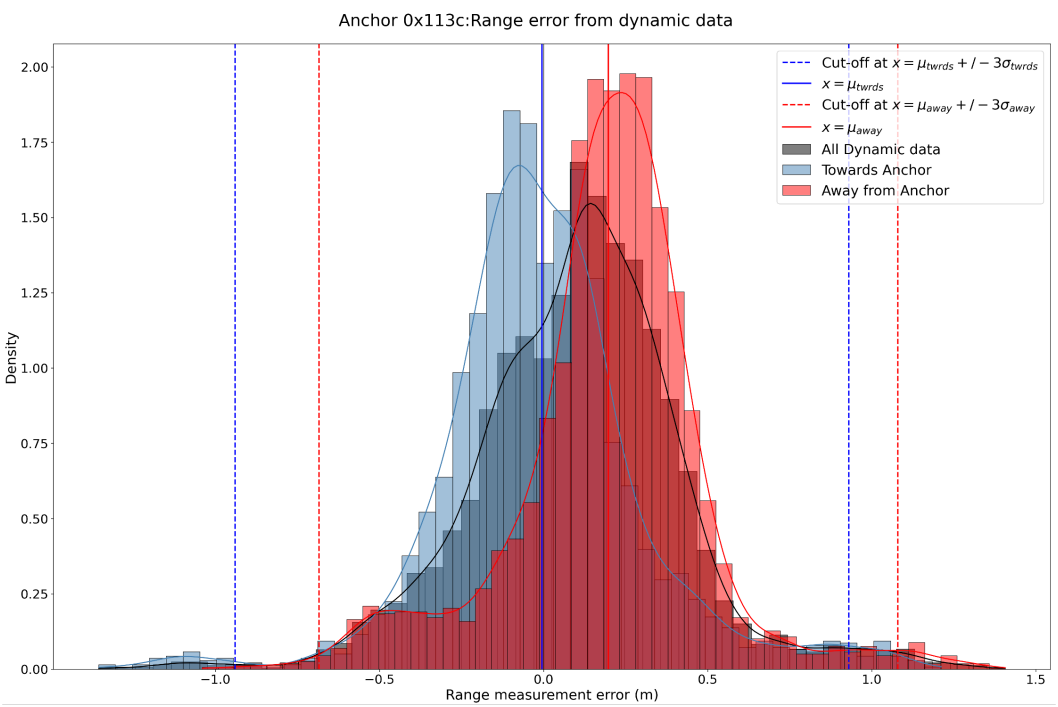


Figure 76: Measurement error distributions of dynamic data for anchor 0x113c.

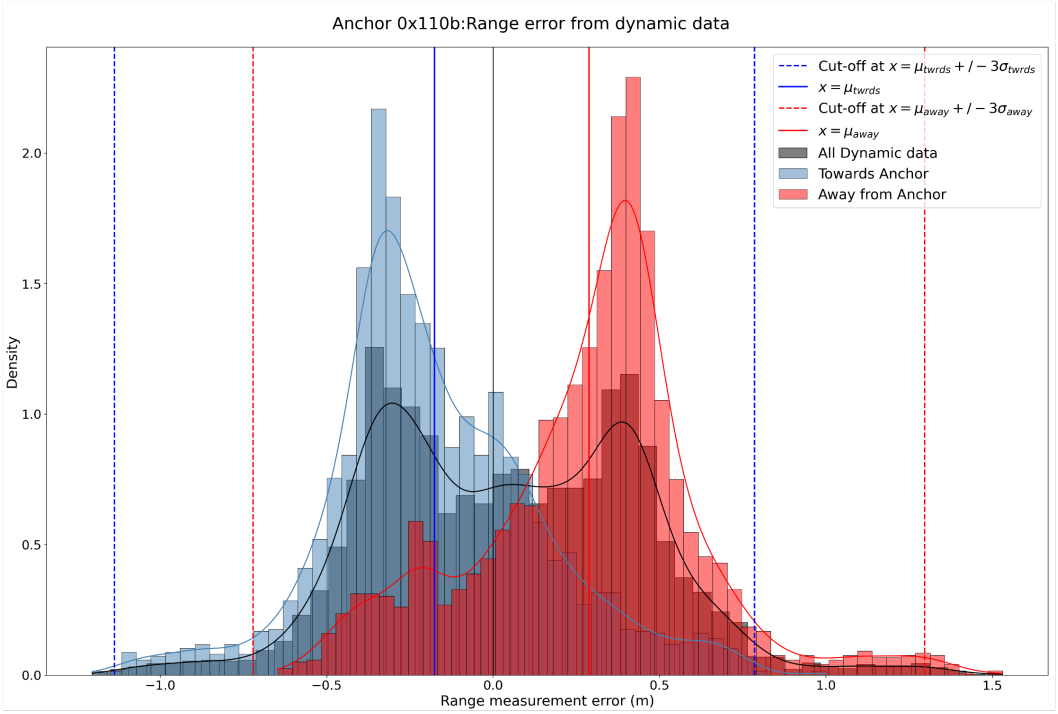


Figure 77: Measurement error distributions of dynamic data for anchor 0x110b.

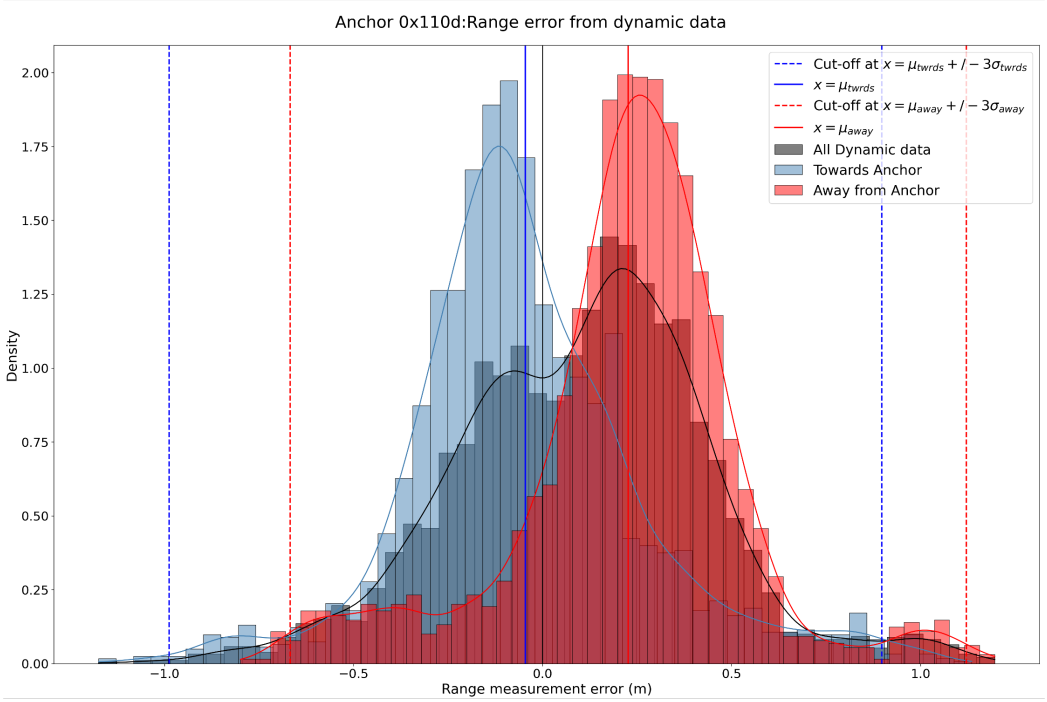


Figure 78: Measurement error distributions of dynamic data for anchor 0x110d.

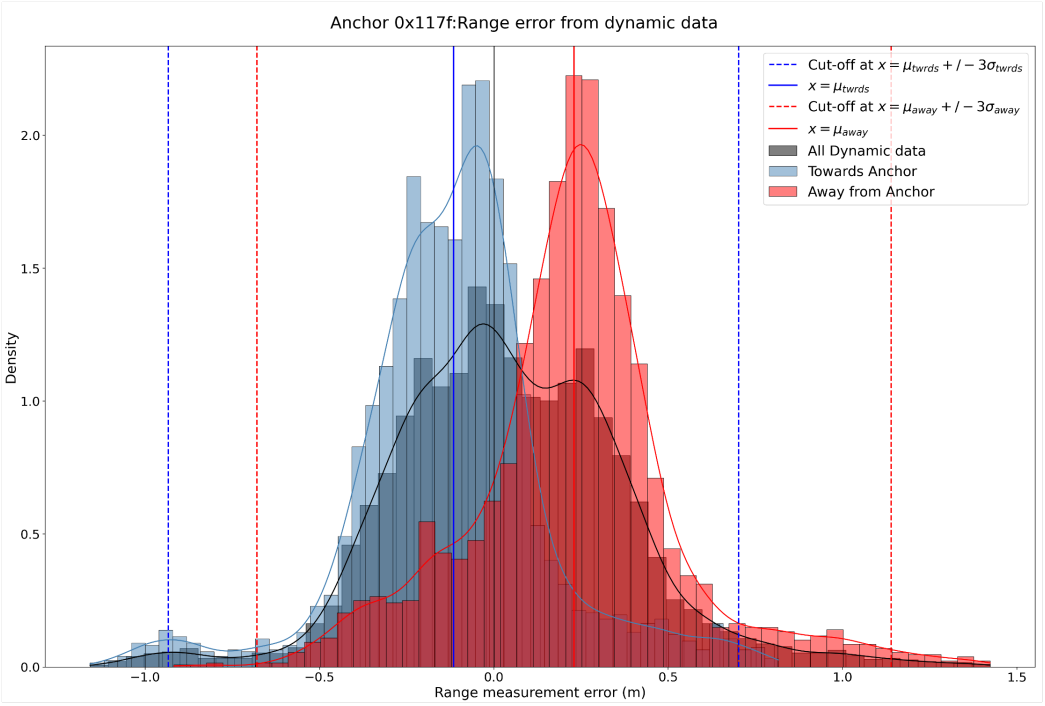


Figure 79: Measurement error distributions of dynamic data for anchor 0x117f.

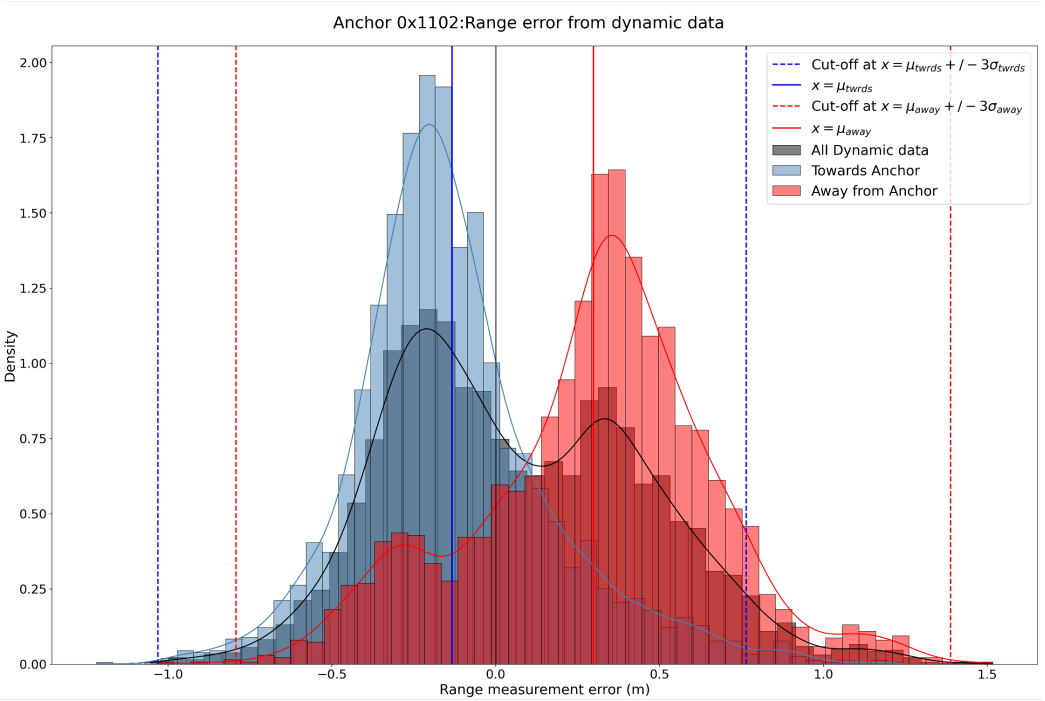


Figure 80: Measurement error distributions of dynamic data for anchor 0x1102.

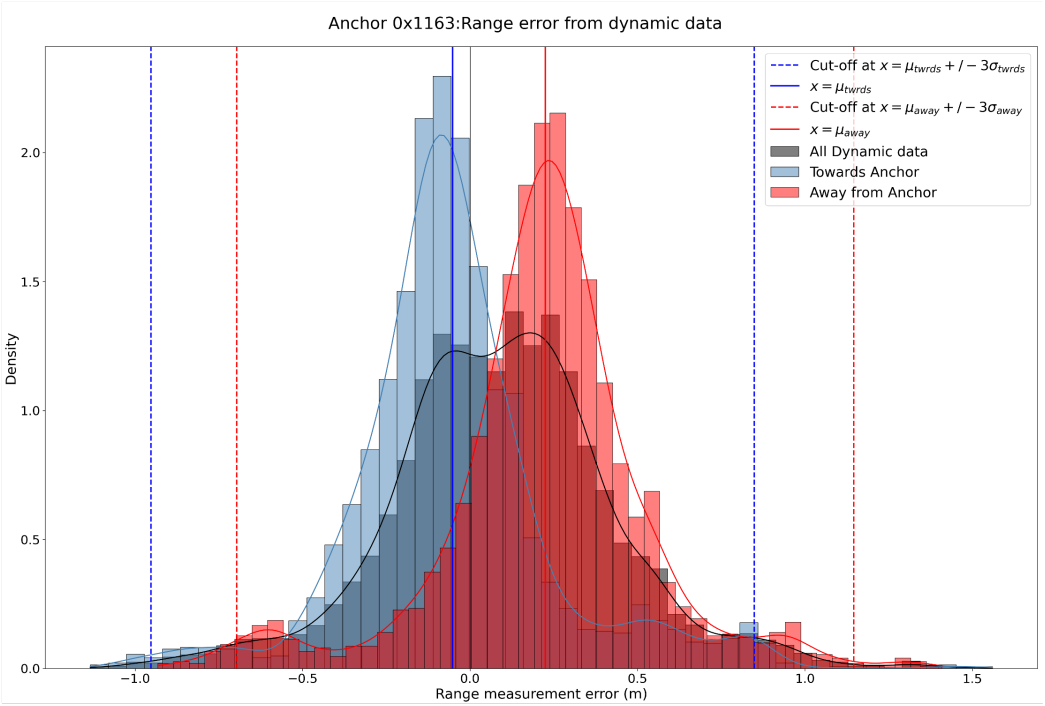


Figure 81: Measurement error distributions of dynamic data for anchor 0x1163.

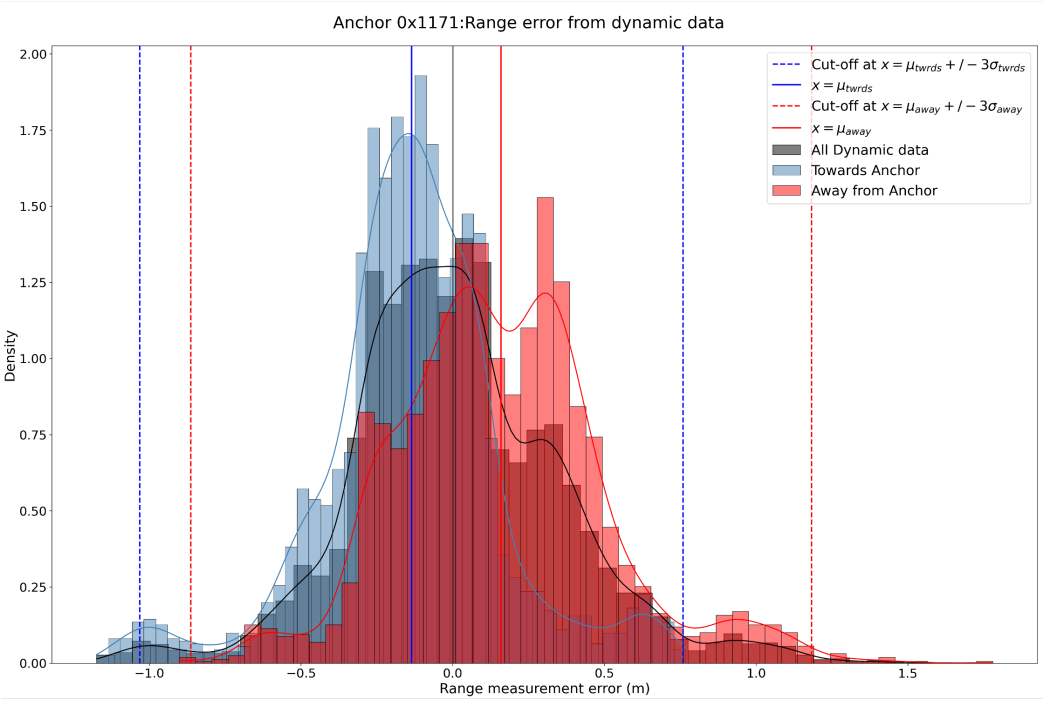


Figure 82: Measurement error distributions of dynamic data for anchor 0x1171.

7.2.3. Range error dependency on speed of motion

This section investigates whether there is any dependency of range measurement errors on speed of motion. In order to ensure the investigation of the dependency on speed rather than velocity, this section focuses only on data collected from the paths where the robot is moving in a straight line. In this way, any potential impact of changes in the direction of motion on the errors, is eliminated. As shown in table 6, the straight line paths are classified into three groups according to their speed, while all straight lines follow almost the same path, as shown in figure 83. The first group of straight lines includes paths 'diag1', 'diag2' and 'diag3', where the robot was moving at a speed of about $0.181m/s$, the second group includes paths 'diag4' and 'diag5' at a speed of about $0.386m/s$, and the third group includes paths 'diag7', 'diag8' and 'diag9' at a speed of about $0.568m/s$. The similar path followed by all straight lines and the three different speed groups allow for a comparison of measurement errors with speed.

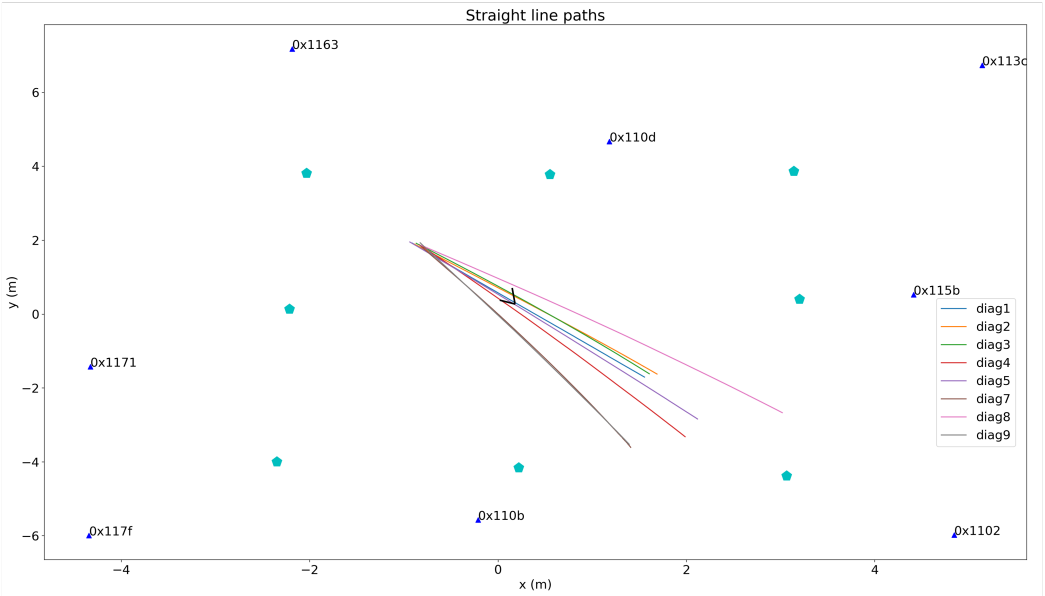


Figure 83: The straight line dynamic paths.

An initial step in investigating the dependency of measurement errors on speed of motion is the visualisation of measurement errors with respect to the distance from the

anchor. A figure illustrating the variation of measurement error with distance from the anchor, for all the straight lines, is produced for every anchor. An example is illustrated in figure 84 for anchor node $0x1163$. As can be seen from figure 84, every column includes the straight line paths that share the same speed of motion. An observation made for some of the anchors, is that the measurement errors against distance from the anchor relationships, have a similar trend across the straight line paths. A common observation is that increasing or decreasing trends of measurement error with distance from the anchor, vary in a linear way. An additional observation is that the linear trends are interrupted by sudden and large deviations in errors, marked by the red lines in figure 84. The large deviations could be related to surrounding environmental conditions, as they occur at similar distances from the anchor.

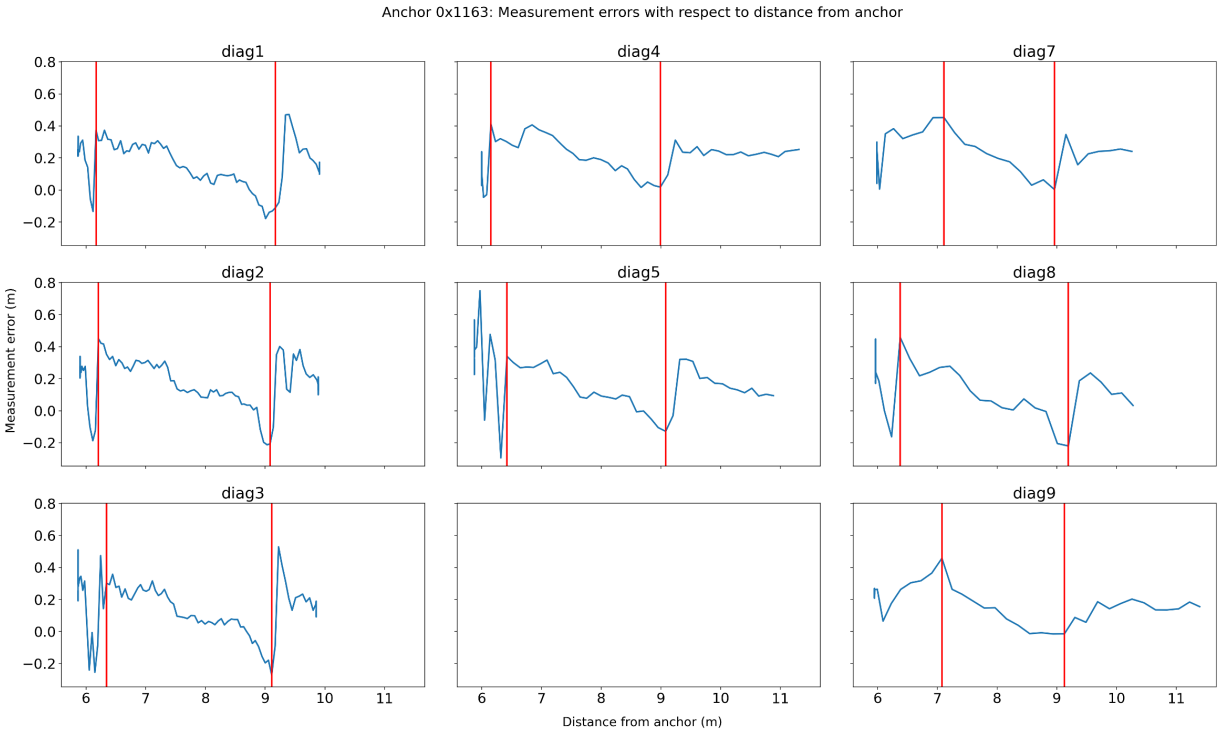


Figure 84: Measurement errors against distance from the anchor for anchor 0x1163. The red lines illustrate the large deviations of the trend of measurement errors with distance from the anchor

A closer observation of the linear trends of some of the anchors, like the one shown in figure 84, illustrate that the gradient of the linear trend, has a dependency on the speed of motion. The gradient of the linear trend represents the rate of change of measurement errors with respect to distance from the anchor. To further investigate this observation, the gradients of the linear trends of all the straight line paths, are calculated for every anchor. Pearson correlation is then used to determine whether there is any correlation between the gradients and speed of motion.

The anchors that result in a strong correlation between the gradients and speed of motion are found in the corners of the room, anchors $0x1163$, $0x113c$, $0x1102$ and $0x117f$. The measurement errors against distance from the anchor, together with the calculated linear trend of all the straight lines paths, for these four anchors are illustrated in figures 85 to 88. It is interesting to note that the straight line path 'diag8' does not provide any trends for anchors $0x113c$ and $0x117f$ as the distance of the dynamic platform with respect to the these anchors does not change significantly along 'diag8'. As a result, 'diag8' is not used when calculating the correlation between the gradient of measurement error and speed of motion for these two anchors.

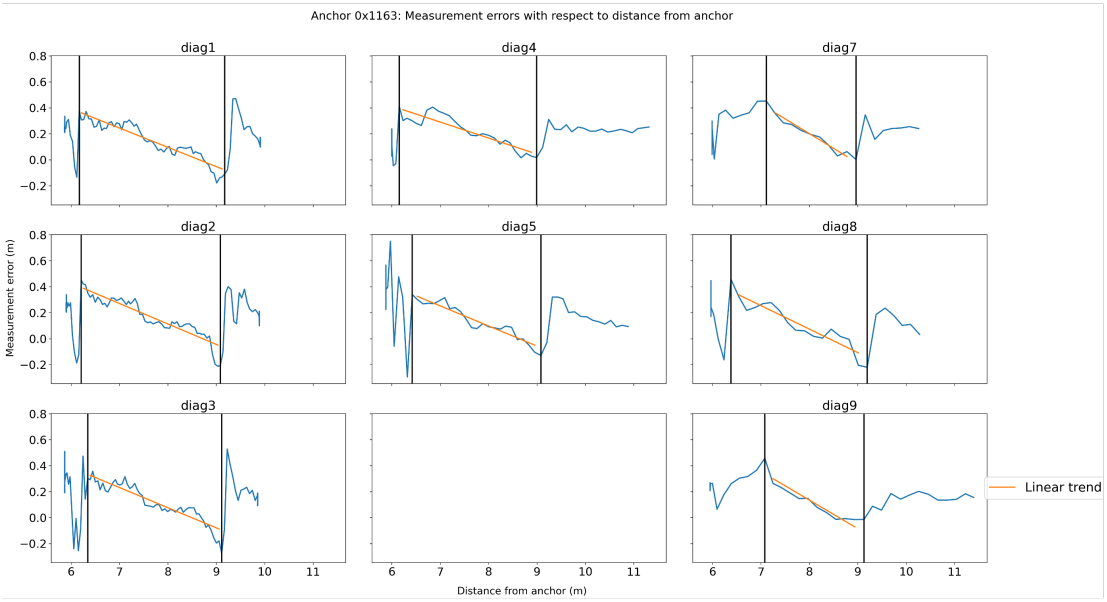


Figure 85: Measurement errors against distance from the anchor for anchor 0x1163, with the linear trend for every path.

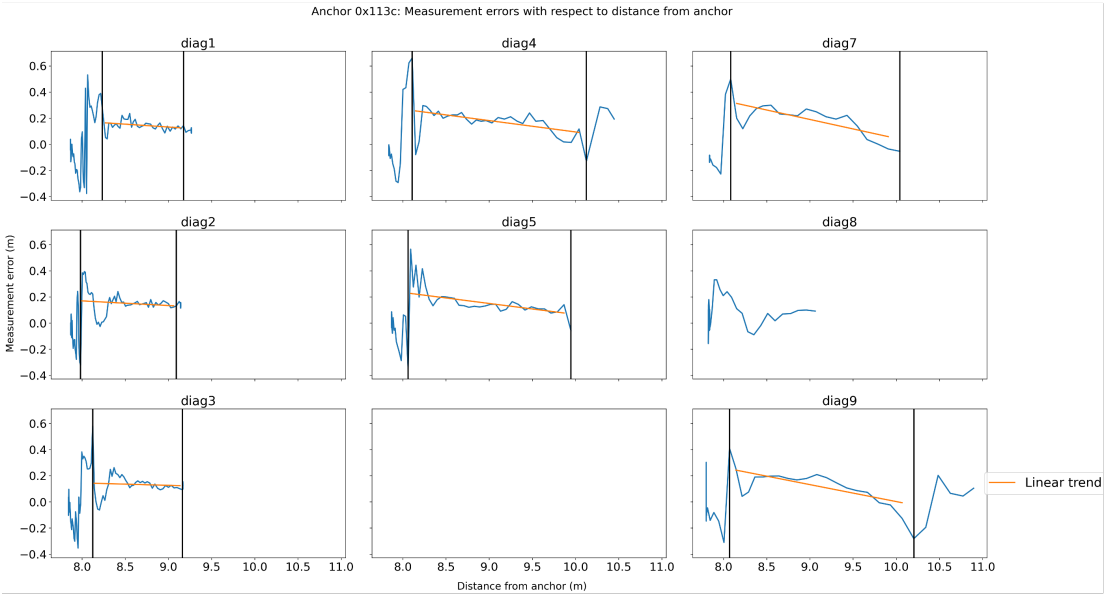


Figure 86: Measurement errors against distance from the anchor for anchor 0x113c, with the linear trend for every path.

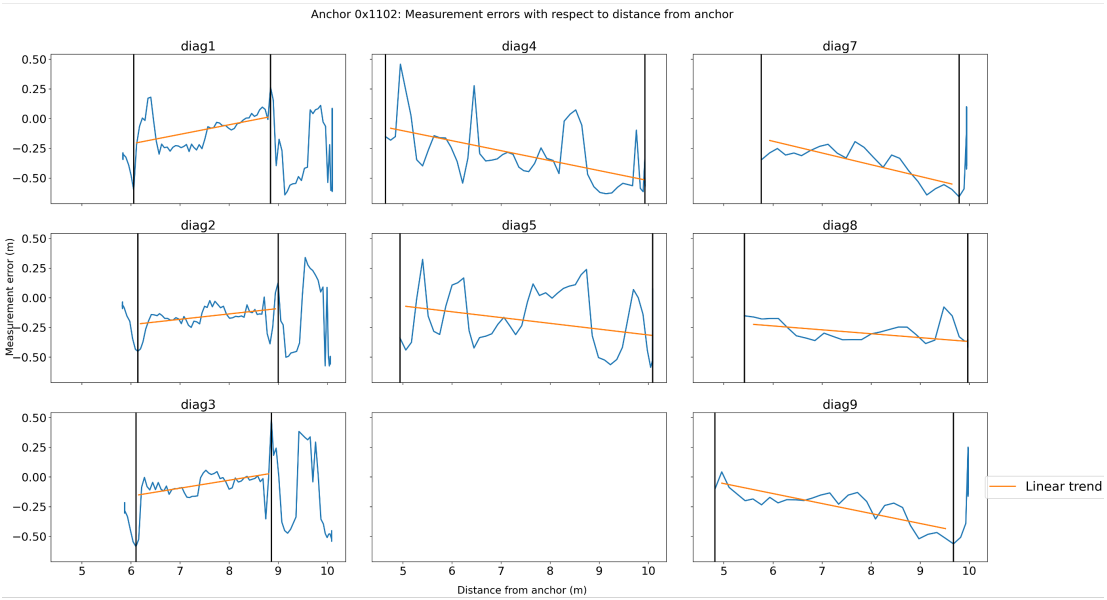


Figure 87: Measurement errors against distance from the anchor for anchor 0x1102, with the linear trend for every path.

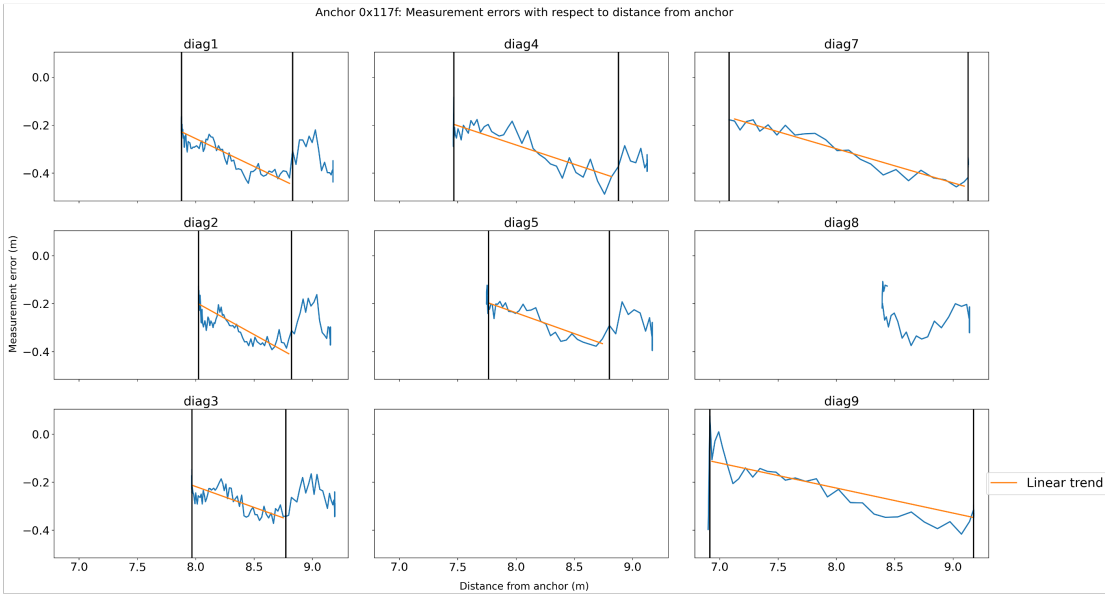
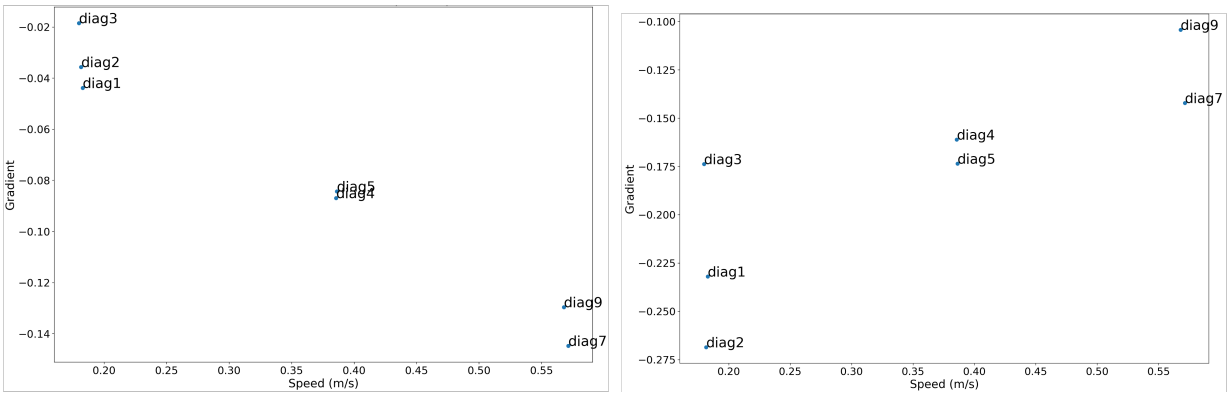


Figure 88: Measurement errors against distance from the anchor for anchor 0x117f, with the linear trend for every path.

Figure 89, illustrate the gradients of the straight line paths against the speed of motion

for two of the anchors. The calculated correlations between the gradients and the speed of motion are illustrated in table 9, showing a high correlation between the rate of change of measurement errors with respect to distance from the anchor and speed of motion. In addition, it is interesting to note that the correlation can be either negative or positive. As absolute errors are not used, this simply means that the rate of change of measurement error with respect to distance from the anchor becomes larger with increasing speeds.



(a) Anchor: 0x113c

(b) Anchor: 0x117f

Figure 89: The gradients of the linear trends with respect to the speed of motion.

Table 9: The correlation coefficients between the gradient of measurement errors and speed of motion. Calculated using Pearson correlation

Anchors	Correlation coefficient
0×1163	-0.701
$0 \times 113c$	-0.984
0×1102	-0.876
$0 \times 117f$	0.835

The high rate of change of measurement error with distance suggests that at higher speeds, the errors deviate from zero at a higher rate, resulting in larger measurement errors. The high correlation values observed suggest that the measurement errors have a

dependency on speed of motion. However, since this phenomenon is not observed for all the anchors in the room, this might be an indication that measurement errors depend on the speed of motion, as well as the geometry between the anchor and the trajectory.

7.2.4. Measurement corrections

The analysis of dynamic measurements in sections 7.2.2 and 7.2.3, showed dependencies of measurement errors on the direction of motion with respect to the anchor and speed of motion, respectively. Although, the latter dependency has been observed, a relationship between measurement errors and speed of motion was not formulated. This is primarily because there likely is a further geometric dependency of the way measurement errors vary with speed. This is evident in section 7.2.3, as the strong correlation between measurement errors and speed of motion was observed only from the corner anchors (related to 2D geometry), which were also the ones attached close to the ceiling of the room (related to elevation). UWB measurement corrections depending on the speed of motion are hence not applied in this thesis, however, further investigation in this relationship is thought to be beneficial.

The dependency of measurement errors on direction of motion with respect to the anchor was presented in section 7.2.2. Corrections related to direction of motion were presented in table 8. In order for these measurement corrections to be applied, it is essential to know the direction of motion with respect to the anchor, which requires the use of an orientation sensor. As a result these corrections are applied in chapter 9, where the integration of the INS and UWB is performed. The orientation provided from the INS is used to determine the correction to apply on the UWB measurements.

The remaining of this chapter, uses the uncorrected UWB measurements, since there is no sense of direction of motion, to correct the UWB measurements. Multilateration and hyperbolic methods, discussed in section 7.1, are applied and their performance is evaluated and compared using accuracy metrics.

7.3. Position estimation based on collected data

Following the analysis of the range measurements, the UWB-base position is estimated. The two methods presented in section 7.1 are used to calculate the positioning solution using the UWB measurements. The positioning solutions of both methods, are illustrated in figures 90 to 92 for three dynamic cases, representing the three different path types. In addition to the positioning solution, 2D position error distributions of the same three cases are illustrated in figures 93 to 95. A visual observation of the positioning solutions in figures 90 to 92 and the error distributions figures 93 to 95, show that the results of the two methods are not significantly different. The positioning solutions and position error distributions of both methods provide comparable results, however, the error distributions of multilateration tend to be narrower compared to the hyperbolic method. The comparison of the two methods is further investigated by a numerical evaluation of their errors.

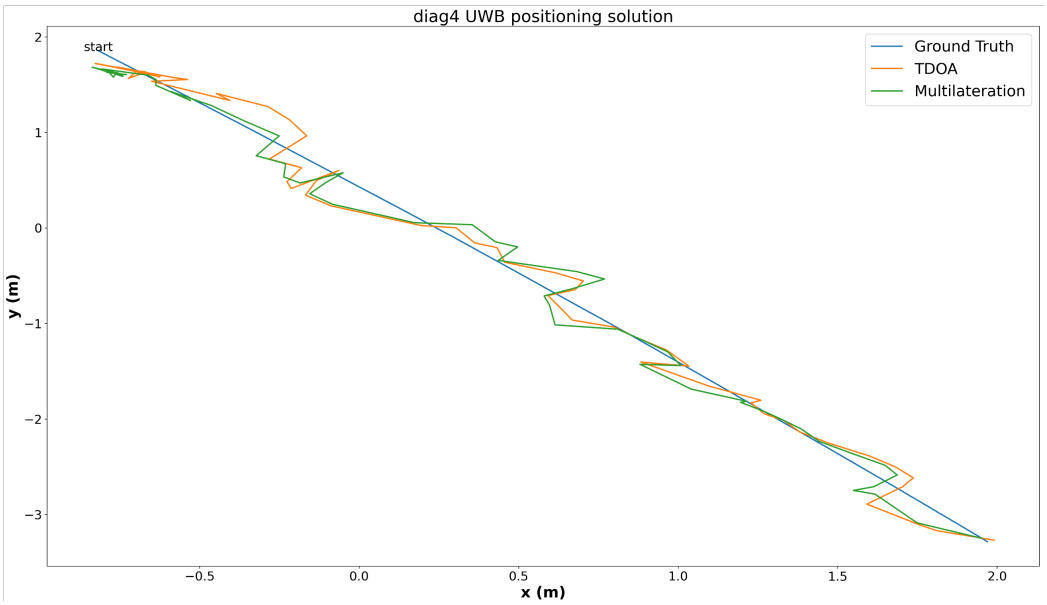


Figure 90: UWB positioning solution of straight line diag4.

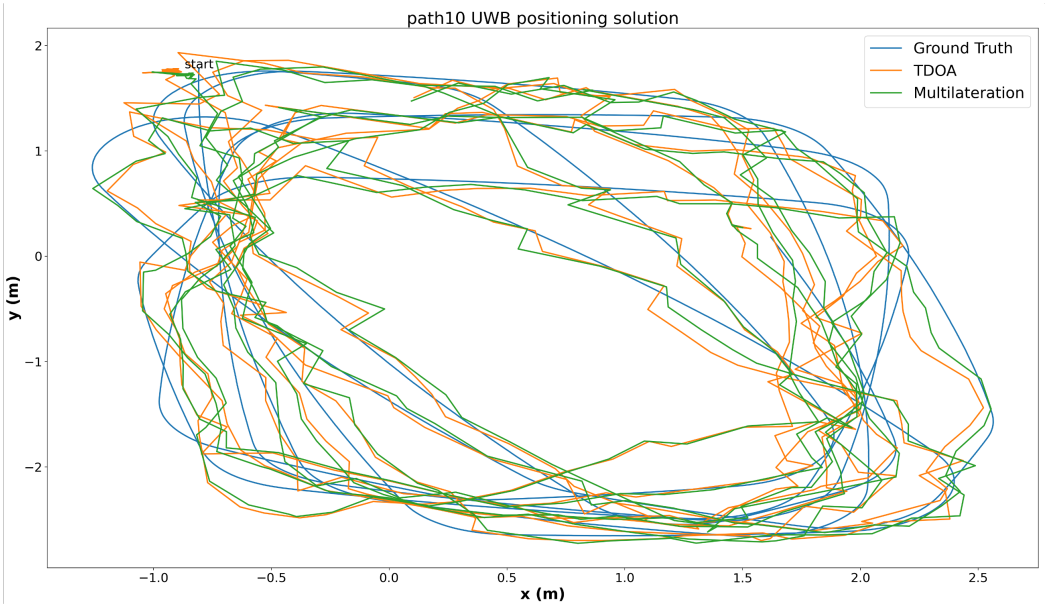


Figure 91: UWB positioning solution of random path path10.

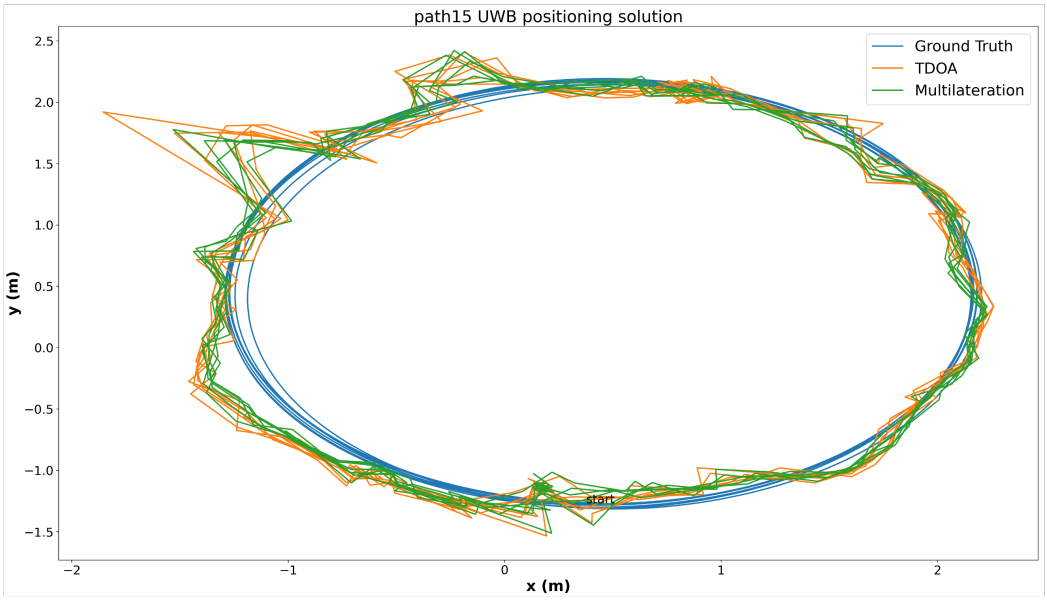


Figure 92: UWB positioning solution of circular path path15.

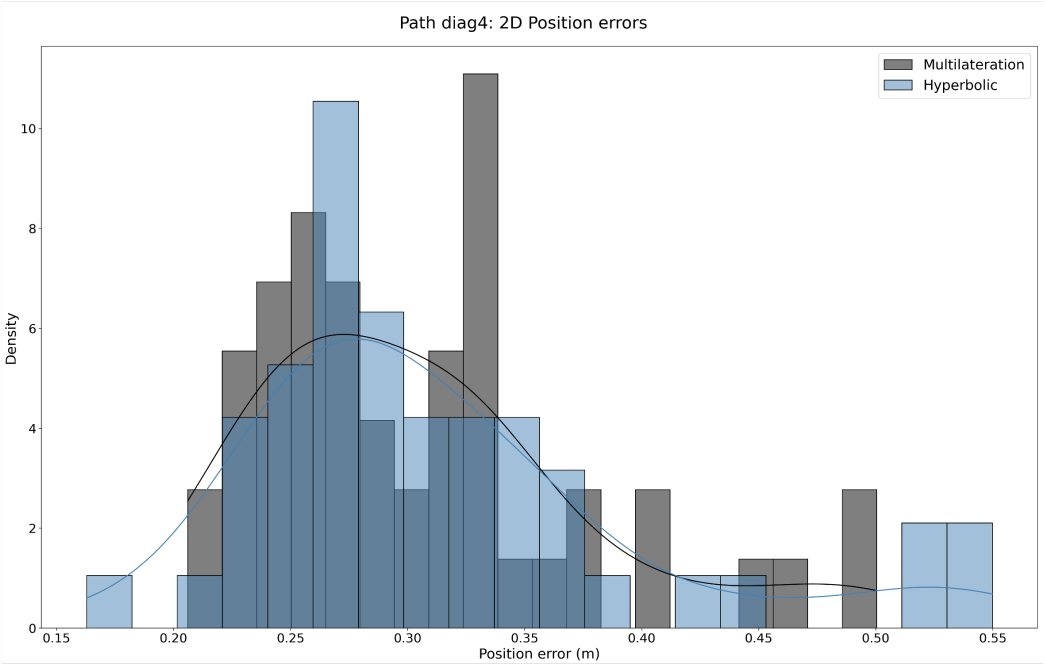


Figure 93: UWB 2D positioning solution errors for straight line diag4.

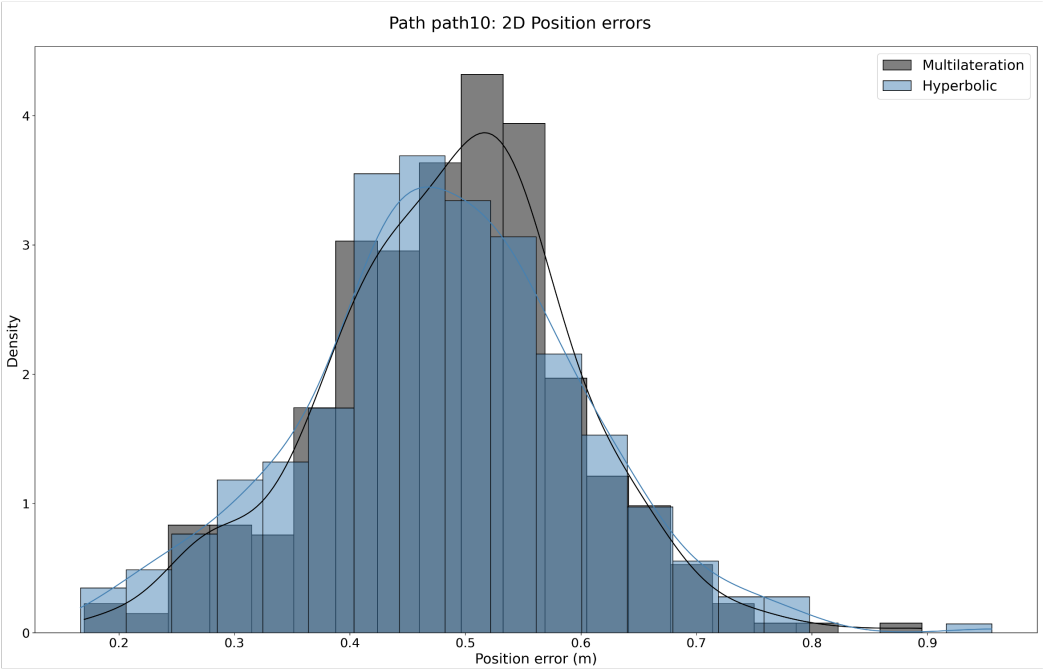


Figure 94: UWB 2D positioning solution errors of random path path10.

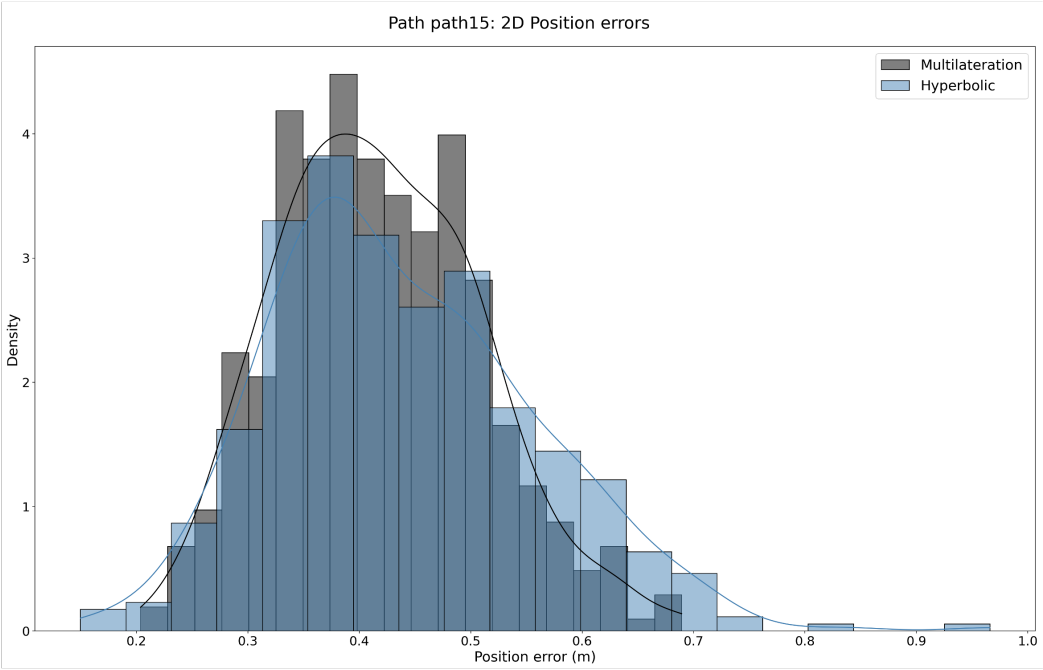


Figure 95: UWB 2D positioning solution errors of circular path path15.

In order to numerically compare the performance of the two methods the RMSE and the accuracy at a 95% confidence level are calculated for both methods and are shown in table 10, for all dynamic paths. For every path, the value highlighted in green corresponds to the method resulting in lower position errors (higher accuracy). As shown in table table 10 for the majority of the paths (75% of the cases), mulilateration results in better accuracy performance compared to the hyperbolic method. Multilateration performs on average 6.5% better than the hyperbolic method when considering accuracy (95%) and 3% better when considering RMSE. A potential reason for this is that the hyperbolic method might be more sensitive to errors in the measurements, compared to multilateration. As discussed in section 7.1.3, in the hyperbolic method, the position at every epoch is calculated as the average of eight position estimations. If an erroneous measurement is present within an epoch, it will have an effect on all eight position estimations, which will propagate to the positioning solution when the average is taken.

Table 10: Accuracy at 95% and RMSE of multilateration and hyperbolic methods for dynamic paths.

Path	Multilateration		Hyperbolic		Difference (m)	
	Accuracy at 95% (m)	RMSE (m)	Accuracy at 95% (m)	RMSE (m)	Accuracy at 95%	RMSE (m)
diag1	**0.373**	**0.273**	0.404	0.281	0.031	0.008
diag2	0.376	**0.263**	**0.374**	0.268	0.002	0.006
diag3	0.439	0.282	**0.383**	**0.260**	0.056	0.022
diag4	**0.463**	**0.314**	0.514	0.324	0.052	0.010
diag5	**0.401**	**0.264**	0.442	0.279	0.041	0.014
diag7	0.627	0.529	**0.504**	**0.394**	0.123	0.134
diag8	0.419	0.326	**0.375**	**0.310**	0.044	0.017
diag9	**0.368**	**0.247**	0.406	0.264	0.038	0.017
path1	**0.527**	**0.385**	0.550	0.390	0.024	0.004
path2	**0.613**	**0.432**	0.627	0.441	0.013	0.009
path3	0.396	0.273	**0.394**	**0.272**	0.002	0.001
path5	**0.448**	**0.308**	0.474	0.314	0.026	0.005
path6	**0.433**	**0.310**	0.477	0.319	0.044	0.009
path7	**0.396**	**0.269**	0.403	0.271	0.007	0.002
path8	**0.364**	**0.250**	0.392	0.252	0.028	0.002
path9	**0.527**	**0.387**	0.555	0.389	0.028	0.002
path10	**0.655**	0.498	0.662	**0.494**	0.007	0.004
path11	**0.305**	**0.201**	0.318	0.214	0.012	0.013
path12	**0.668**	**0.528**	0.718	0.528	0.050	0.000
path13	**0.847**	**0.694**	0.875	0.707	0.028	0.014
path14	**1.199**	1.065	1.226	**1.057**	0.026	0.008
path15	**0.575**	**0.427**	0.644	0.454	0.069	0.027

Static data are used to further investigate the impact of excluding measurement outliers prior to the calculation of the positioning solution. The outliers of the range measurements are identified by calculating the interquartile range (IQR) of range measurements of every anchor at a specific position. The IQR defines the upper and lower bounds of range measurements, outside which measurements are considered as outliers. Once the range measurement outliers are eliminated, the remaining measurements are used to calculate the positioning solution using both positioning methods.

The position error distributions of the two methods are illustrated in figures 96 to 103. A visual observation of the error distributions shows that for pos4 and pos7 (figures 99 and 102) the hyperbolic method performs better than multilateration. Alternatively, for pos1, pos2, pos5 and pos6 (figures 96, 97, 100 and 101), multilateration outperforms the hyperbolic method. For pos3 and pos8 (figures 98 and 103), the error distributions of the two methods overlap, however, multilateration result in a narrower distribution and hence its value of accuracy (95%) is expected to be better compared to the hyperbolic method.

To numerically evaluate and compare the performance of the two methods, table 11, presents the accuracy(95%) and the RMSE at every position. Table 11 and the error distributions in figures 96 to 103, illustrate that for the majority of static positions, multilateration performs on average 19% better to the hyperbolic method, when considering accuracy (95%).

Table 11: Accuracy at 95% and RMSE of multilateration and hyperbolic methods for static points.

Path	Multilateration		Hyperbolic		Difference (m)	
	Accuracy at 95% (m)	RMSE (m)	Accuracy at 95% (m)	RMSE (m)	Accuracy at 95%	RMSE (m)
pos1	**0.237**	**0.182**	0.318	0.267	0.083	0.090
pos2	0.097	**0.046**	**0.096**	0.140	0.001	0.094
pos3	**0.061**	**0.041**	0.075	0.048	0.014	0.007
pos4	0.178	0.157	**0.153**	**0.130**	0.025	0.027
pos5	**0.227**	**0.194**	0.255	0.220	0.027	0.025
pos6	**0.126**	**0.105**	0.148	0.566	0.023	0.462
pos7	0.209	0.179	**0.199**	**0.164**	0.010	0.015
pos8	**0.168**	**0.114**	0.175	0.138	0.014	0.034

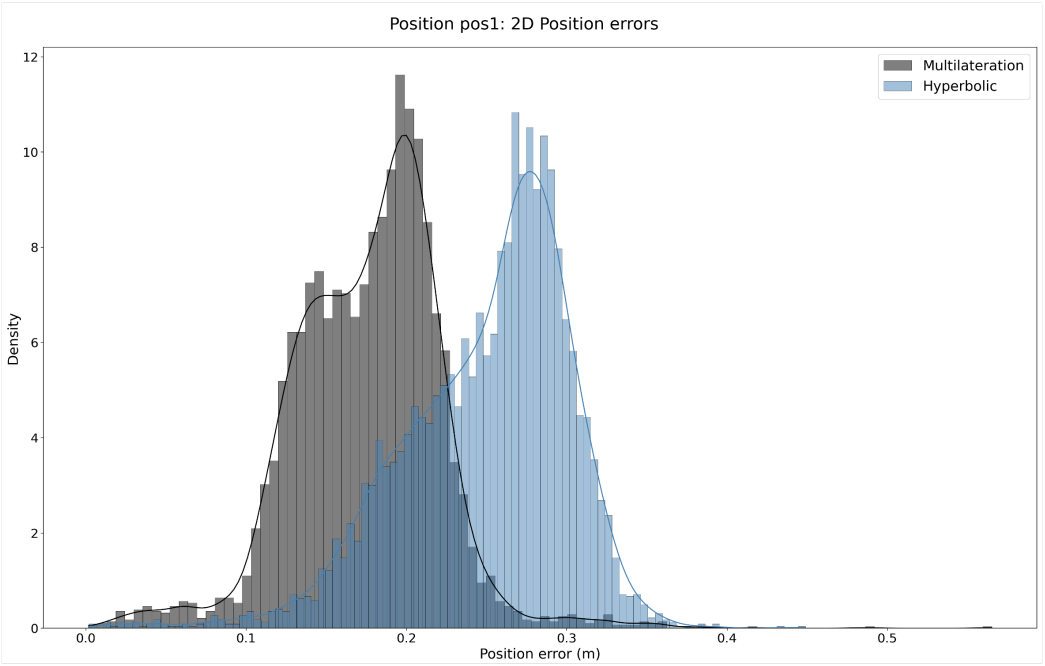


Figure 96: UWB 2D positioning solution errors for position pos1.

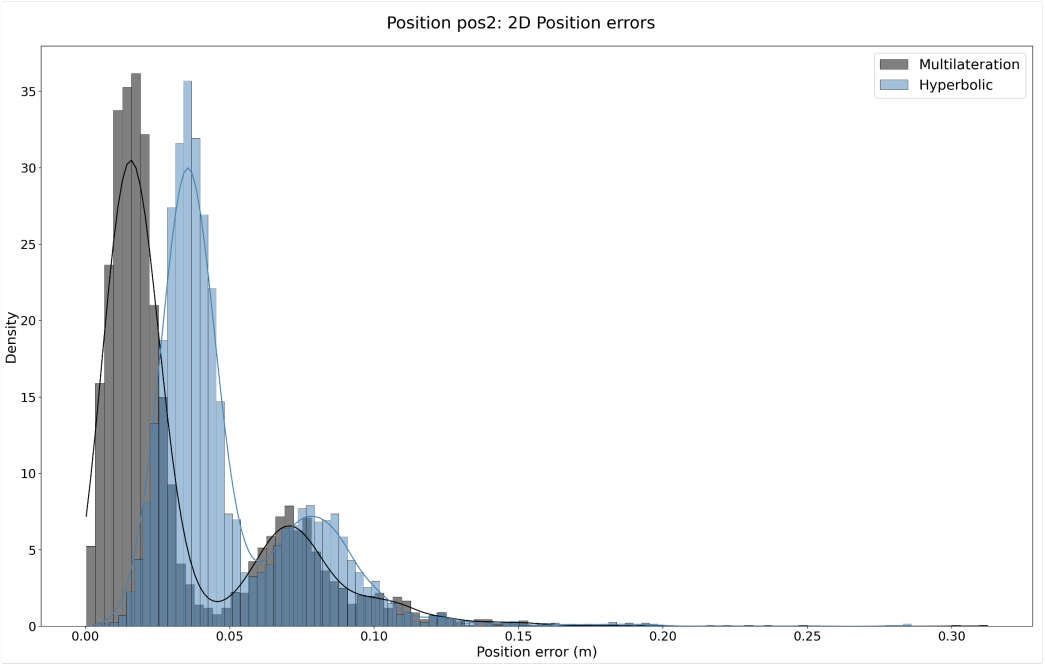


Figure 97: UWB 2D positioning solution errors for position pos2.

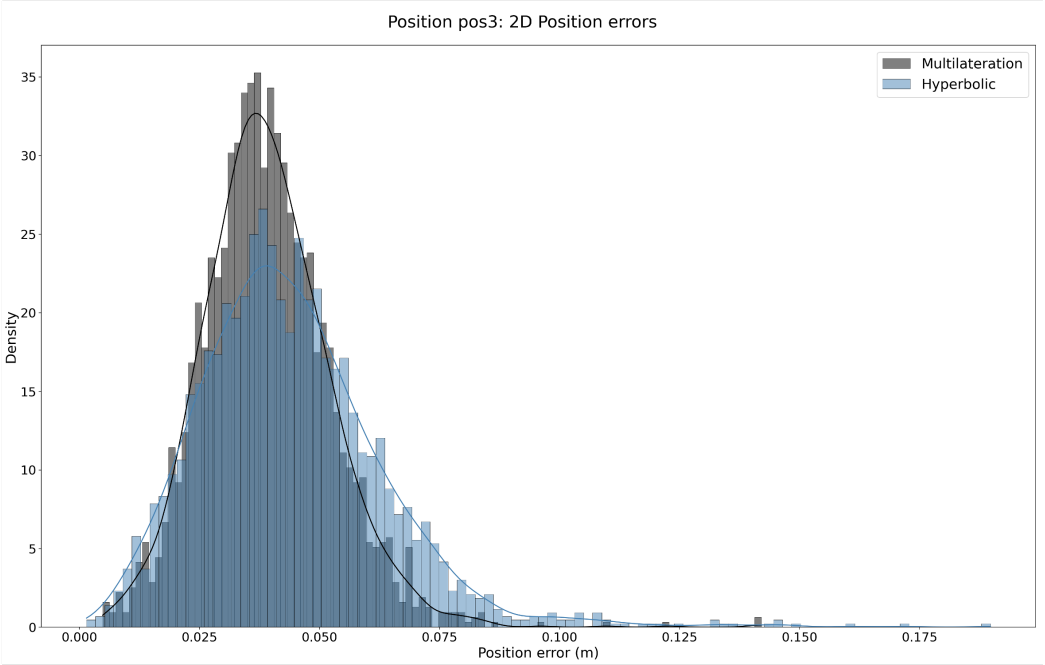


Figure 98: UWB 2D positioning solution errors for position pos3.

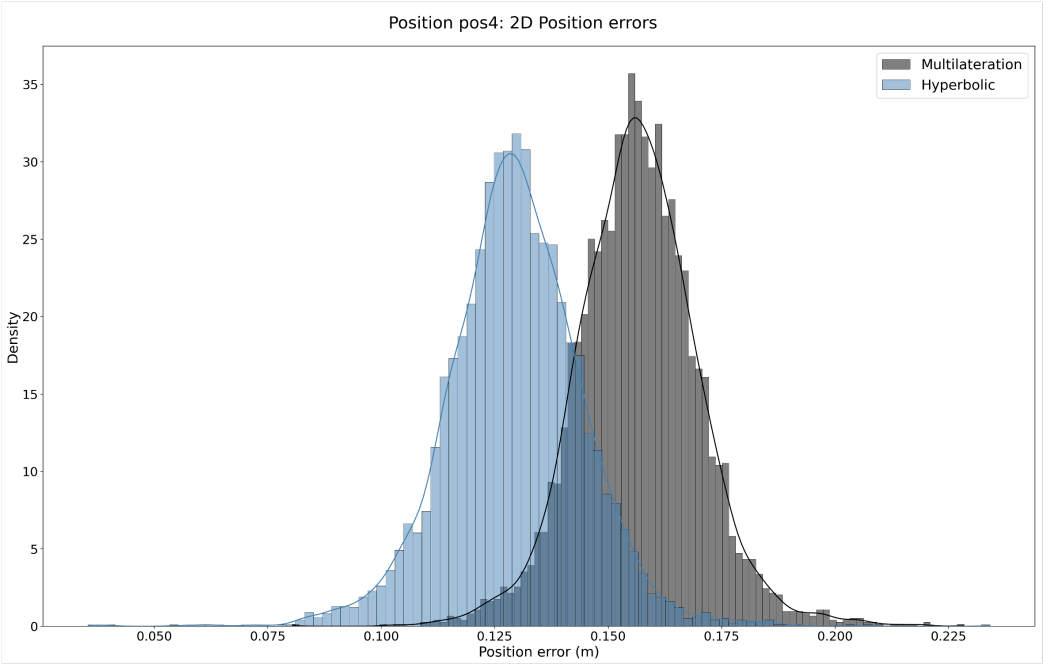


Figure 99: UWB 2D positioning solution errors for position pos4.

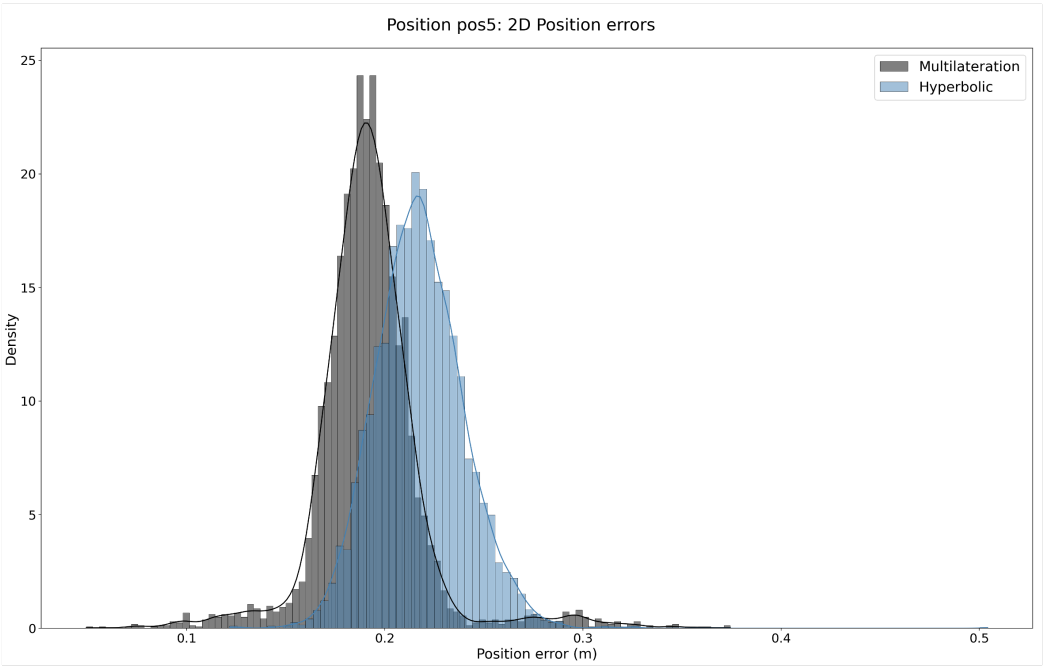


Figure 100: UWB 2D positioning solution errors for position pos5.

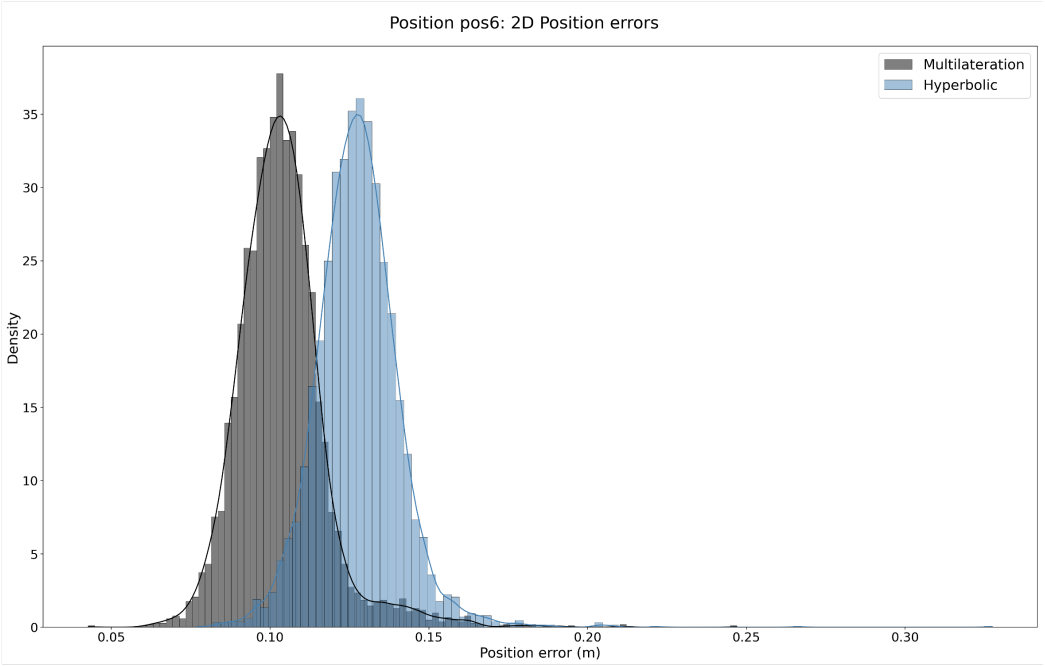


Figure 101: UWB 2D positioning solution errors for position pos6.

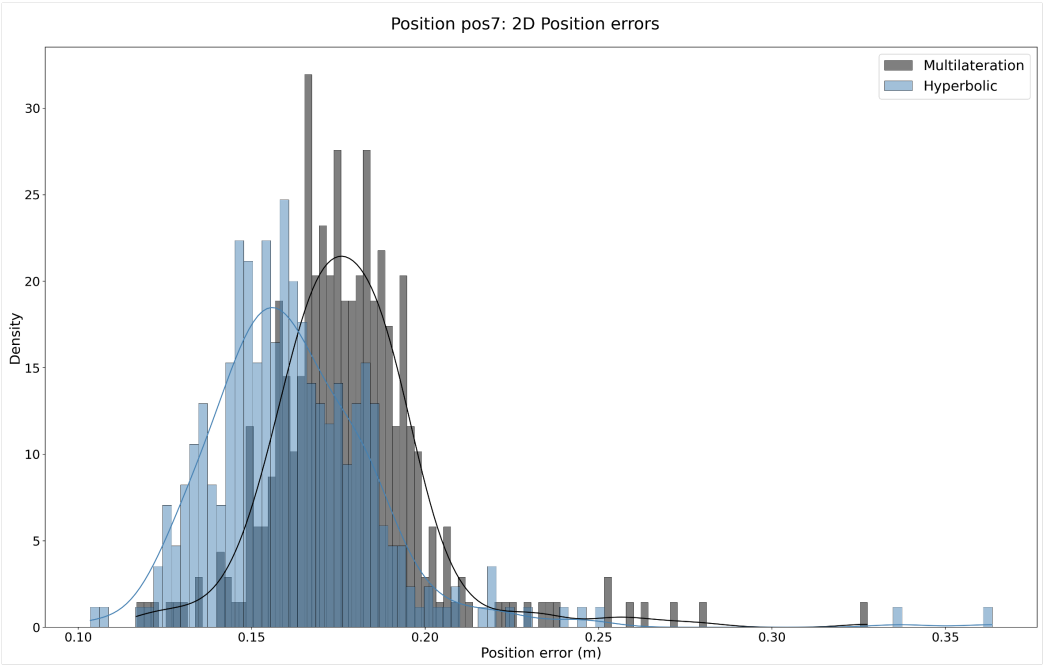


Figure 102: UWB 2D positioning solution errors for position pos7.

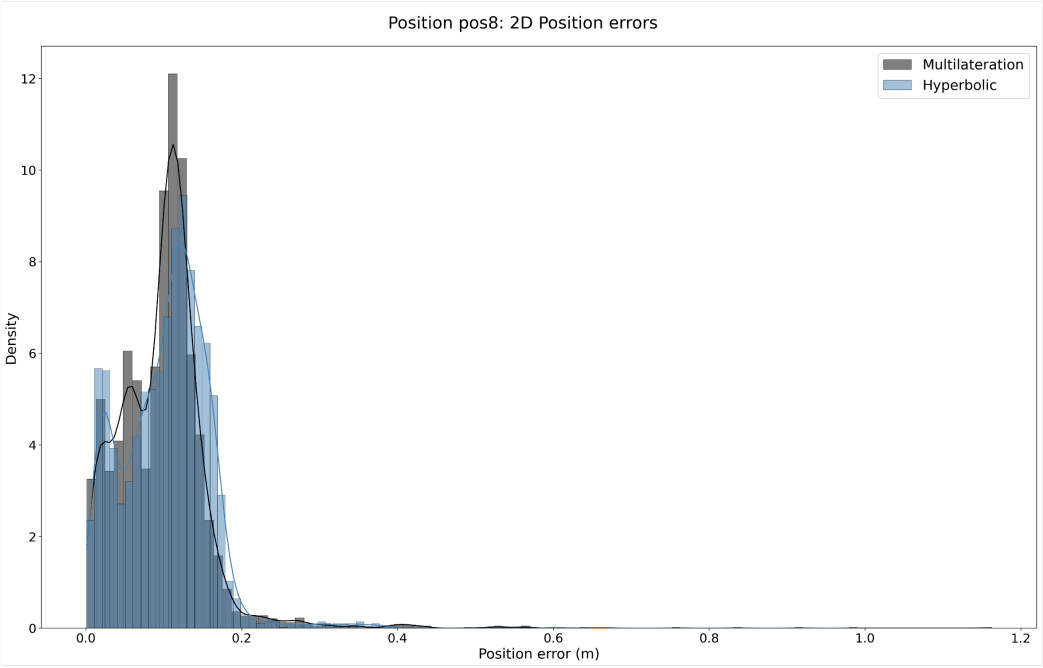


Figure 103: UWB 2D positioning solution errors for position pos8.

Although multilateration outperforms the hyperbolic method for the majority of paths, the difference in performance, shown in tables 10 and 11, is not significant compared to the target levels of accuracy. The difference in accuracy (95%) and RMSE between the two methods is at a centimeter level for all the paths and static points, while for the majority it is less than 5cm, which is not considered significant in our case.

Both methods provide long term positioning stability, which refers to a system ability to maintain consistent levels of accuracy of the positioning solution. However, the provision of positioning solution will be compromised in scenarios where signals are either unavailable or NLOS and multipath interference occur. Moreover, the best levels of accuracy (95%) achieved by the UWB system is about $9cm$ for the static cases and $31cm$ for the dynamic cases. Even though these satisfy the decimetric levels of accuracy requirements (defined in chapter 2), there are dynamic paths where the accuracy is compromised, like path14 with an accuracy of $1.22m$. The integration of UWB system with an alternative system able to maintain a positioning solution when UWB is unavailable can bridge the gaps in UWB solutions and improve the accuracy performance.

7.4. Summary

In this chapter the raw UWB range measurements were analysed and the UWB-based positions were calculated using two positioning methods. The analysis of the raw measurement data was done for static and dynamic cases in terms of the range measurement errors. Range errors of the static cases were fist investigated, where multi-peak error distributions were observed. The error distributions for eight static positions for every anchor were also investigated, with the majority of the cases exhibiting normal distributions and the others having multi-peak distributions. Multi-peak distributions did not occur for the same position across anchors. A plausible explanation for the multiple peaks is the effect of signal propagation errors, or error related to the geometry between the anchor and the position.

The analysis of dynamic measurement errors illustrated the dependencies of measurement errors on direction and speed of motion. The measurement errors when the platform is moving towards the anchor tended to be normally distributed with a negative mean. Alternatively, when the platform is moving away from the anchor, normally distributed errors with a positive mean were observed. The mean and standard deviation of both scenarios were calculated, however, measurement corrections were not applied prior to the UWB-standalone position calculation. The implementation of these corrections require knowledge of the direction of motion and hence they are used in chapter 9, in the integration of IMU and UWB, where the orientation calculated by the IMU is used to calculate the direction of motion.

The analysis of the dynamic case also revealed a dependency of measurement errors on speed of motion. The strong correlation between the rate of change of measurement error with distance from the anchor and speed of motion was observed mainly from the four corner anchors, alluding to a further dependency on geometry. The observed strong correlations, suggest that at higher speeds, the rate of change of measurement errors with distance from the anchor, becomes greater in magnitude, resulting in larger errors. Although UWB measurement corrections depending on the speed of motion are not applied

in this thesis, further investigation of this relationship will be beneficial.

The analysis of measurement errors was followed by the calculation of the UWB-based positioning solution, using the hyperbolic method and multilateration. The accuracy performance of the two methods was evaluated using RMSE and accuracy (95%) metrics. The method of multilateration performed on average 6.5% and 19% better than the hyperbolic method in terms of accuracy (95%), for the dynamic and static cases respectively. Although this is the case, the difference in performance of the two methods, for all the paths and static points is at a centimeter level, with the majority being less than 5cm. In our case, this is not considered a significant difference when compared to the decimetric level of accuracy requirement. In addition, for the majority of the cases, multilateration results in sub-meter levels of accuracy, with the exception of path14 where the accuracy is 1.2m.

Although UWB is a system that can provide long term stability, its performance can be compromised in scenarios where the signal is either blocked, NLOS or affected by multipath, causing short term instabilities. The integration of UWB with an alternative system can bridge the gaps in UWB solution and improve the accuracy performance. A system able to provide short term stability and of which the performance is not affected by the environmental conditions is an INS. The next chapter makes use of the data collected by the IMU, to calculate the INS-based solution and evaluate its performance.

Chapter 8

Inertial Measurement Unit measurements and position domain performance characterisation

An inertial navigation system (INS) uses the inertial data collected by the inertial measurement unit (IMU) to calculate the navigation solution of the sensor. As discussed in section 5.4.2, the data collected by the IMU were acceleration and angular velocities, tracking the motion of the dynamic platform. This chapter, describes the method used to calculate the INS-based solution using the Madgwick filter for orientation estimation of the sensor and relying on the principles of DR for position estimation. A prerequisite of the DR method, is the initialisation of the system. The IMU is initialised using the camera truthing system, where the initial position and orientation are determined.

This chapter analyses the raw accelerations and angular velocities, to identify errors in the measurements. The raw IMU data and in particular acceleration are very noisy, necessitating pre-processing before calculating the IMU orientation and positioning solution. Acceleration data are filtered to smooth the raw measurements and reduce noise. The pre-processed acceleration and angular velocity data are then used in a Madgwick filter to estimate the orientation of the dynamic platform at each epoch, and used once the system is initialised to calculate the INS-based positioning solution. Three approaches are used to calculate the INS-based positioning solution. Each uses different degrees of input from the accelerometer to evaluate the main sources of errors in the INS position.

The process followed in this chapter is illustrated in Figure 104. Section 8.1 introduces the Madgwick filter used to estimate the orientation of the IMU at every epoch, and describes the IMU initialisation process. Section 8.2, analyses the raw IMU measurements

and observes the noise in acceleration measurements. This leads to the pre-processing of the acceleration measurements in section 8.3, in order to reduce the noise in the measurements. Once the raw measurements are pre-processed, section 8.4 explains the three approaches used to calculate the INS-based position. A comparison of the three approaches and a discussion about the main sources of positioning errors is then presented in section 8.5.

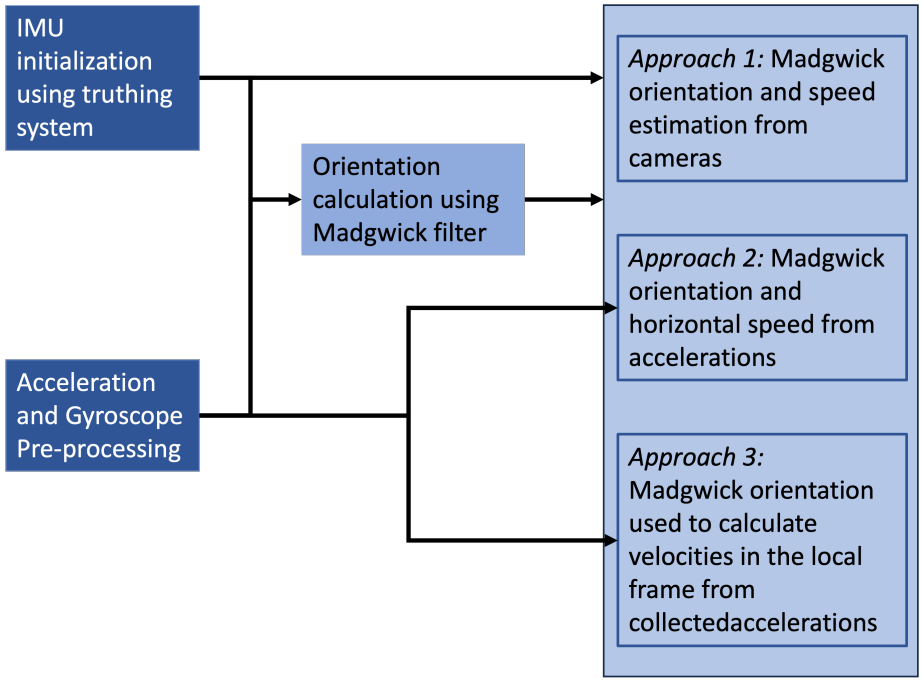


Figure 104: IMU position estimation method.

8.1. Methodology

8.1.1. Madgwick filter

The Madgwick filter is used to estimate the orientation of the dynamic platform, using data from an IMU. The IMU measurements used are accelerations and angular rates from the accelerometer and gyroscope respectively. The Madgwick filter uses quaternions as a representation of orientation instead of Euler angles, as quaternions can eliminate the issue of singularities (also known as gimbal lock) that are associated with Euler angles. The orientation of the INS is calculated by fusing the orientation estimations derived based on the angular rates and acceleration measurements.

The orientation calculated using angular rate measurements ($q_{\omega,t}$) is shown in (33). $q_{\omega,t}$ at time t is calculated by adding the change in orientation ($\dot{q_{\omega,t}}\Delta t$) to the estimated orientation of the previous time-step $q_{est,t-1}$. The rate of change of orientation $\dot{q_{\omega,t}}$ is calculated as shown in equation (32), where ω_t is the quaternion representation of the raw angular velocity measurements. The raw angular velocities in the x, y and z direction of the IMU are ω_x, ω_y and ω_z respectively.

$$\omega_t = \begin{bmatrix} 0 & \omega_x & \omega_y & \omega_z \end{bmatrix} \tag{31}$$

$$\dot{q_{\omega,t}} = 0.5 q_{est,t-1} \otimes \omega_t \tag{32}$$

$$q_{\omega,t} = q_{est,t-1} + \dot{q_{\omega,t}}\Delta t \tag{33}$$

Alternatively, the orientation calculation using acceleration measurements is based on the knowledge of the direction of Earth's acceleration field. If the direction of an Earth's field is known, and measurements of the specific field in the sensor frame exist, the orientation of the sensor can be calculated. Within the Madgwick filter, this is achieved through the formulation of an optimisation problem. The optimisation problem tries to solve for the orientation of the sensor q, that minimises the difference between the predefined field in the Earth's frame ${}^{E}d$ and the measurement of the field in the sensor frame ${}^{S}s$, defined by $f(q, {}^{E}d, {}^{S}s)$. The formulation of the optimisation problem is as follows:

$$\min f(q, {}^{E}d, {}^{S}s) \tag{34}$$

$$f(q, {}^{E}d, {}^{S}s) = q^* \otimes {}^{E}d \otimes q - {}^{S}s \tag{35}$$

$$q = \begin{bmatrix} q_w & q_x & q_y & q_z \end{bmatrix} \tag{36}$$

$$^{E}d = \begin{bmatrix} 0 & d_x & d_y & d_z \end{bmatrix} \tag{37}$$

$$^{S}s = \begin{bmatrix} 0 & s_x & s_y & s_z \end{bmatrix} \tag{38}$$

This optimisation problem can be solved using the gradient descent algorithm as described in equations (39) to (42). μ is the step-zise used in the iterative process, and $\nabla f(q_{est,t-1}, {}^{E}d, {}^{S}s)$ is the gradient of the solution which is computed using the objective function and its Jacobian as shown in (40).

$$q_{\nabla,t} = q_{est,t-1} - \mu \frac{\nabla f(q_{est,t-1}, {}^{E}d, {}^{S}s)}{\parallel \nabla f(q_{est,t-1}, {}^{E}d, {}^{S}s) \parallel} \Delta t \tag{39}$$

where:

$$\nabla f(q_{est,t-1}, {}^{E}d, {}^{S}s) = J^{T}(q_{est,t-1}, {}^{E}d) f(q_{est,t-1}, {}^{E}d, {}^{S}s) \tag{40}$$

$$f(q_{est,t-1}, {}^{E}d, {}^{S}s) = \begin{bmatrix} 2d_x(0.5 - q_y^2 - q_z^2) + 2d_y(q_w q_z + q_x q_y) + 2d_z(q_x q_z - q_w q_y) - s_x \\ 2d_x(q_x q_y - q_w q_z) + 2d_y(0.5 - q_x^2 - q_z^2) + 2d_z(q_w q_x + q_y q_z) - s_y \\ 2d_x(q_w q_y + q_x q_z) + 2d_y(q_y q_z - q_w q_x) + 2d_z(0.5 - q_x^2 - q_y^2) - s_z \end{bmatrix} \tag{41}$$

$$J^{T}(q_{est,t-1}, {}^{E}d) = \begin{bmatrix} 2d_y q_z - 2d_z q_y & 2d_y q_y + 2d_z q_z & -4d_x q_y + 2d_y q_x - 2d_z q_w & -4d_x q_z + 2d_y q_w + 2d_z q_x \\ -2d_x q_z + 2d_z q_x & 2d_x q_y - 4d_y q_x + 2d_z q_w & 2d_x q_x + 2d_z q_z & -2d_x q_w - 4d_y q_z + 2d_z q_y \\ 2d_x q_y - 2d_y q_x & 2d_x q_z - 2d_y q_w - 4d_z q_x & 2d_x q_w + 2d_y q_z - 4d_z q_y & 2d_x q_x + 2d_y q_y \end{bmatrix} \tag{42}$$

The process described by equations (34) to (42), illustrates the method used for determining the orientation given by any measurements of a known Earth field. Data from the

accelerometer represent any accelerations experienced by the sensor, including acceleration due to gravity. The usual convention is to assume that the direction of gravity is equivalent to the vertical, z axis of the sensor. The predefined gravitational field ${}^{E}g$ and acceleration measurements normalised with gravity in the sensor frame ${}^{S}\alpha$ are hence defined as:

$$ {}^{E}g = \begin{bmatrix} 0 & 0 & 0 & 1 \end{bmatrix} \tag{43} $$

$$ {}^{S}\alpha = \begin{bmatrix} 0 & \alpha_x & \alpha_y & \alpha_z \end{bmatrix} \tag{44} $$

Replacing ${}^{E}d$ and ${}^{S}s$ with ${}^{E}g$ and ${}^{S}\alpha$ respectively in equations(41) and (42), results in equations (45) and (46).

$$ f_g(q, {}^{S}a) = \begin{bmatrix} 2(q_x q_z - q_w q_y) - a_x \\ 2(q_w q_x + q_y q_z) - a_y \\ 2(0.5 - {q_x}^2 - {q_y}^2) - a_z \end{bmatrix} \tag{45} $$

$$ J_g(q) = \begin{bmatrix} -2q_y & 2q_z & -2q_w & 2q_x \\ 2q_x & 2q_w & 2q_z & 2q_y \\ 0 & -4q_x & -4q_y & 0 \end{bmatrix} \tag{46} $$

The orientation of the IMU (q_t) is estimated by fusing the orientations calculated by the angular velocities ($q_{\omega,t}$), given by (33), and accelerations ($q_{\nabla,t}$), given by (39). The fusion is a weighted sum of the two orientations with weights γ_t and $(1 - \gamma_t)$ applied to $q_{\nabla,t}$ (39) and $q_{\omega,t}$ (33) respectively as shown by:

$$ q_t = \gamma_t q_{\nabla,t} + (1 - \gamma_t)\, q_{\omega,t}, 0 \leq \gamma_t \leq 1 \tag{47} $$

Weighting parameter γ_t, is determined such that the weighted divergence of $q_{\omega,t}$ is equal to the weighted convergence of $q_{\nabla,t}$. The divergence rate of $q_{\omega,t}$ is defined as β and the rate of convergence of $q_{\nabla,t}$ as $\frac{\mu_t}{\Delta t}$. The fusion of the two orientations is described in detail by

Madgwick et al. (2010) with γ_t approximated as shown in (48) and the fusion of the two orientations q_t shown by (49):

$$\gamma_t \approx \frac{\beta \Delta t}{\mu_t} \tag{48}$$

$$q_t = q_{t-1} + \left(\dot{q_{\omega,t}} - \beta \frac{J_g^T(q_{t-1}) f_g(q_{t-1}, {}^S a_t)}{\parallel J_g^T(q_{t-1}) f_g(q_{t-1}, {}^S a_t) \parallel} \right) \Delta t \tag{49}$$

The measured accelerations and angular velocities are used in the Madgwick filter to estimate the orientation of the IMU. The orientation estimate in integration with an estimation of the distance traveled, result in the position estimation of the IMU, using the principles of DR. A position estimation is calculated only when the INS is initialised, as explained in the following section.

8.1.2. INS initialisation

An INS is a relative positioning system, and its position estimates are related to the last known/estimated position by changes in orientation and displacements. A position estimation using the IMU, can only be achieved when the initial position and orientation is known, making initialisation an essential step in INS positioning. The aim of the initialisation is the calculation of an initial position and orientation, both of which are then used in integration with changes in orientation and displacements to calculate IMU position. In this thesis INS initialisation is performed using the initial position and orientation derived from the camera-based truthing system. As discussed in section 6.2, camera measurements include time and position information. The initial position for every path is hence determined using the raw camera measurements.

Alternatively, initial orientation is calculated using camera positions, during the first 10 epoches of the moving trajectory. As describe in section 6.1.3, the initial tangent of the trajectory corresponds to the direction towards which the robot is moving. This is equivalent to the initial orientation of the robot and hence of the IMU sensor. The initial tangent of the trajectory is used to calculate the heading and hence the orientation of the IMU with respect to the local coordinate frame (ϑ), shown in Figure 49.

8.2. Raw acceleration and angular velocity data

Prior to the use of the IMU measurements, the raw acceleration and angular velocity measurements are investigated. The raw measurements of acceleration and angular velocity of a straight line path, diag1, are illustrated in figures 105 and 106 respectively, as an example of the collected data. The black vertical lines on the figures represent the interval of motion of the dynamic platform. The specific intervals is used to split the data of the path into static and dynamic. When visually comparing the raw measurements of accelerations and angular velocities, both appear noisy, while the accelerations seem to be slightly biased, as the accelerations during the static part of the trajectory do not oscillate about zero. The measurements collected during the static part of the trajectory, are used to investigate the raw acceleration and angular velocity measurements.

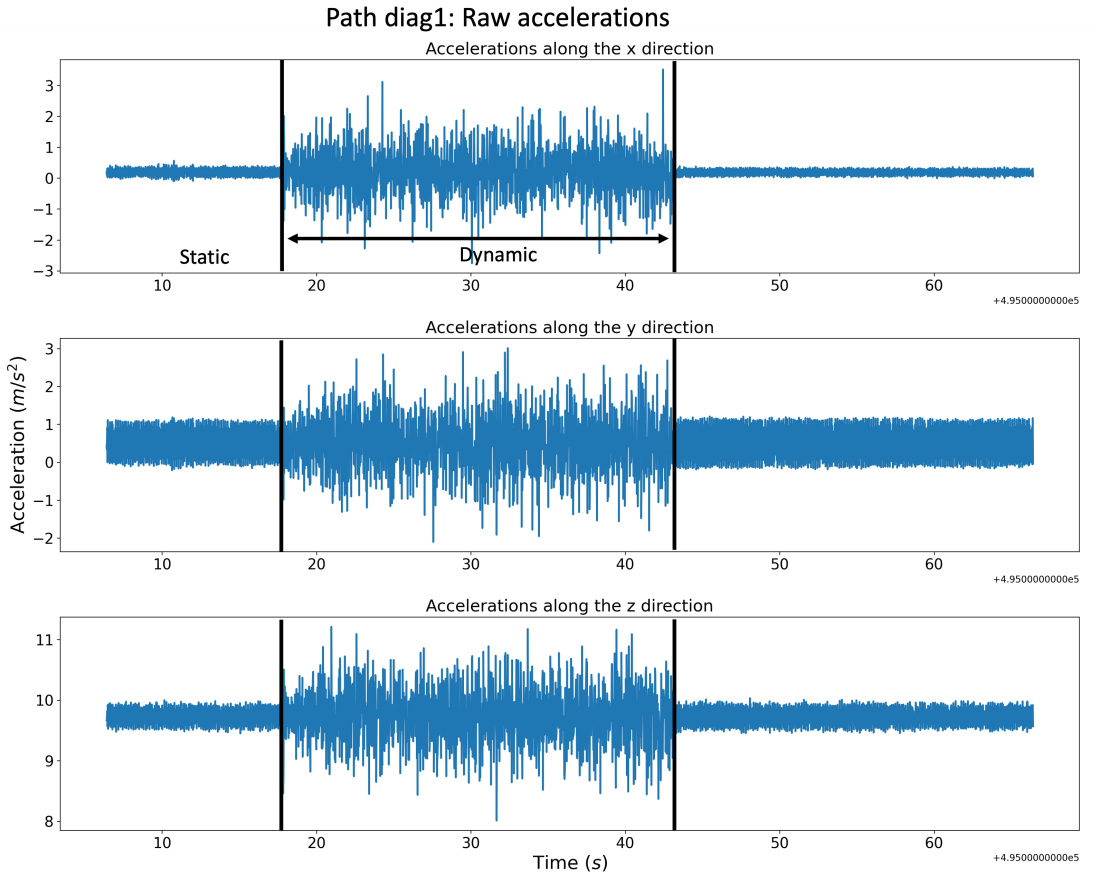


Figure 105: Raw acceleration measurements of path diag1.

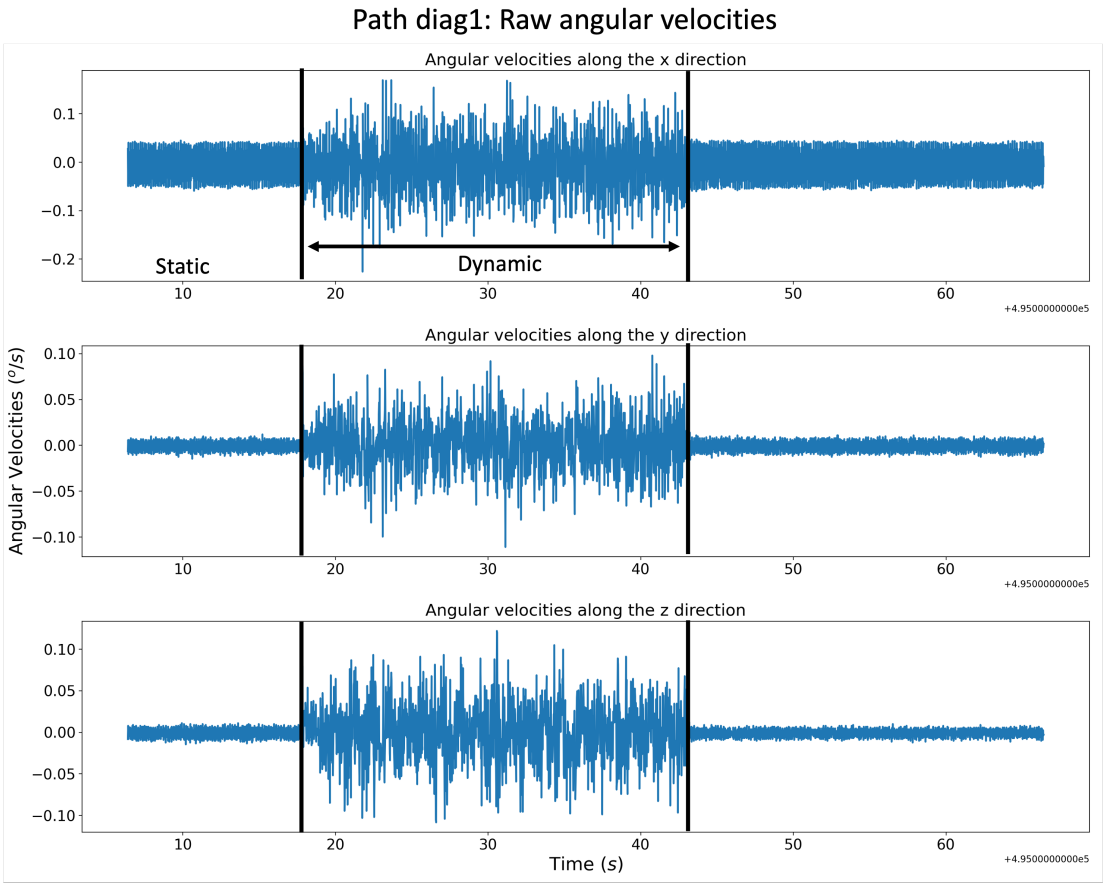


Figure 106: Raw angular velocity measurements of path diag1.

An initial step is the investigation of measurement biases using the static part of the trajectory. The bias in the measurements is determined by calculating the average of the raw measurements for the static part of the trajectory. While the dynamic platform is static, all three angular velocities (along the x, y, and z axis) and the accelerations along the x and y axis should be zero. Alternatively, the accelerations along the z axis should correspond to the value of gravity. The biases are calculated by subtracting the value of zero from the average angular velocities and accelerations in the x and y direction, and the value of gravity from the average accelerations in the z direction. Table 12, shows the calculated biases of every path and illustrate that the biases of angular velocities are insignificant.

On the other hand the accelerations have some bias. However, the biases are not

consistent across the different paths. A potential reason for this is that not all the paths start from the same position and hence the static data are collected from different locations around the room. This might indicate a slightly uneven surface that causes the bias to vary across the room. An uneven ground profile, causes the gravitational force not to be perfectly aligned with the z axis of the IMU, resulting in the projection of the gravitational force in the x and y axis of the sensor. The projection of the gravitational force in the three axis instead of just the z axis, results in non-zero measurements of acceleration in the x and y axis while the IMU is static. The projection of the gravitational force in the x and y components might be misinterpreted as bias in the measurements.

In addition to bias, the standard deviation of the measurements is used to evaluate the noise in the data. The standard deviations (σ) of measurements collected during the static part of the trajectories are calculated and are shown in table 13. As can be seen from table 13, σ values of the angular velocities are small, while acceleration measurements show a larger variation. The σ of acceleration measurements, indicate the level of noise in the measurements. The next section describes the way acceleration measurements are pre-processed in order to reduce noise.

Table 12: The average measurements during the static phase of the trajectories. The bias of the measurements is the same as the average except for the acceleration along the z-axis which is compared to gravity.

Path	Static Accelerations (m/s^2)				Static Angular velocities ($^\circ/s$)		
	$\mu_x = bias_x$	$\mu_y = bias_y$	μ_z	$bias_z$	$\mu_x = bias_x$	$\mu_y = bias_y$	$\mu_z = bias_z$
diag1	0.197	0.518	9.729	-0.077	$-5.567\text{E}-03$	$-6.197\text{E}-04$	$-8.559\text{E}-04$
diag2	0.200	0.518	9.734	-0.073	$-5.188\text{E}-03$	$-9.203\text{E}-04$	$-9.966\text{E}-04$
diag3	0.200	0.520	9.735	-0.072	$-5.265\text{E}-03$	$-8.377\text{E}-04$	$-1.092\text{E}-03$
diag4	0.194	0.510	9.734	-0.072	$-5.264\text{E}-03$	$-7.263\text{E}-04$	$-1.026\text{E}-03$
diag5	0.192	0.518	9.737	-0.070	$-5.041\text{E}-03$	$-8.216\text{E}-04$	$-1.144\text{E}-03$
diag7	0.211	0.515	9.737	-0.070	$-4.945\text{E}-03$	$-8.040\text{E}-04$	$-8.753\text{E}-04$
diag8	0.190	0.512	9.737	-0.069	$-4.938\text{E}-03$	$-7.379\text{E}-04$	$-7.506\text{E}-04$
diag9	0.220	0.521	9.742	-0.065	$-3.751\text{E}-04$	$1.089\text{E}-04$	$2.327\text{E}-04$
path1	0.194	0.499	9.741	-0.066	$-1.335\text{E}-04$	$-1.221\text{E}-04$	$5.719\text{E}-05$
path2	0.174	0.494	9.743	-0.064	$-1.088\text{E}-06$	$-1.778\text{E}-05$	$-1.285\text{E}-05$
path3	0.031	0.232	9.847	0.040	$-4.049\text{E}-05$	$2.112\text{E}-04$	$-1.033\text{E}-04$
path5	0.003	0.237	9.809	0.002	$-2.335\text{E}-04$	$2.192\text{E}-04$	$-9.882\text{E}-05$
path6	0.003	0.238	9.797	-0.010	$1.321\text{E}-05$	$7.607\text{E}-05$	$-6.769\text{E}-05$
path7	0.005	0.233	9.789	-0.017	$-1.334\text{E}-04$	$1.671\text{E}-04$	$2.414\text{E}-04$
path8	0.052	0.226	9.808	0.002	$7.939\text{E}-04$	$-2.885\text{E}-04$	$7.653\text{E}-04$
path9	0.047	0.221	9.789	-0.018	$-1.336\text{E}-04$	$4.157\text{E}-05$	$9.058\text{E}-04$
path10	0.066	0.223	9.779	-0.028	$8.182\text{E}-05$	$2.254\text{E}-04$	$4.046\text{E}-03$
path11	0.062	0.208	9.767	-0.040	$-4.794\text{E}-04$	$3.376\text{E}-04$	$1.703\text{E}-03$
path12	0.065	0.224	9.767	-0.039	$-5.584\text{E}-04$	$-7.677\text{E}-04$	$-1.828\text{E}-02$
path13	0.035	0.237	9.781	-0.025	$3.994\text{E}-05$	$2.351\text{E}-05$	$-3.298\text{E}-05$
path14	0.036	0.202	9.776	-0.031	$-1.695\text{E}-04$	$-1.131\text{E}-04$	$-5.303\text{E}-03$
path15	0.032	0.186	9.775	-0.032	$8.526\text{E}-05$	$-2.020\text{E}-04$	$-2.705\text{E}-03$

Table 13: The standard deviation σ of measurements during the static phase of the trajectories, representing the noise in the measurements.

Path	Static Accelerations (m/s^2)			Static Angular velocities ($^\circ/s$)		
	σ_x	σ_y	σ_z	σ_x	σ_y	σ_z
diag1	0.136	0.421	0.157	0.033	0.007	0.007
diag2	0.108	0.435	0.150	0.033	0.006	0.006
diag3	0.126	0.435	0.151	0.034	0.006	0.006
diag4	0.120	0.448	0.148	0.034	0.006	0.006
diag5	0.134	0.432	0.154	0.035	0.006	0.007
diag7	0.110	0.422	0.144	0.032	0.006	0.005
diag8	0.145	0.420	0.151	0.034	0.006	0.009
diag9	0.136	0.433	0.151	0.032	0.006	0.008
path1	0.192	0.414	0.147	0.034	0.006	0.010
path2	0.170	0.404	0.142	0.032	0.005	0.008
path3	0.220	0.392	0.098	0.023	0.003	0.011
path5	0.346	0.394	0.106	0.028	0.004	0.019
path6	0.138	0.372	0.091	0.020	0.003	0.008
path7	0.181	0.401	0.094	0.022	0.004	0.011
path8	0.290	0.493	0.247	0.027	0.010	0.015
path9	0.475	0.404	0.174	0.022	0.008	0.022
path10	0.515	0.545	0.371	0.032	0.018	0.024
path11	0.416	0.493	0.335	0.027	0.017	0.015
path12	0.586	0.528	0.330	0.026	0.017	0.070
path13	0.136	0.419	0.090	0.023	0.002	0.008
path14	0.258	0.432	0.195	0.022	0.009	0.033
path15	0.299	0.423	0.123	0.028	0.006	0.023

8.3. Acceleration pre-processing

This section illustrates the way in which the accelerations are pre-processed to reduce noise in the data before being used for position estimation. The noise in acceleration data, as seen in the previous section, leads to large standard deviations. A large value of standard deviation and hence noise in the data corresponds to signals of high frequencies. The Fourier transform of the acceleration signal is investigated to identify frequencies that correspond to any potential sources of noise in the measurements. A Fourier transform is performed separately for the static and dynamic part of the trajectory, to see whether there are any similarities between the two. An example of the Fourier transform of a straight line path ('diag1') is shown in figures 107 and 108.

In figures 107 and 108, the Fourier transform of the static and dynamic parts illustrate a peak in amplitude around $22Hz$. The fact that this peak in amplitude is present in both static and dynamic data indicates that the specific oscillation is not related to the motion of the robot. When considering the static phase of the trajectory, the only source of oscillation is the fan of the Jetson nano, the small computer used for the operation of the robot. The fan is located right underneath the IMU and is suspected as being the source of oscillation happening at about $20Hz$. When observing the Fourier transform of the acceleration data, especially for the static part of all the paths, it is seen that the frequency providing the peak in amplitude is not always the same. Another example of the Fourier transform is presented in figure 109 for the static part of 'path3'. As shown in figure 109, the peak in amplitude occurs for a frequency of about $19Hz$.

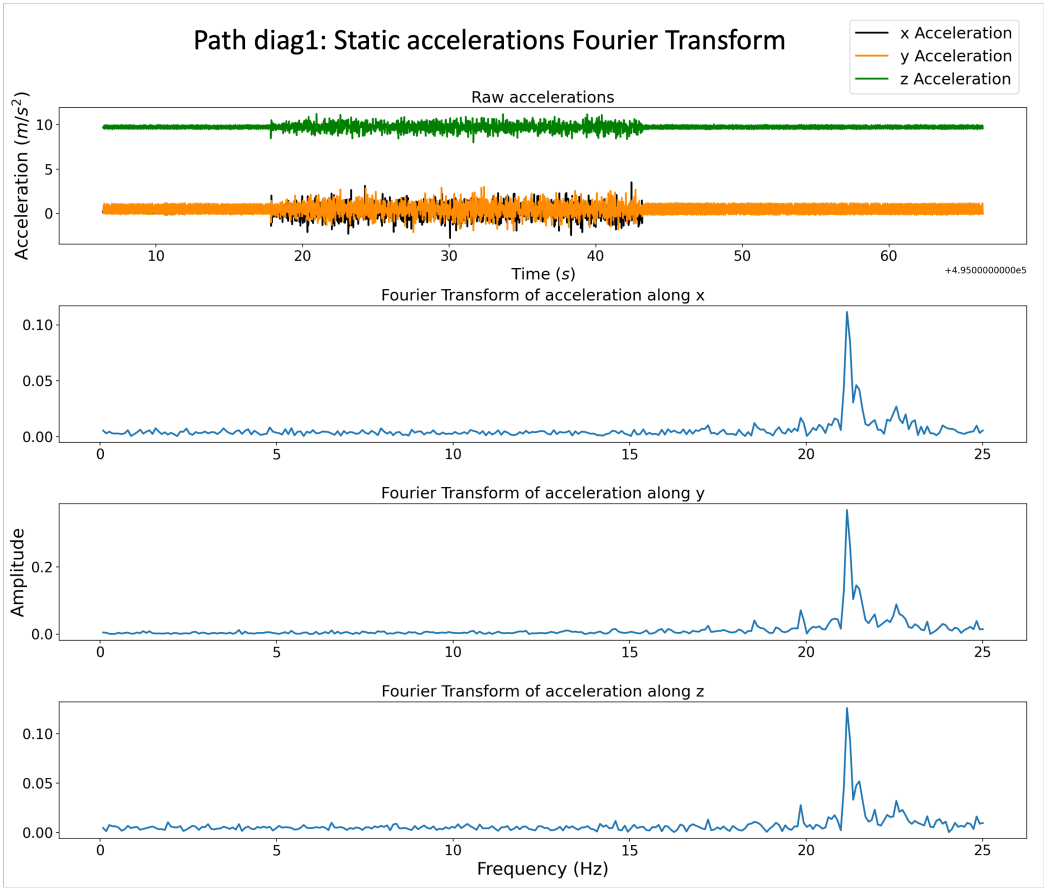


Figure 107: Fourier transform of accelerations collected during the static part of the trajectory.

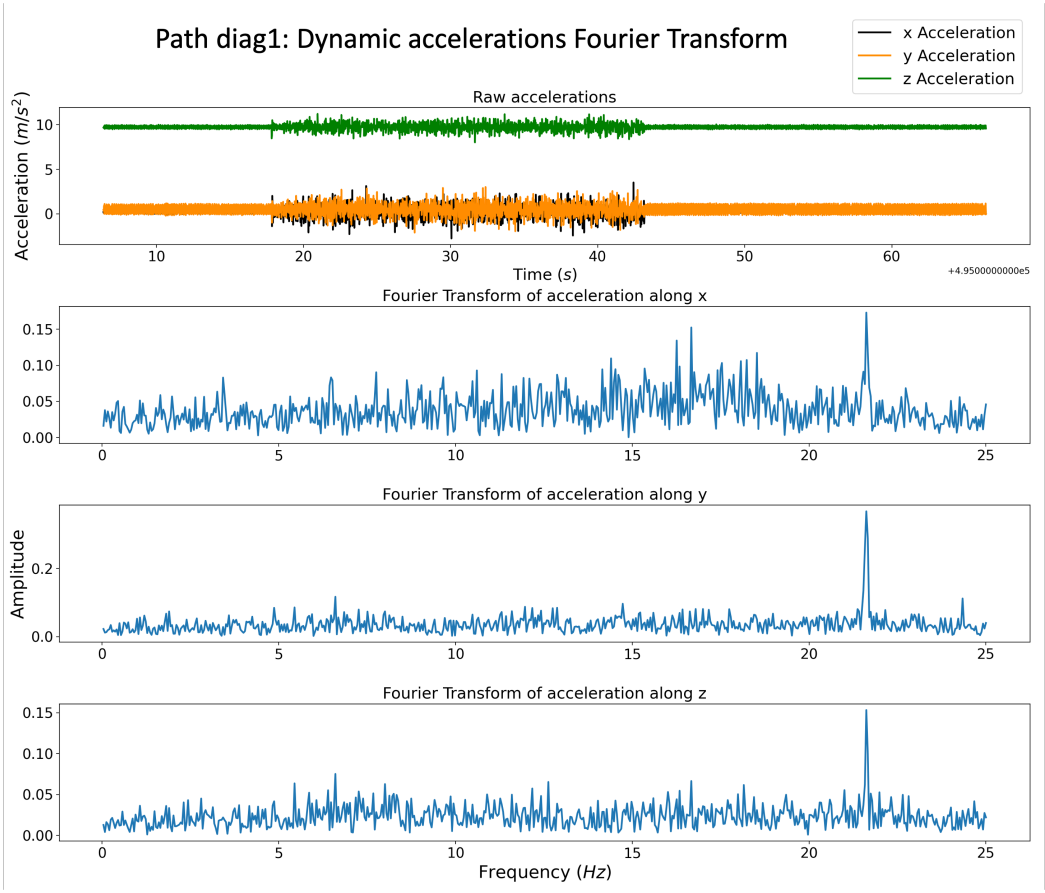


Figure 108: Fourier transform of accelerations collected during the dynamic part of the trajectory.

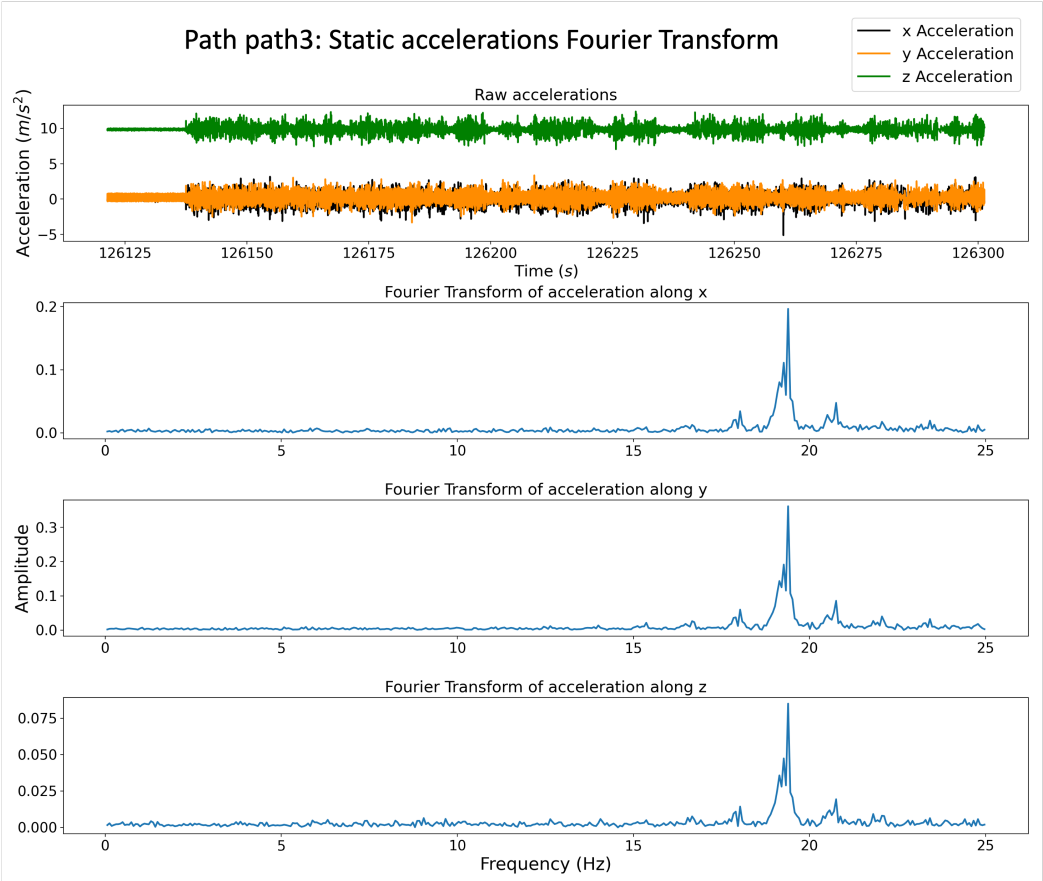


Figure 109: Fourier transform of accelerations collected during the static part of the trajectory.

A low-pass filter is used to filter the acceleration measurements and exclude any caused by the oscillations of the fan. A low-pass filter is used to filter out signals with frequencies above a pre-defined cut-off frequency. In this case, acceleration measurements are treated as a signal of measurements and the low-pass filter is used to eliminate signals of acceleration measurements resulting in frequencies above the pre-defined cut-off. The low-pass filter used is the Butterworth filter. This filter is easy to implement, while it also results in a maximally flat frequency response in the passband, reducing the distortion to the signal (AlHinai 2020). The filter is used with a cut-off frequency of $15Hz$ and filters out all the accelerations with a signal frequency larger than $15Hz$. Figure 110, shows the raw accelerations in the three orthogonal directions in blue, and the low-pass filtered accelerations in

red. It can be seen that although the low-pass filter during the static part of the trajectory works well, it does not for the dynamic part of the trajectory.

This is also evident from tables 14 and 15, where the former shows the standard deviations of the raw and filtered measurements from the static part of the trajectory and table 15 the standard deviations for the dynamic part of the trajectory. Although the static data show a significant improvement in terms of standard deviation and hence noise, when the low-pass filter is applied, the dynamic data are not significantly improved. This is a reasonable response according to the difference between the Fourier transforms of static and dynamic data, in figures 107 and 108 respectively. Fourier transform of the static measurements shows only one peak in amplitude at about $22Hz$, while the dynamic measurements have multiple peaks in amplitude at different frequencies. Consequently filtering out frequencies above $15Hz$ is expected to significantly mitigate noise of the static data but not for the dynamic.

Table 14: The standard deviation σ of the raw and filtered acceleration measurements during the static phase of the trajectories, representing the noise in the measurements.

Path	Raw Accelerations (m/s^2)			Low pass Filtered accelerations (m/s^2)			Moving Average accelerations (m/s^2)		
	σ_x	σ_y	σ_z	σ_x	σ_y	σ_z	σ_x	σ_y	σ_z
diag1	0.136	0.421	0.157	0.035	0.049	0.054	0.016	0.018	0.019
diag2	0.108	0.435	0.150	0.021	0.050	0.026	0.018	0.020	0.015
diag3	0.126	0.435	0.151	0.030	0.053	0.028	0.021	0.019	0.015
diag4	0.120	0.448	0.148	0.023	0.052	0.028	0.010	0.022	0.016
diag5	0.134	0.432	0.154	0.029	0.055	0.027	0.010	0.025	0.010
diag7	0.110	0.422	0.144	0.022	0.043	0.026	0.016	0.017	0.013
diag8	0.145	0.420	0.151	0.031	0.047	0.032	0.013	0.022	0.011
diag9	0.136	0.433	0.151	0.044	0.093	0.043	0.028	0.027	0.012
path1	0.192	0.414	0.147	0.033	0.055	0.030	0.015	0.015	0.010
path2	0.170	0.404	0.142	0.042	0.061	0.033	0.023	0.023	0.013
path3	0.220	0.392	0.098	0.040	0.054	0.031	0.014	0.021	0.014
path5	0.346	0.394	0.106	0.086	0.070	0.041	0.028	0.017	0.014
path6	0.138	0.372	0.091	0.045	0.056	0.030	0.045	0.025	0.013
path7	0.181	0.401	0.094	0.064	0.049	0.028	0.039	0.015	0.011
path8	0.290	0.493	0.247	0.199	0.180	0.175	0.138	0.052	0.063
path9	0.475	0.404	0.174	0.200	0.180	0.101	0.166	0.066	0.030
path10	0.515	0.545	0.371	0.330	0.290	0.213	0.170	0.099	0.075
path11	0.416	0.493	0.335	0.267	0.240	0.232	0.204	0.088	0.102
path12	0.586	0.528	0.330	0.315	0.283	0.186	0.203	0.118	0.061
path13	0.136	0.419	0.090	0.051	0.077	0.033	0.050	0.035	0.021
path14	0.258	0.432	0.195	0.167	0.167	0.127	0.126	0.082	0.038
path15	0.299	0.423	0.123	0.143	0.118	0.064	0.076	0.046	0.020

Table 15: The standard deviation σ of the raw and filtered acceleration measurements during the dynamic phase of the trajectories, representing the noise in the measurements.

Path	Raw Accelerations (m/s^2)			Low pass Filtered accelerations (m/s^2)			Moving Average accelerations (m/s^2)		
	σ_x	σ_y	σ_z	σ_x	σ_y	σ_z	σ_x	σ_y	σ_z
diag1	0.778	0.783	0.473	0.517	0.487	0.336	0.168	0.166	0.108
diag2	0.768	0.810	0.474	0.508	0.478	0.316	0.179	0.139	0.110
diag3	0.740	0.784	0.450	0.480	0.474	0.312	0.168	0.147	0.099
diag4	0.988	1.190	0.948	0.660	0.922	0.669	0.211	0.267	0.230
diag5	1.006	1.109	0.985	0.672	0.852	0.686	0.211	0.229	0.244
diag7	1.772	1.446	1.034	1.011	0.914	0.643	0.371	0.259	0.205
diag8	1.715	1.455	1.040	0.909	0.842	0.605	0.260	0.264	0.203
diag9	1.709	1.513	1.142	1.029	0.944	0.679	0.404	0.277	0.202
path1	1.095	1.303	0.982	0.715	0.995	0.725	0.291	0.402	0.339
path2	1.074	1.274	0.998	0.697	0.959	0.728	0.284	0.384	0.324
path3	0.875	0.808	0.674	0.533	0.552	0.481	0.186	0.186	0.212
path5	0.843	0.812	0.667	0.538	0.579	0.473	0.191	0.191	0.131
path6	1.348	1.323	1.264	0.795	1.078	0.732	0.252	0.348	0.261
path7	1.342	1.331	1.229	0.814	1.069	0.725	0.266	0.358	0.257
path8	1.338	1.325	1.205	0.806	1.072	0.714	0.263	0.362	0.242
path9	2.162	2.463	1.465	1.209	1.771	0.912	0.333	0.545	0.270
path10	2.124	2.526	1.438	1.224	1.810	0.854	0.337	0.588	0.241
path11	2.119	2.542	1.448	1.228	1.830	0.889	0.387	0.595	0.254
path12	2.159	2.286	1.445	1.133	1.702	0.823	0.324	0.465	0.212
path13	2.056	2.331	1.408	1.134	1.748	0.835	0.307	0.466	0.213
path14	2.031	2.381	1.405	1.127	1.754	0.860	0.314	0.469	0.220
path15	1.326	1.269	1.395	0.726	1.027	0.729	0.274	0.368	0.237

Since the dynamic platform moves at a constant speed throughout the trajectories, a smoother acceleration profile was expected for the dynamic part. It is believed that some of the noise in the accelerations is also caused by the motors that rotate the wheels of the dynamic platform. The impact of the motor on accelerations is only applicable for the dynamic part of the trajectory. As a result, the Fourier transform of the static part of the trajectory cannot be used to identify the frequencies of the motors that further impact the accelerations. Moreover, the Fourier transforms of the dynamic part of the trajectories, do not clearly indicate frequencies that might correspond to the motion of the motor.

Furthermore, the two motors of the dynamic platform might operate at different frequencies depending on the velocity and direction of motion. If the dynamic platform moves in a straight line at a constant speed, the two motors should operate at the same frequency that does not change throughout the path. The Fourier transform of the dynamic part of the straight line path 'diag1' (shown in figure 108), does not indicate the frequency that might represent the frequency of the motors. Alternatively, in situations when the dynamic platform changes direction, the two motors operate at different frequencies, which are not necessarily constant throughout the whole path. Since it is not easy to identify the frequencies of the motors, an alternative method is used to further smooth the acceleration measurements.

An additional step at the pre-processing phase therefore, is data smoothing using the moving average, a method commonly used to smooth data and mitigate the noise in measurements. The moving average is determined by calculating the average over a sliding window of a predefined size. A sliding window of 10 measurements is used in this thesis, which corresponds to a duration of 0.2s, since the IMU data collection rate is 50Hz. The 0.2s sliding window is not expected to alter the data as the dynamic platform moves at low speeds, as shown in table 6, with a maximum speed of $0.741 m/s$. At this speed, the dynamic platform moves a distance of $0.148m$ within the $0.2s$ window. This is a distance within which no significant changes in the true acceleration of the dynamic platform are expected. Applying the moving average to the filtered accelerations, results in the acceleration profile shown in figure 110.

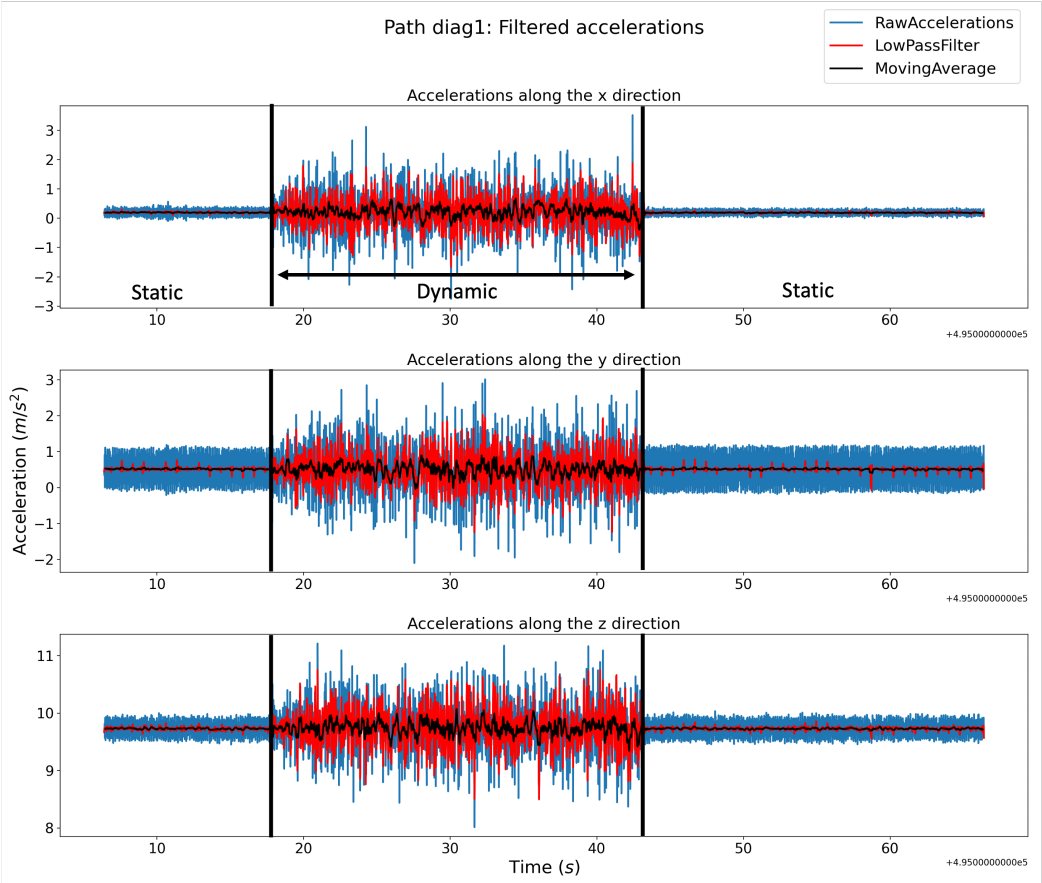


Figure 110: Filtered accelerations

As can be seen in figure 110, the moving average method succeeds in smoothing the accelerations, especially during the dynamic phase of the trajectory. The standard deviations of the two methods for both the static and dynamic parts of the trajectory are shown in tables 14 and 15 respectively. These tables illustrate numerically that although the low pass filter succeeds in smoothing the accelerations of the static part of the trajectory, the addition of the moving average method is required to smooth the accelerations in the dynamic part of the trajectory. Therefore, at the pre-processing phase the acceleration data are primarily filtered using the low-pass filter, and then furthered smoothed using the moving average with a sliding window of 10 measurements. The accelerations from this pre-processing phase are used to estimate the INS-based positioning solution once the system is initialised.

8.4. INS position estimation

INS position estimation uses the principle of DR, where changes in orientation and displacement information is used to estimate position once the system is initialised. Pre-processed accelerations, angular velocity measurements and calculated initial orientation, are used as inputs in the Madgwick filter, for orientation estimation. Alternatively, displacement information is usually provided through either a wheel encoder or accelerometer. In this case, a wheel encoder is not used and hence displacement information is provided by the accelerations. This is done to evaluate the performance of IMU position estimation without information from additional sources.

As illustrated in section 8.3, the accelerations appear noisy and biased compared to the angular velocity measurements. To investigate the impact of acceleration measurements, three IMU position estimation methods are implemented. Each method uses varying degrees of acceleration information. The first method does not use acceleration information at all, instead it uses the ground truth to estimate displacements. The second method uses the accelerations, without considering their direction, while the third method considers both magnitude and direction of accelerations. As the first approach relies solely on gyroscope measurements and acceleration measurements are gradually integrated in the second and third approaches, a comparison between the three, is used to evaluate the impact of acceleration and gyroscope measurements on IMU position estimation.

The first approach considers the attitude estimation of the Madgwick filter, in integration with a constant speed calculated using the truthing system. A schematic representation of this approach is illustrated in figure 111. The IMU accelerations and angular velocities are pre-processed prior to their use in the Madgwick filter, where they are used for orientation estimation as discussed in section 8.1.1. The heading of the IMU sensor is then calculated from the estimated orientation. Additionally, measurements from the camera-based truthing system, are used to calculate the distance travelled in the local frame. The derived heading and distance travelled estimates are used in the method of DR to calculate the position of the IMU. A more detailed explanation of this process is

presented below.

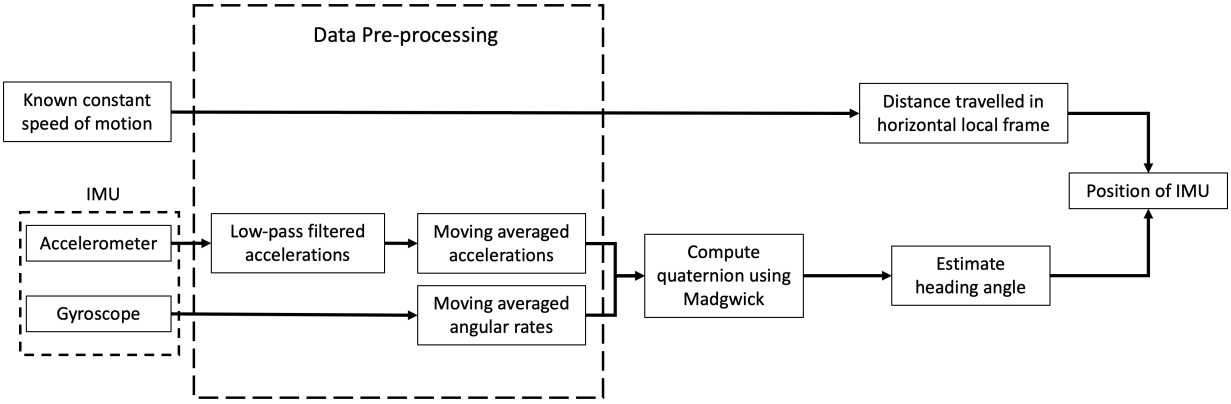


Figure 111: Approach 1 used to calculate the INS-based solution.

This is done to evaluate the performance of gyroscope measurements without the impact of accelerations and to replicate the effect of a wheel encoder, which is commonly used when considering wheeled agents. A wheel encoder measures the number of wheel rotations and converts it into a measurement of the distance travelled, which can generally result in more accurate displacement estimation compared to the use of acceleration measurements. In this case, since the dynamic platform moves at a constant speed along the trajectory, the positions provided by the camera system, are used to calculate the average speed (s). The speed and time intervals between IMU epoches Δt are used to estimate the distance travelled between consecutive IMU epoches ($distT$) as shown in equation (50). The position of the robot is determined using the principle of DR, where knowing the previous position estimate, the current position is calculated by applying the displacement in the orientation defined by the Madgwick filter.

$$distT = s\Delta t \tag{50}$$

The quaternions calculated by the Madgwick filter are used to calculate Euler angles of roll (φ), pitch (ϑ) and yaw (ψ) as shown in (51), which represent the orientation of the IMU in the room. The yaw angle ψ, corresponds to the heading of the dynamic platform.

$$\boldsymbol{\theta}\{\boldsymbol{q}\} = \begin{bmatrix} \varphi \\ \vartheta \\ \psi \end{bmatrix} = \begin{bmatrix} \arctan\left(\frac{2\cdot(q_w\cdot q_x + q_y\cdot q_z)}{1-2\cdot\left(q_x^2+q_y^2\right)}\right) \\ \arcsin\left(2\cdot(q_w\cdot q_y - q_z\cdot q_x)\right) \\ \arctan\left(\frac{2\cdot(q_w\cdot q_z + q_x\cdot q_y)}{1-2\cdot\left(q_y^2+q_z^2\right)}\right) \end{bmatrix} \tag{51}$$

The distance travelled ($distT$) is used in integration with the heading angle ψ, to calculate displacements in the x and y directions at time t, $disp_{x,t}$ and $disp_{y,t}$ respectively as shown in equation (52).

$$disp_{x,t} = distT cos(\psi) \tag{52a}$$

$$disp_{y,t} = distT sin(\psi) \tag{52b}$$

$$\tag{52c}$$

The displacements in the two directions ($disp_{x,t}, disp_{y,t}$) are added to the position of the previous IMU epoch, X_{t-1}, to calculate the current IMU position, X_t, as shown in figure 112 and equation (53).

$$X_t = \begin{bmatrix} x_{t-1} + \text{disp}_{x,t} \\ y_{t-1} + \text{disp}_{y,t} \end{bmatrix} \tag{53}$$

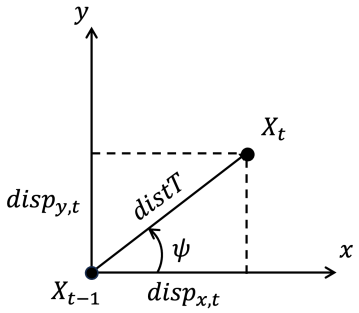


Figure 112: Using the distance travelled and heading angle to estimate the INS-based position.

The second approach, shown in figure 113, is similar to the first, the main difference being that this approach calculates the distance travelled ($distT$) using acceleration measurements. As shown in figure 113 the orientation calculated by the Madgwick filter is used to transform accelerations from the IMU frame to the local frame. The distance travelled is then defined using the horizontal accelerations in the local frame and the final position estimation is calculated using the method of DR.

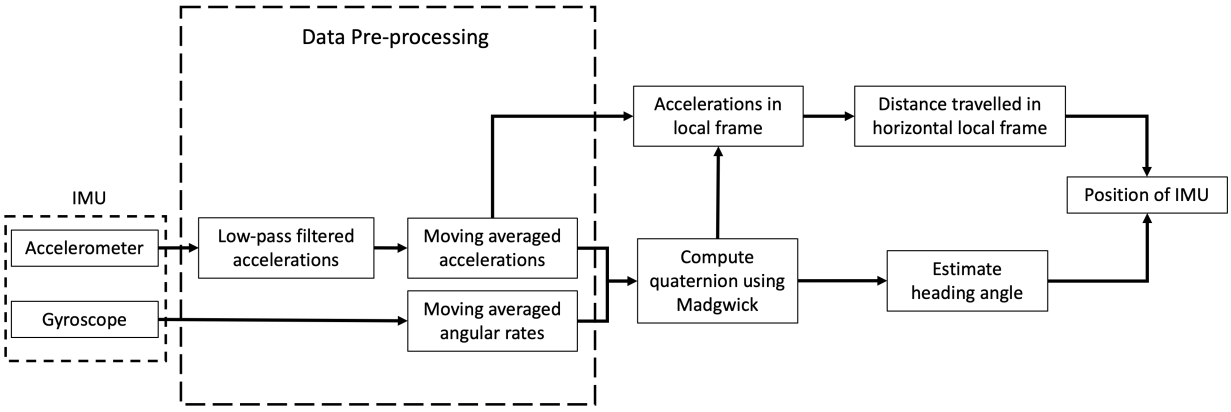


Figure 113: Approach 2 used to calculate the INS-based solution.

Although acceleration measurements are used in the calculation, their direction is not considered. The quaternions provided by the Madgwick filter, represent the orientation of the IMU and are used to transform the accelerations from the IMU to local frame. For every IMU epoch, equation (54) is used to transform the three orthogonal accelerations from the IMU frame to the local frame using the calculated quaternion q. In equation (54) α^{IMU} is the quaternion representation of accelerations in the IMU frame and α^{L} are the accelerations in the local frame.

$$\alpha^{IMU} = \begin{bmatrix} 0 & \alpha_x & \alpha_y & \alpha_z \end{bmatrix} \tag{54a}$$

$$\alpha^{L} = q \otimes \alpha^{IMU} \otimes q^{*} \tag{54b}$$

The horizontal accelerations in the local frame are then used to calculate displacements in the x and y directions in the local frame. The displacement calculation is presented by

equation (55), where the double integration of the accelerations is used to calculate the displacements. The displacement in a specific direction at time t, is expressed as $disp_{dir,t}$, while the velocity in the specific direction as $v_{dir,t}$. The calculation of displacement and velocity at every epoch is dependent on the velocity of the previous IMU epoch at $t-1$, the acceleration measurement of the current epoch t and the time difference between the two epoches ($\Delta t = t - (t-1)$).

$$disp_{dir,t} = v_{dir,t-1} \cdot \Delta t + a_{dir,t} \cdot \frac{\Delta t^2}{2} \tag{55a}$$

$$v_{dir,t} = v_{dir,t-1} + a_{dir,t} \cdot \Delta t \tag{55b}$$

The distance travelled ($distT$) is then calculated as the euclidean norm of the two displacements, as shown in equation (56).

$$distT = \sqrt{disp_{x,t}^2 + disp_{y,t}^2} \tag{56}$$

Having calculated the distance travelled using the accelerations, the principle of DR is applied in the same way as in the first approach.

The third approach incorporates acceleration measurements by considering both their magnitude and direction. A schematic of the third approach is illustrated in figure 114, which illustrates the same process of calculating horizontal displacements in the local frame, as in approach 2. The difference between approach 2 and 3 is that in the latter the displacements ($disp_{x,t}, disp_{y,t}$) are directly added to the previous known position as shown by equation (53) to calculate the position at time t. In this way the direction of the accelerations is also taken into consideration together with its magnitude. Alternatively, as shown in figure 114, Madgwick orientation is only used to transform accelerations from the IMU to the local frame and not to estimate heading as in approaches 1 and 2 (figures 111 and 113 respectively).

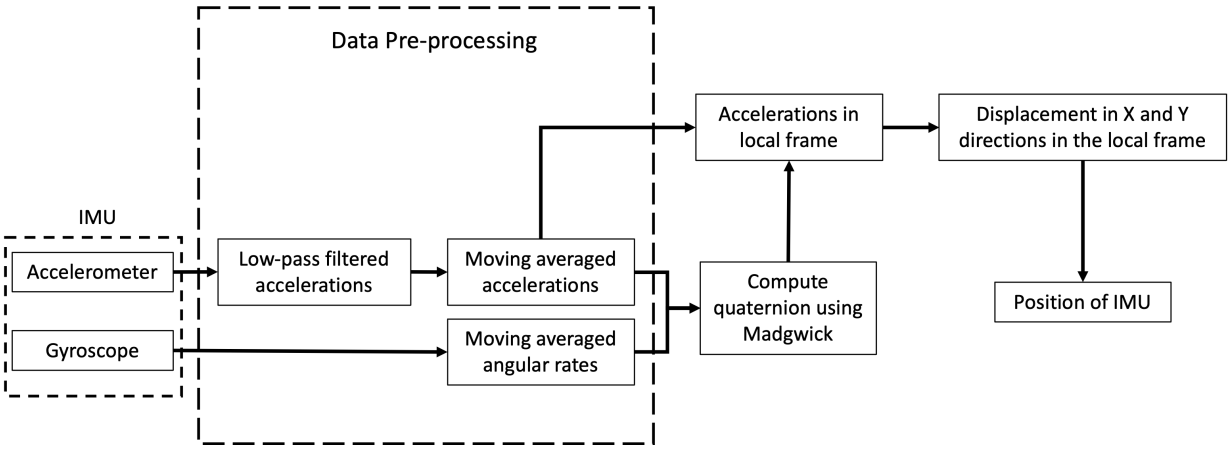


Figure 114: Approach 3 used to calculate the INS-based solution.

It is interesting to note that in all three approaches, the angular rates are also smoothed using the method of moving average. This is because the method of moving average reduces the number of measurements by the size of the sliding window (in our case by 10 measurements). When the accelerations go through the process of moving average, the number of observations reduces by 10, and hence the smoothed accelerations at time t do not match the angular velocities at time t. Angular velocities are also smoothed using the moving average with a sliding window of 10, so that acceleration and angular velocity measurements match.

The three approaches described in this section are implemented and the their results are discussed in the following section.

8.5. Comparison of the three approaches

The three approaches are implemented to estimate the position of the dynamic platform using the pre-processed IMU measurements. The position estimation of the three approaches for one of the circular paths, 'path15' are illustrated in figure 115.

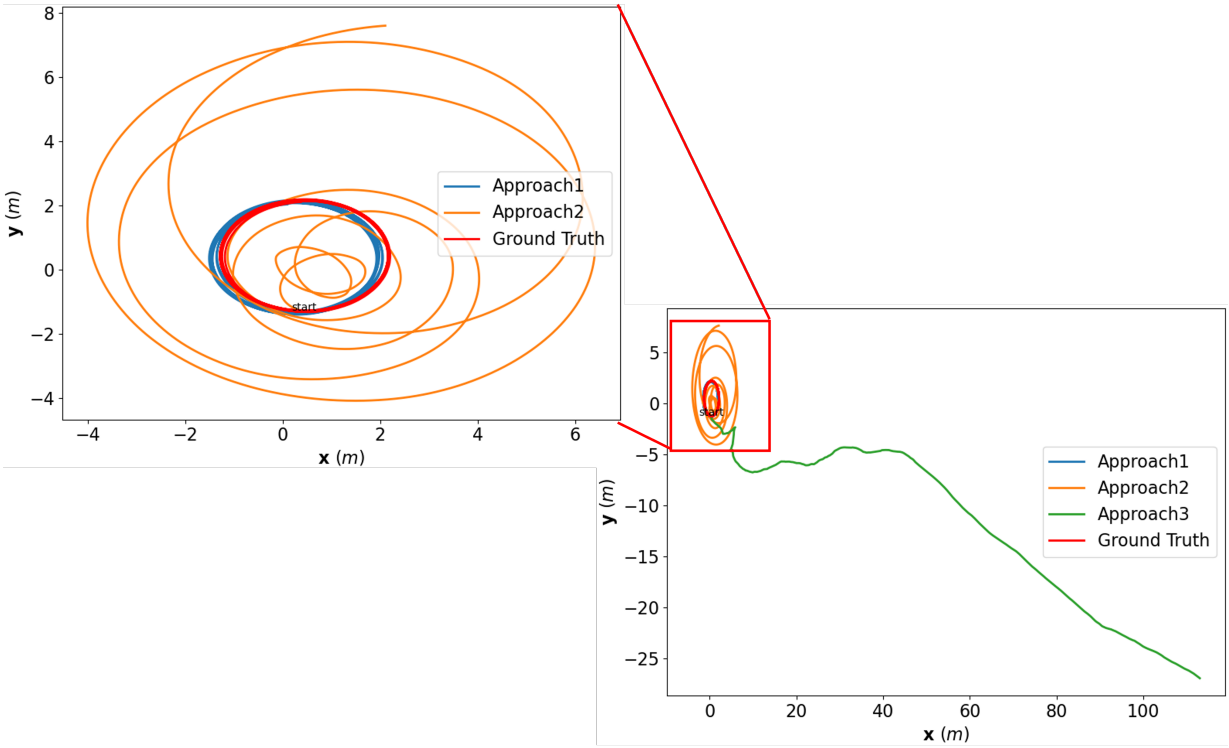


Figure 115: The three approaches used for the calculation of IMU-only solution of path15.

In figure 115 the positioning solution computed by the three approaches are shown against the ground truth illustrated by the red trajectory. Figure 115 demonstrates that the positioning solution is compromised as the input from accelerations increases. The most accurate solution out of the three is provided by the first approach (illustrated by the blue line), where the truthing system is used to calculate the distance travelled $distT$. Moving through the second and third approaches (in orange and green respectively), where the input from accelerations is gradually increasing, leads to a reduction in accuracy. The least accurate solution out of the three is given by the third approach, as it results in the trajectory that deviates the most from the truth. In the third approach, the magnitude and direction of accelerations are considered, making this approach the one that has the highest number of inputs from accelerations amongst the three.

To numerically evaluate the performance of the three approaches, the resulting position errors are analysed. Position errors are calculated for a number of straight line and circular

paths, using each of the approaches. Position errors at different percentiles (68%, 95% and 99.5%) are calculated to compare the performance of the three approaches. The calculated errors are presented in table 16.

As the INS positioning solution is known to drift over time, the percentile calculation is done using errors for the same trajectory duration, in the presented paths. The trajectory duration used is $24s$ as this is the duration of the shortest path in table 16, diag1. Table 16, shows that for all the paths, approach 3 results in the largest error for all the percentiles, amongst the three approaches. Additionally, comparing the percentile errors amongst different paths, shows that approach 3 results in comparable errors of around $6m - 8.8m$ at 68%, $10.5m - 18m$ at 95% and $11.6m - 20m$ at 99.7%, during the $24s$ path duration.

An additional measure of performance presented in table 16, is the time taken for the solution to drift a distance of $2m$. This corresponds to the accuracy requirement of the IPS defined in section 2.3.3, as positioning errors should not exceed $2m$ for safety critical applications. As shown in table 16, approach 1 does not reach the $2m$ threshold for any of the paths. Moreover, when comparing approach 2 and 3, the latter reaches the $2m$ threshold faster than the former. This is shown by the shorter time taken by approach 3 to reach a position error of $2m$ compared to approach 2.

In addition to percentile error and the time required to reach the $2m$ threshold, the drift rates using the three approaches is calculated. To calculate the drift rate, a second order polynomial is fitted through the calculated position errors over time. The drift rate is equivalent to the linear component of the fitted polynomial, which represents the way the position error varies with time. One of the straight line paths is used as an example to illustrate the way position error varies with time when using the three approaches (shown in figure 116). Figure 116 illustrates that a second order polynomial is a good fit for approaches 2 and 3, however approach 1, shown in purple, does not follow a second order polynomial. This trend is common amongst most dynamic paths.

The position errors of approach 1 oscillate, while for some of the cases illustrate a slight increase over time, as shown from the purple line in figure 116. This is because in approach 1 the accelerations of the IMU are not considered, and the true speed is used instead to

approximate the displacements. In this way the position errors of approach 1 correspond to the errors caused by the errors in the gyroscope measurements. As discussed in section 8.2, the angular velocities from the gyroscope do not have significant levels of bias and noise. This is also demonstrated here as when the solution relies only on the angular measurement of the IMU, the solution does not significantly drift away from the truth. The results presented in table 16, show that approach 1 results in the smallest position errors between $0.05m - 0.8m$ in 99.7% of the times, while it also results in the smallest drift rates $0.004m/s - 0.02m/s$. This emphasises the fact that the main source of errors in the INS-based solution stems from accelerations.

The comparison between the three approaches suggests that the main source of position error from a consumer grade IMU, stems from accelerations. Wheel encoders provide more accurate estimation of travelled distance compared to an accelerometer, and are commonly used in the cases of wheeled agents. The use of wheel encoders, eliminates the input from accelerations, however, it introduces an additional sensor to the system. In this thesis, an alternative approach is proposed (in section 9.2.2) to correct the INS-based speed and distance estimation. This method is applicable in situations where an INS is integrated with a ranging system, such as UWB. In such cases, the range measurements are used to approximate the speed of the dynamic agent and correct velocities derived from IMU accelerations.

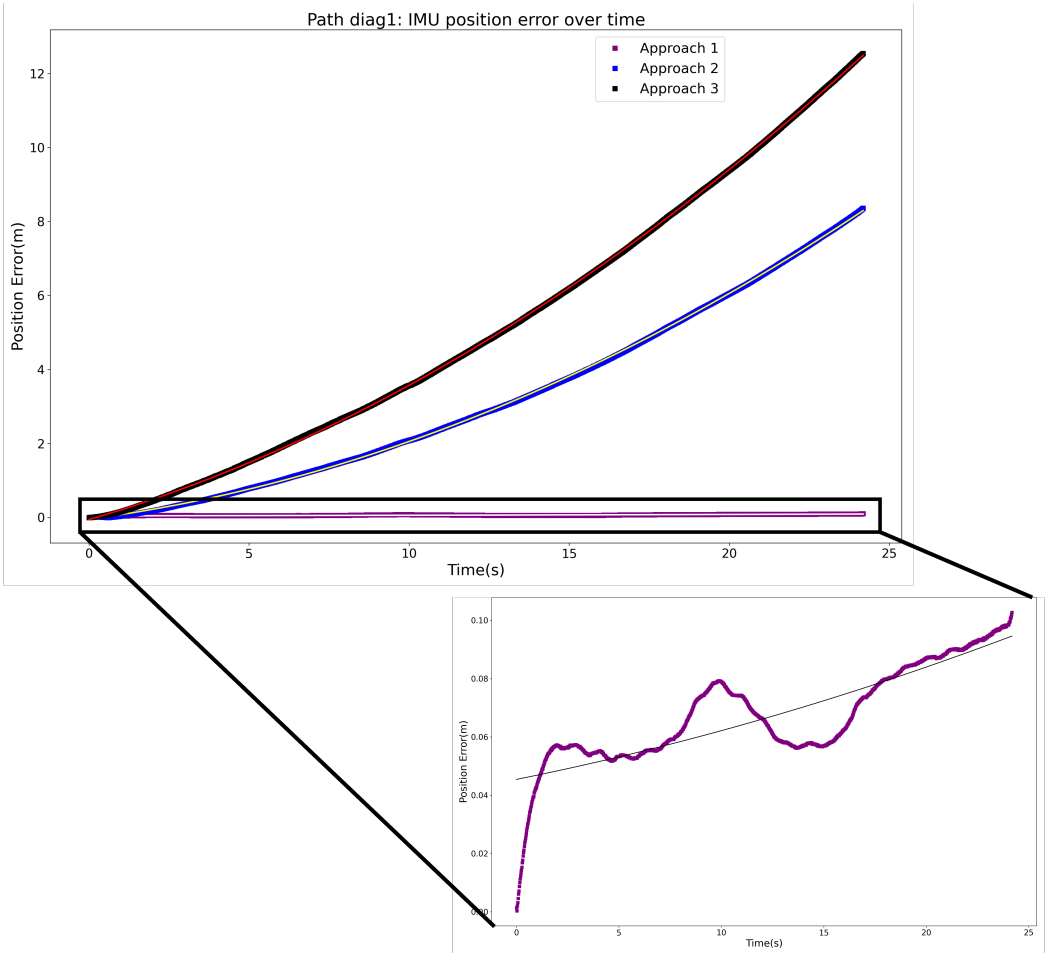


Figure 116: The IMU solution drift when using the three approached for the straight line diag1.

Table 16: Percentiles of measurement errors during the first $24s$ of the trajectories, time required to reach the $2m$ position error and position drift rates.

Path	Approach	Percentile position errors (m)			Time to reach $2m$ position error (s)	Position drift rate (m/s)
		68%	95%	99.7%		
diag1	Approach 1	0.076	0.094	0.098	-	0.009
	Approach 2	4.306	7.521	8.171	9.822	0.108

	Approach 3	6.960	11.423	12.310	6.342	0.244
diag2	Approach 1	0.072	0.092	0.095	-	0.013
	Approach 2	4.280	7.425	8.059	10.340	0.084
	Approach 3	7.055	11.545	12.424	6.860	0.216
diag3	Approach 1	0.048	0.062	0.065	-	0.008
	Approach 2	3.384	6.442	7.225	11.640	0.064
	Approach 3	6.148	10.549	11.590	6.981	0.194
path12	Approach 1	0.396	0.457	0.458	135.538	0.004
	Approach 2	2.666	3.683	4.653	12.840	0.122
	Approach 3	8.767	18.478	20.151	2.679	0.910
path13	Approach 1	0.578	0.793	0.822	-	0.020
	Approach 2	1.752	2.186	2.222	19.760	0.117
	Approach 3	6.005	11.550	12.318	5.348	0.535

8.6. Summary

This chapter investigated the impact of the IMU measurements on the INS solution. The IMU measurements used to calculate the position estimates are the accelerations and angular velocities in the three orthogonal directions of the IMU. This chapter begun with the analysis of the raw data, which led to the pre-processing methods used for acceleration data. Acceleration data are initially filtered using a low-pass filter with a cut-off frequency of $15Hz$ and are further smoothed using the moving average method with a sliding window of 10 measurements. The angular velocities from the gyroscope are also smoothed using the process of moving average, in to order to match the number of filtered and smoothed accelerations.

The pre-processed IMU data were then used to calculate the INS position estimation using three approaches. These approaches were used to investigate the impact of progressively increasing the input of accelerations in the calculated solution. The solution of the three approaches were compared both visually and numerically emphasising the

fact that the main source of error in an INS positioning solutions stems from acceleration measurements.

An INS, is a system that provides short term stability in the positioning solution, however, it is characterised by long term instability since its positioning solution drifts with time. INS is a system to integrate with UWB to improve the availability and accuracy of the UWB-standalone positioning solution. Moreover, knowing that the limitations of the INS-based solution stem from accelerations, leads to the development of a method that aims to reduce the impact of accelerations on the integrated solution. This method uses UWB measurements to estimate the speed of the dynamic platform and correct the velocities derived from the accelerations of the IMU. It is a method that can be implemented when the INS is integrated with a ranging system, like UWB, and will be introduced in the following chapter.

Having identified the strengths and weaknesses of the UWB and INS in chapters 7 and 8 respectively, the two systems compliment one another. The next chapter, presents the integration of the two systems in an attempt to develop a high-accuracy positioning system able to support indoor positioning and navigation.

Chapter 9

Ultra-Wideband/Inertial Measurement Unit integration: Models and performance

The collected UWB measurements were analysed in chapter 7 and used to estimate UWB standalone positions. Alternatively, IMU measurements were analysed in chapter 8 and used to calculate the INS solution once the system was initialised. As the results showed, the UWB system exhibits long term stability, while the INS has short term stability. Moreover, the sampling rate of the IMU is higher than the UWB, allowing the INS to bridge potential gaps in the UWB solution caused by the lower data rate and erroneous range measurements. The two systems complement each other and their integration is advantageous in improving both positioning accuracy and availability of the UWB standalone system.

This chapter presents the integration of the two sensors, and incorporates the findings of chapters 7 and 8. The integration method used is the ESKF using quaternion kinematics, presented by Sola (2017), due to its superior performance over an EKF, as discussed in chapter 5. The implementation of the ESKF considers the findings of chapter 7 and 8, by either applying the relevant corrections to the measurements or creating additional measurements to use in the integration.

The UWB measurement corrections depend on the direction of motion with respect to the anchor, as calculated in section 7.2.2, and are applied to correct the raw range measurements. The corrected measurements are further filtered prior to integration, to remove any measurements with significant errors. When considering the IMU measurements, it was shown in chapter 8, that accelerations are the main source of error in an

INS solution. As a result, this chapter proposes a method of correcting the errors of the IMU accelerations which relies on speeds estimated using the UWB measurements. The integration presented in this chapter is a tight (or measurement domain) integration, to ensure that all UWB range measurements are used, even if the range measurements are not enough to estimate a UWB position. This should result in better accuracy and availability performance compared to a loose integration, as it will reduce the interval between UWB measurement updates, reducing the impact of INS solution drift.

This chapter presents the proposed implementation of the ESKF and results obtained using empirical data. It begins with a detailed explanation of the ESKF used, in section 9.1. Section 9.2, explains the implementation of the ESKF and additional components proposed in this thesis, based on the analysis in chapters 7 and 8. Section 9.3 presents the results of the integration using real UWB and IMU measurements. Finally, section 9.4, summarises the main findings of this chapter.

9.1. Mathematical model of ESKF

The integration method implemented in this thesis is the ESKF using quaternion kinematics, presented by Sola (2017). In an error state filter, the true state is assumed to be composed of two parts, nominal state and error state. The latter considers the errors of the nominal state with the true state being a composition of the nominal and error states as shown in table 17. The ESKF solves for the error state which, as mention in chapter 5, is characterised by linear dynamics. The estimated error state is then added to the predicted nominal state to estimate the true state. Table 17 describes the state variables used in this thesis, nominal and error states and their composition to estimate the true state. The state variable are the position and velocity of the dynamic agent in three orthogonal directions and orientation described using a quaternion.

Table 17: The variable used in the ESKF

Magnitude	True	Nominal	Error	Composition
Full state	$\mathbf{x}_t$	$\mathbf{x}$	$\delta\mathbf{x}$	$\mathbf{x}_t = \mathbf{x} \oplus \delta\mathbf{x}$
Position	$\mathbf{p}_t$	$\mathbf{p}$	$\delta\mathbf{p}$	$\mathbf{p}_t = \mathbf{p} + \delta\mathbf{p}$
Velocity	$\mathbf{v}_t$	$\mathbf{v}$	$\delta\mathbf{v}$	$\mathbf{v}_t = \mathbf{v} + \delta\mathbf{v}$
Quaternion	$\mathbf{q}_t$	$\mathbf{q}$	$\delta\mathbf{q}$	$\mathbf{q}_t = \mathbf{q} \otimes \delta\mathbf{q}$

9.1.1. Nominal state

The nominal state uses kinematic equations to describe system motion. It uses the observations of an IMU to predict the next state, without considering any errors and noises. The kinematic equations describing the nominal state are shown in equations (57).

$$\dot{\mathbf{p}} = \mathbf{v} \tag{57a}$$

$$\dot{\mathbf{v}} = \mathbf{R}\left\{\boldsymbol{q}\right\}\mathbf{a}_m + \mathbf{g} \tag{57b}$$

$$\dot{\mathbf{q}} = \frac{1}{2}\mathbf{q} \otimes \boldsymbol{\omega}_m \tag{57c}$$

where: $\mathbf{a}_m$ and $\boldsymbol{\omega}_m$ are the measurements of acceleration and angular velocities respectively.

Considering discrete time intervals, kinematic equations (57) become:

$$\boldsymbol{p}_t = \boldsymbol{p}_{t-1} + \boldsymbol{v}_{t-1} \cdot \Delta t + \left(\boldsymbol{R}\left\{\boldsymbol{q}_{t-1}\right\} \cdot \boldsymbol{a}_t + \boldsymbol{g}\right) \cdot \frac{\Delta t^2}{2} \tag{58a}$$

$$\boldsymbol{v}_t = \boldsymbol{v}_{t-1} + \left(\boldsymbol{R}\left\{\boldsymbol{q}_{t-1}\right\} \cdot \boldsymbol{a}_t + \boldsymbol{g}\right) \cdot \Delta t \tag{58b}$$

$$\boldsymbol{q}_t = \boldsymbol{q}_{t-1} \otimes \boldsymbol{q}\left(\boldsymbol{\omega}_t \cdot \Delta t\right) \tag{58c}$$

where the subscripts t and $t-1$ show how the state changes from time $t-1$ to t, $\boldsymbol{R}\{\boldsymbol{q}\}$ is the rotation matrix associated with quaternion $\boldsymbol{q}$, as shown in (59) and $\boldsymbol{q}(v)$ is the quaternion associated with rotation $\boldsymbol{v}$ with Euler angles, yaw ψ, pitch θ and roll ϕ, as shown in (60).

$$\boldsymbol{R}\{\boldsymbol{q}\} = \begin{bmatrix} q_w^2 + q_x^2 - q_y^2 - q_z^2 & 2(q_x q_y - q_w q_z) & 2(q_x q_z + q_w q_y) \\ 2(q_x q_y + q_w q_z) & q_w^2 - q_x^2 + q_y^2 - q_z^2 & 2(q_y q_z - q_w q_x) \\ 2(q_x q_z - q_w q_y) & 2(q_y q_z + q_w q_x) & q_w^2 - q_x^2 - q_y^2 + q_z^2 \end{bmatrix} \tag{59}$$

$$\boldsymbol{q}(\boldsymbol{v}) = \begin{bmatrix} \cos(\phi/2)\cdot\cos(\theta/2)\cdot\cos(\psi/2) + \sin(\phi/2)\cdot\sin(\theta/2)\cdot\sin(\psi/2) \\ \sin(\phi/2)\cdot\cos(\theta/2)\cdot\cos(\psi/2) - \cos(\phi/2)\cdot\sin(\theta/2)\cdot\sin(\psi/2) \\ \cos(\phi/2)\cdot\sin(\theta/2)\cdot\cos(\psi/2) + \sin(\phi/2)\cdot\cos(\theta/2)\cdot\sin(\psi/2) \\ \cos(\phi/2)\cdot\cos(\theta/2)\cdot\sin(\psi/2) - \sin(\phi/2)\cdot\sin(\theta/2)\cdot\cos(\psi/2) \end{bmatrix} \tag{60}$$

Equation (58), illustrates the way the state variables evolve over time from x_{k-1} to x_k using kinematic equations.

9.1.2. Error state

The error state represents model imperfections and process noises. The linearised dynamics of the error state are used to represent the error state kinematics of the system. The error state kinematics are shown in equation (61), and their derivation can be found in Sola (2017).

$$\dot{\delta}\mathbf{p} = \delta\mathbf{v} \tag{61a}$$

$$\dot{\delta}\mathbf{v} = -\boldsymbol{R}\{\boldsymbol{q}\}[\mathbf{a}_m]_\times \delta\boldsymbol{\theta} \tag{61b}$$

$$\dot{\delta}\boldsymbol{\theta} = -\boldsymbol{R}\{\boldsymbol{q}\}[\boldsymbol{\omega}_m]_\times \delta\boldsymbol{\theta} \tag{61c}$$

where $[a]_\times$ is a skew symmetric matrix operator defined by:

$$[\boldsymbol{a}]_\times = \begin{bmatrix} 0 & -a_z & a_y \\ a_z & 0 & -a_x \\ -a_y & a_x & 0 \end{bmatrix} \tag{62}$$

The error state in discrete time intervals is given by (63):

$$\delta \boldsymbol{p} = \delta \boldsymbol{p}_{t-1} + \delta \boldsymbol{v}_{t-1} \cdot \Delta t \tag{63a}$$

$$\delta \boldsymbol{v} = \delta \boldsymbol{v}_{t-1} + \left(-\boldsymbol{R}\left\{\boldsymbol{q}_{t-1}\right\} \cdot [\boldsymbol{a}_{t-1}]_{\times} \cdot \delta \boldsymbol{\theta}_{t-1}\right) \cdot \Delta t + \boldsymbol{v_i} \tag{63b}$$

$$\delta \dot{\boldsymbol{\theta}} = \boldsymbol{R}\left\{\boldsymbol{\omega}_{t-1} \cdot \Delta t\right\}^{T} \cdot \delta \boldsymbol{\theta}_{t-1} + \boldsymbol{\theta_i} \tag{63c}$$

where $\boldsymbol{v_i}$ and $\boldsymbol{\theta_i}$ are small perturbations applied to the velocity and orientation estimates. These perturbations are normally distributed with a mean of zero and variances of:

$$\mathbf{V_i} = \sigma^2_{\tilde{\mathbf{a}}_n} \Delta t^2 \mathbf{I} \tag{64a}$$

$$\Theta_{\mathbf{i}} = \sigma^2_{\tilde{\boldsymbol{\omega}}_n} \Delta t^2 \mathbf{I} \tag{64b}$$

where $\sigma^2_{\tilde{\mathbf{a}}_n}$ and $\sigma^2_{\tilde{\boldsymbol{\omega}}_n}$ are the variance in accelerations and angular velocities, respectively.

9.1.3. State prediction step

State prediction uses the the previous time estimate to predict the nominal and error states. The prediction of the nominal state, shown in equations (65) (Brigadnov et al. 2023), represent the way the posteriori state estimate of the previous epoch (x^{+}_{t-1}) relates to the priori (x^{-}_{t}) prediction of the state of the current epoch.

$$\boldsymbol{p}^{-}_{t} = \boldsymbol{p}^{+}_{t-1} + \boldsymbol{v}^{+}_{t-1} \cdot \Delta t + \left(\boldsymbol{R}\left\{\boldsymbol{q}^{+}_{t-1}\right\} \cdot \boldsymbol{a}_t + \boldsymbol{g}\right) \cdot \frac{\Delta t^2}{2} \tag{65a}$$

$$\boldsymbol{v}^{-}_{t} = \boldsymbol{v}^{+}_{t-1} + \left(\boldsymbol{R}\left\{\boldsymbol{q}^{+}_{t-1}\right\} \cdot \boldsymbol{a}_t + \boldsymbol{g}\right) \cdot \Delta t \tag{65b}$$

$$\boldsymbol{q}^{-}_{t} = \boldsymbol{q}^{+}_{t-1} \otimes \boldsymbol{q}\left(\boldsymbol{\omega}_t \cdot \Delta t\right) \tag{65c}$$

The covariance matrix of the predicted state also needs to be determined. As the ESKF estimates the error state, the priori error covariance matrix (P^{-}) is related to the Jacobian of the error state with respect to the errors (F_x) and perturbations (F_i), as shown in equation (66). The Jacobian matrices with respect to the errors and perturbations (F_x, F_i)

and the covariance matrices of the perturbations (Q_i) are shown in equations (67), (68), (69).

$$\boldsymbol{P}_t^- = \boldsymbol{F}_x \boldsymbol{P}_{t-1}^+ \boldsymbol{F}_x^T + \boldsymbol{F}_i \boldsymbol{Q}_i \boldsymbol{F}_i^T \tag{66}$$

$$\boldsymbol{F}_x = \left.\frac{\partial f}{\partial \delta \mathbf{x}}\right|_{\mathbf{x}} = \begin{bmatrix} \mathbf{I}_{3\times3} & \mathbf{I}_{3\times3} \cdot \Delta t & 0 \\ 0 & \mathbf{I}_{3\times3} & -\left[\boldsymbol{R}\left\{\hat{\boldsymbol{q}}_t^+\right\} \cdot \boldsymbol{a}_{t-1}\right]_\times \cdot \Delta t \\ 0 & 0 & \boldsymbol{R}\left\{\hat{\boldsymbol{q}}_{t-1}^+\right\} \cdot (\boldsymbol{\omega} \cdot \Delta t)^T \end{bmatrix} \tag{67}$$

$$\boldsymbol{F}_i = \left.\frac{\partial f}{\partial \delta \mathbf{i}}\right|_{\mathbf{x}} = \begin{bmatrix} \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} \\ \boldsymbol{I}_{3\times3} & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & \boldsymbol{I}_{3\times3} \end{bmatrix} \tag{68}$$

$$\boldsymbol{Q}_i = \begin{bmatrix} \mathbf{V_i} & 0 \\ 0 & \Theta_\mathbf{i} \end{bmatrix} \tag{69}$$

Having calculated the nominal state and error covariance matrix, the next step is to update the predicted state using measurements from additional sensors.

9.1.4. Measurement update step

The measurement update step corrects the error state using measurements from additional sensors. For the correction of the errors, the Kalman gain and innovation need to be determined. The Kalman gain acts as a weight applied to the difference between the prediction and measurement step, known as the innovation, and it is a representation of the level of trust in the two stages. In order to calculate the Kalman gain and innovation, a relationship between the measurements and state needs to be defined, such that:

$$y = h\left(x_t\right) + v \tag{70}$$

In equation (70), $h()$ describes how the true state x_t is related to the measurement y, and v is a Gaussian noise, $v \sim \mathcal{N}\{0, \mathbf{V}\}$, with V representing the measurement covariance. The calculation of the Kalman gain (K), error estimate ($\delta\hat{\boldsymbol{x}}_t$) and updated covariance matrix

($\boldsymbol{P}_t$), are shown in equations (71), (72) and (73), respectively. The innovation representing the difference between the measurement and prediction is represented by $\left(\boldsymbol{y}_t - \hat{\boldsymbol{x}}_t^-\right)$ in (72).

$$\boldsymbol{K}_t = \boldsymbol{P}_t^- \boldsymbol{H}_t^T \left(\boldsymbol{H}_t \boldsymbol{P}_t^- \boldsymbol{H}_t^T + \boldsymbol{V}\right)^{-1} \tag{71}$$

$$\delta\hat{\boldsymbol{x}}_t^+ = \boldsymbol{K}_t \left(\boldsymbol{y}_t - \hat{\boldsymbol{x}}_t^-\right) \tag{72}$$

$$\boldsymbol{P}_t^+ = \left(\boldsymbol{I} - \boldsymbol{K}_t \boldsymbol{H}_t\right) \boldsymbol{P}_t^- \tag{73}$$

As the solution is for errors in the estimate, H is the observation matrix and is defined are the Jacobian of $h()$ with respect to the error state as shown in equation 74.

$$\mathbf{H} \equiv \left.\frac{\partial h}{\partial \delta \mathbf{x}}\right|_{\mathbf{x}} \tag{74}$$

H can be computed using the chain rule, such that:

$$\mathbf{H} \triangleq \left.\frac{\partial h}{\partial \delta \mathbf{x}}\right|_{\mathbf{x}} = \left.\frac{\partial h}{\partial \mathbf{x}_t}\right|_{\mathbf{x}} \left.\frac{\partial \mathbf{x}_t}{\partial \delta \mathbf{x}}\right|_{\mathbf{x}} = \mathbf{H}_{\mathbf{x}} \mathbf{X}_{\delta \mathbf{x}} \tag{75}$$

where as shown in Sola (2017),

$$\mathbf{X}_{\delta \mathbf{x}} = \triangleq \left.\frac{\partial \mathbf{x}_t}{\partial \delta \mathbf{x}}\right|_{\mathbf{x}} = \begin{bmatrix} \mathbf{I}_6 & 0 & 0 \\ 0 & \mathbf{Q}_{\delta \boldsymbol{\theta}} & 0 \\ 0 & 0 & \mathbf{I}_9 \end{bmatrix} \tag{76}$$

$$\mathbf{Q}_{\delta \boldsymbol{\theta}} = \frac{1}{2} \begin{bmatrix} -q_x & -q_y & -q_z \\ q_w & -q_z & q_y \\ q_z & q_w & -q_x \\ -q_y & q_x & q_w \end{bmatrix} \tag{77}$$

and $\mathbf{H}_{\mathbf{x}}$ is the Jacobian of $h()$ with respect to the true state variables. Since at the current state, the error state ($\delta \boldsymbol{x}$) is equal to zero, the $\mathbf{H}_{\mathbf{x}}$ Jacobian matrix is evaluated at the

nominal state. The $\mathbf{H_x}$ matrix, depends on the sensor used and the way the measurements relate to the state.

In this thesis, UWB range measurements are used and the $h()$ function describing the relationship between the measurements d_{meas} and the state variables is given by:

$$d_{meas} = \sqrt{(x_i - x_x)^2 + (y_i - x_y)^2 + (z_i - x_z)^2} \tag{78}$$

where (x_i, y_i, z_i) are the coordinates of the anchor from which range measurements are collected and (x_x, x_y, x_z) are the coordinates of the dynamic tag (the state variables). For every measurement used in the integration, the $\mathbf{H_x}$ matrix is:

$$\mathbf{H_x} = \begin{bmatrix} \frac{\partial d_{meas}}{\partial x_x} & \frac{\partial d_{meas}}{\partial x_y} & \frac{\partial d_{meas}}{\partial x_z} & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{79}$$

where

$$\frac{\partial d_{meas}}{\partial x_x} = \frac{x_x - x_i}{\sqrt{(x_x - x_i)^2 + (x_y - y_i)^2 + (x_z - z_i)^2}} \tag{80a}$$

$$\frac{\partial d_{meas}}{\partial x_y} = \frac{x_y - y_i}{\sqrt{(x_x - x_i)^2 + (x_y - y_i)^2 + (x_z - z_i)^2}} \tag{80b}$$

$$\frac{\partial d_{meas}}{\partial x_z} = \frac{x_z - z_i}{\sqrt{(x_x - x_i)^2 + (x_y - y_i)^2 + (x_z - z_i)^2}} \tag{80c}$$

Once the measurements are used to estimate the error in the state, the true state is estimated as the composition of the nominal and error state as illustrated in table 17, such that:

$$\hat{\boldsymbol{p}}_t^+ = \hat{\boldsymbol{p}}_t^- + \delta\hat{\boldsymbol{p}}_t^+ \tag{81a}$$

$$\hat{\boldsymbol{v}}_t^+ = \hat{\boldsymbol{v}}_t^- + \delta\hat{\boldsymbol{v}}_t^+ \tag{81b}$$

$$\hat{\boldsymbol{q}}_t^+ = \hat{\boldsymbol{q}}_t^- \otimes \boldsymbol{q}\left\{\delta\hat{\boldsymbol{\theta}}_{\boldsymbol{t}}^+\right\} \tag{81c}$$

9.1.5. Error reset

Once the measurements have been used to estimate the error state, the error state $\delta\hat{\boldsymbol{x}}_t^+$ and the error covariance matrix $\boldsymbol{P}_t^+$ are reset, in preparation for the state estimate of the next point in time. The reset of the error state and error covariance matrix is as follows:

$$\delta\mathbf{x} \leftarrow g(\delta\mathbf{x}) = \delta\mathbf{x} \ominus \hat{\delta\mathbf{x}} \tag{82}$$

where $g()$ is the error reset function such that:

$$\delta\hat{\boldsymbol{x}}_t^+ = 0 \tag{83}$$

and

$$\boldsymbol{P}_t^+ = \boldsymbol{G}\boldsymbol{P}_t^+\boldsymbol{G}^{\boldsymbol{T}} \tag{84}$$

where G is the Jacobian matrix of the error reset function $g()$and is defined as:

$$\boldsymbol{G} = \begin{bmatrix} \boldsymbol{I} & 0 & 0 \\ 0 & \boldsymbol{I} & 0 \\ 0 & 0 & \boldsymbol{I} - \left[\frac{1}{2}\delta\hat{\boldsymbol{\theta}}_{\boldsymbol{t}}^+\right]_\times \end{bmatrix} \tag{85}$$

9.2. IMU/UWB integration

Implementation of the ESKF requires system initialisation, definition of initial process covariance matrix, calculation of measurement covariance and knowledge of sensor offsets. The initialisation of the system is done using the camera system as described in section 6.1.3. The camera system is used to determined the initial position (x_x^0, x_y^0, x_z^0) and orientation $\begin{bmatrix} q_w^0 & q_x^0 & q_y^0 & q_z^0 \end{bmatrix}$ of the dynamic agent, while the initial velocity $(x_{vx}^0, x_{vy}^0, x_{vz}^0)$ is known to be zero since the dynamic agent starts its motion from static. These variables describe the initial state of the system and correspond to the state variables in table 17. The ESKF is used to integrate measurements from UWB and IMU sensors. However, as shown in figure 48, the UWB tag and IMU are not installed at the same position on the dynamic platform. Therefore, offsets between the two sensors, also referred to as lever arm, need to be calculated.

The offsets were calculated using measurements from the camera system. A reflective marker was placed at the centre of the UWB tag shown in Figure 48, the coordinates of which were given by the camera system. IMU installation is aligned with the vehicle axes, and the vehicle rigid body in the camera system is also aligned with vehicle axes. This allows for the tracking point of camera system to represent the IMU coordinates. Therefore, the IMU's position is assumed to coincide with the camera tracking marker and the changes in orientation and acceleration used are the measurements recorded by the IMU. The dynamic platform was placed in such a way that the IMU coordinate frame and local coordinate frame align and the offsets between the IMU and UWB calculated as the difference between the coordinates recorded by the cameras. As the two coordinate frames were aligned, the offsets calculated are the constant offsets in the IMU frame, which vary in the local frame depending on the orientation of the dynamic platform in the room. The calculated offsets in the IMU frame are shown in table 18.

Table 18: Offset of UWB tag from the Camera tracking marker in the IMU frame

Description	Symbol	value
Offset in x(m)	$of_x^{IMU_{fr}}$	0.148
Offset in y(m)	$of_y^{IMU_{fr}}$	0.035
Offset in z(m)	$of_z^{IMU_{fr}}$	0.007

When considering the prediction and update stages, the predicted state refers to the position of the IMU, while the measurements in the measurement update state correspond to the position of the UWB tag on the dynamic platform. The offsets between the IMU and UWB tag in the IMU frame, shown in table 18, and the quaternion representing the orientation of the dynamic platform in the room, are used to shift between the position of the IMU and the UWB tag in the room frame. This is done by computing the offset from the IMU to the UWB tag in the local frame. The offsets are different for every epoch and vary with the orientation of the dynamic platform. The offsets in the local frame ($of^{Local_{fr}}$) are calculated by (86), using the quaternion q representing the orientation of

the dynamic platform in the room and the quaternion form of the offsets in the IMU frame, $of^{IMU_{fr}} = \begin{bmatrix} 0 & of_x^{IMU_{fr}} & of_y^{IMU_{fr}} & of_z^{IMU_{fr}} \end{bmatrix}$.

$$of^{Local_{fr}} = q \otimes of^{IMU_{fr}} \otimes q^* \tag{86}$$

The position of the UWB tag at time t, X_t^{UWB}, is then calculated using the predicted position of the IMU at time t, X_t, and the offsets in the local frame, $of^{Local_{fr}}$, as shown by (87).

$$X_t^{UWB} = \begin{pmatrix} X_{t,x} \\ X_{t,y} \\ X_{t,z} \end{pmatrix} + \begin{pmatrix} of_x^{Local_{fr}} \\ of_y^{Local_{fr}} \\ of_z^{Local_{fr}} \end{pmatrix} \tag{87}$$

Then considering the measurement update stage of the ESKF, the predicted state of the IMU (X_t) is initially used to calculate the predicted state of the UWB tag (X_t^{UWB}) as shown in (86) and (87). Once the updated state is estimated, the updated quaternion is used in (86) to calculate the updated offsets based on the corrected orientation. The updated offsets are then used to calculate the update state of the IMU using (87).

The remaining of this section discusses the different components used in the integration. In this thesis, two integration methods are used, as shown from the schematic in figure 117. The difference between the two integration is the addition of UWB speed estimations as measurements in the second integration. IMU/UWB without speed estimates, referred to as Integration 1, integrates the pre-processed IMU data with the UWB corrected and filtered range measurements. Alternatively, the IMU/UWB with speed estimates, referred to as Integration 2, integrates the IMU pre-processed data, the UWB corrected and filtered range measurements and filtered speed estimates. The components of range measurement correction, measurement filtering and UWB speed estimation are processes discussed in detail in sections 9.2.1 to 9.2.4. Additionally a more detailed explanation of the two integration implementations is presented in section 9.2.5.

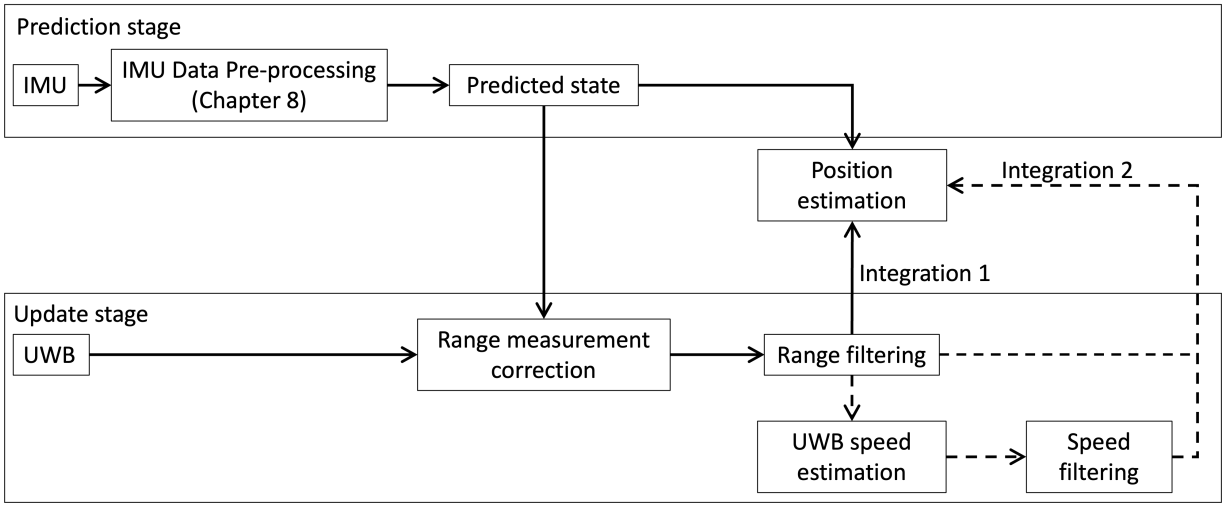


Figure 117: IMU/UWB integration implementations.

9.2.1. Measurement corrections

The measurements used for integration originate from the IMU and UWB. The pre-processing of the IMU measurements is explained in section 8.3. The accelerations from the IMU are initially filtered using a low-pass filter at the cutoff frequency of $15Hz$ and are then smoothed further using the method of moving average with a sliding window of 10 measurements. The angular velocities are also smoothed in the same way, resulting in the same number of acceleration and angular velocity measurements. The pre-processed accelerations and angular velocities are used in the state prediction step of the ESKF.

In the measurement update step of the ESKF, the UWB measurements are corrected before being used in the integration. As seen in section 7.2.2, errors in the UWB measurements depend on the direction of motion of the UWB tag with respect to the anchor. When the UWB tag moves away from the anchor, the range errors tend to be positive, or negative otherwise. The calculated range errors and standard deviations for the two cases, are presented in table 8 in section 7.2.2. These results are used in the integration in order to correct UWB measurements depending on the direction of motion of the dynamic platform with respect to the anchor. The correction of every measurement depends on the anchor from which data are collected and on the direction of motion of the dynamic

platform with respect to the anchor.

In order to determine whether the dynamic platform is moving towards or away from the anchor, the dot product method is used, which determines whether two vectors point in a similar direction. In general, the angle γ between two vectors $\mathbf{A} = \langle a_1, a_2, a_3 \rangle$ and $\mathbf{B} = \langle b_1, b_2, b_3 \rangle$, is related to the dot product of the vectors as:

$$\mathbf{A} \cdot \mathbf{B} = a_1 b_1 + a_2 b_2 + a_3 b_3 \tag{88a}$$

$$\mathbf{A} \cdot \mathbf{B} = |\mathbf{A}||\mathbf{B}| \cos(\gamma) \tag{88b}$$

As cosine has a positive value for angles between $0^o - 90^o$, and negative value for angles $90^o - 180^o$, a positive dot product denotes that vectors $\mathbf{A}$ and $\mathbf{B}$ point in a similar direction, while a negative dot product denotes that the two vectors point in opposite directions. The cosine of angle γ between the two vectors, has a value between $-1 \leq cos(\gamma) \leq 1$. The closer the $cos(\gamma)$ is to 1, the the smaller the γ angle, with 1 denoting that $\mathbf{A}$ and $\mathbf{B}$ point in the same direction. Alternatively the closer $cos(\gamma)$ is to -1, the larger the γ angle, with -1 denoting that $\mathbf{A}$ and $\mathbf{B}$ point in opposite directions.

The cosine of γ, $cos(\gamma)$, is used to determine whether the dynamic platform is moving towards or away from the the anchor using the vector representing the direction of motion in the local frame $\mathbf{U}$ and the vector connecting the dynamic platform to the anchor $\mathbf{D_{an}}$. Vector $\mathbf{D_{an}}$ is calculated as the difference between the anchor position (x_{an}, y_{an}, z_{an}) and last known position of the UWB tag (X_{t-1}^{UWB}) as shown in (89). X_{t-1}^{UWB} is calculated using equations (86) and (87) and the position estimate of the ESKF at the previous time-step (X_{t-1}).

$$D_{an} = X^{an} - X_{t-1}^{UWB} = \begin{bmatrix} x_{an} \\ y_{an} \\ z_{an} \end{bmatrix} - \begin{bmatrix} x_{t-1}^{UWB} \\ y_{t-1}^{UWB} \\ z_{t-1}^{UWB} \end{bmatrix} \tag{89}$$

The direction vector $\mathbf{U}$ represents the direction of motion of the dynamic platform

in the local frame. As mentioned in section 6.1.3, the IMU's x-axis points in the forward direction of motion and hence the direction vector in the IMU frame is $U_{imu} = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix}$. The direction vector U_{imu} is expressed in the form of a quaternion as $v_{imu} = \begin{bmatrix} 0 & 1 & 0 & 0 \end{bmatrix}$. The quaternion v_{imu} is rotated to the local frame using the quaternion estimation of the ESKF in the previous time step q_{t-1} as shown in (90).

$$q_{t-1}^{L} = q_{t-1} \otimes v_{IMU} \otimes q_{t-1}^{*} \tag{90}$$

Equation (90) results in quaternion q_{t-1}^{L}, representing the orientation of the dynamic agent in the local frame. The direction vector $\mathbf{U}$, is then defined as:

$$U = \begin{pmatrix} q_{t-1,x}^{L} \\ q_{t-1,y}^{L} \\ q_{t-1,z}^{L} \end{pmatrix} = \begin{pmatrix} u_x \\ u_y \\ u_z \end{pmatrix} \tag{91}$$

Once the direction vector $\mathbf{U}$ and vector connecting the dynamic platform to the anchor D_{an} are defined, $cos(\gamma)$ is calculated using (88b), and used to determine whether the dynamic platform is moving towards or away from the anchor, allowing for a UWB measurement correction to be applied.

For the correction of the UWB measurements, three main categories of measurements are considered, which are defined by the direction of motion. The motion of the dynamic platform can be either towards the anchor, away from the anchor or neutral motion with respect to the anchor.

Although in section 7.2.2, there appears to be a dependency of the measurement errors on the direction of motion, there are some areas which can be considered as neutral. This can be seen in figure 74, where although the majority of the measurements are colour-coded as either blue or red, corresponding to negative and positive measurement errors, there are some measurements colour-coded as white. The white colour occurs at the transition zones between negative and positive measurement errors, indicating areas with errors close to zero. Moreover, observing the error distributions in figures 75 to 82, it can be seen that the distributions of measurement errors for the two directions of motion (moving towards

and away from the anchor) are characterised by distinct peaks, however they also have a large overlap. This suggests that for the measurements of which the errors fall within the overlap, applying the correction based on the direction of motion can result in increasing the error in the measurement rather than reducing it. As a result the neutral category in terms of the direction of motion is introduced.

The distinction between the three categories is related to the value of $cos(\gamma)$. When very close to either 1 or -1, it suggests that the dynamic platform is moving almost directly towards or away from the anchor. At $cos(\gamma)$ value larger than 0.95, the γ angle is between $0^o - 18.2^o$, alternatively at a value smaller than -0.95, γ is between $161.8^o - 180^o$. Measurements with a value of $cos(\gamma)$ larger than 0.95 are classified as those collected when the dynamic platform is moving towards the anchor. Alternatively measurements with a $cos(\gamma)$ value less than -0.95 are classified as those collected when the dynamic platform is moving away from the anchor. The measurements with a $cos(\gamma)$ value $-0.95 \leq cos(\gamma) \leq 0.95$ are classified as neutral. The use of 0.95 as a threshold was determined by applying different thresholds and tuning the value using the empirical data in the integration.

The classification amongst the three groups, allows for the selection of the appropriate correction to be applied. For measurements classified as collected while moving either towards or away form the anchor, the mean (μ) of the measurement errors calculated in table 8, are used respectively as corrections. Alternatively, when classified as neutral with respect to the direction of motion, the correction applied is set to zero, as it cannot be concluded whether it belongs to any of the other groups. The raw range measurement ($range_{raw}$) is used to calculate the corrected range measurement ($range_{corr}$) as:

$$range_{corr} = \begin{cases} range_{raw} - \mu_{\text{away}}\,, & \cos(\gamma) < -0.95 \\ range_{raw} - \mu_{twrds}, & \cos(\gamma) > 0.95 \\ range_{raw} - 0, & -0.95 \leq \cos(\gamma) \leq 0.95 \end{cases} \tag{92}$$

The corrected UWB measurements and pre-processed accelerations and angular velocities are the measurements used in the integration. The IMU accelerations and angular rates are used at the prediction stage, while UWB measurements are used in the measurement

update stage. An additional set of measurements, used in the measurement update stage, is proposed in this thesis to reduce the impact of erroneous accelerations. The additional measurements are UWB speed estimations and a detailed explanation of their calculation is given in the following section.

9.2.2. UWB speed estimation

IMU errors are corrected through this integration using UWB measurements, which are range measurements between the UWB anchors and dynamic tag. The UWB range measurements are related to the position of the dynamic tag using equation (78), and are commonly used to correct the state elements related to the position (x_x, x_y, x_z). However, as demonstrated in section 8.5, the major source of error from an IMU system is related to the IMU measurements of acceleration, suggesting a significant error in velocity prediction in addition to position. An alternative approach is proposed in this section, where the UWB measurements are used to correct the velocities in addition to position predictions of the nominal stage.

The method presented in this section is used to estimate the speed of the dynamic tag, which is then related to velocities in the three orthogonal directions using equation (93). The UWB speed estimate s_{uwb}, is used as an additional measurement input in measurement update stage of the ESKF to correct the velocity predictions $(x_{v_x}, x_{v_y}, x_{v_z})$.

$$s_{uwb} = \sqrt{x_{v_x}^2 + x_{v_y}^2 + x_{v_z}^2} \tag{93}$$

The speed is estimated using a measurement of the distance travelled between two epoches (d_s), over the corresponding time elapsed (dt). The time elapsed corresponds to the change in time between consecutive IMU measurements, as state estimation occurs at the same rate as IMU measurements. The estimation of distance travelled (d_s) is calculated using UWB range measurements in integration with position and orientation estimates of the state at the previous time-step (X_{t-1}, q_{t-1}). Two different methods are used to calculate the distance travelled and consequently speed. The first method (method 1) calculates d_s assuming there are no uncertainties in the range measurement, while the second method

(method 2) calculates d_s, considering the uncertainty of the UWB measurements. The two different methods are described below.

In method 1 the solution for d_s is calculated by defining the relationship between the UWB range measurement d_{meas} and the distance travelled in the direction of motion d_s, shown in figure 118. The steps in defining the relationship between d_{mean} and d_s are discussed below.

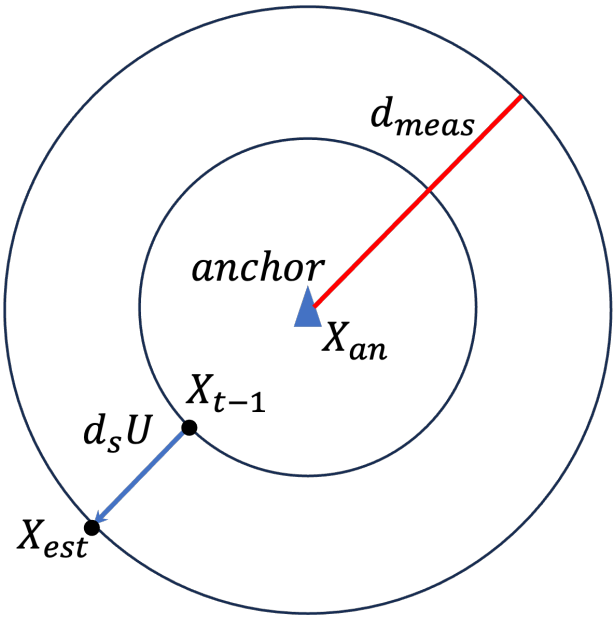


Figure 118: Displacement estimation using UWB measurements.

A direction vector **U** representing the direction of motion in the local frame, is calculated as explained earlier in equations (90) and (91). The estimate of the current position is related to the previous position estimate of the ESKF by:

$$X_{est} = X_{t-1} + d_s U \tag{94}$$

$$\implies \begin{pmatrix} x_{est} \\ y_{est} \\ z_{est} \end{pmatrix} = \begin{pmatrix} x_{t-1} \\ y_{t-1} \\ z_{t-1} \end{pmatrix} + d_s \begin{pmatrix} u_x \\ u_y \\ u_z \end{pmatrix} \tag{95}$$

The distance between the position estimate and the anchor is defined as:

$$d_{meas}^2 = (x_{an} - x_{est})^2 + (y_{an} - y_{est})^2 + (z_{an} - z_{est})^2 \tag{96}$$

Replacing the position estimate in (96) with relationship (95):

$$d_{meas}^2 = (x_{an} - x_{t-1} - d_s u_x)^2 + (y_{an} - y_{t-1} - d_s u_y)^2 + (z_{an} - z_{t-1} - d_s u_z)^2 \tag{97}$$

Let:

$$x_k = x_{an} - x_{t-1}, \quad y_k = y_{an} - y_{t-1}, \quad z_k = z_{an} - z_{t-1} \tag{98}$$

$$d_s^2 \left(u_x^2 + u_y^2 + u_z^2\right) - 2d_s \left(x_k u_x + y_k u_y + z_k u_z\right) + \left(x_k^2 + y_k^2 + z_k^2 - d_{meas}^2\right) = 0 \tag{99}$$

Equation (99), shows a quadratic relationship between the distance measurement d_{meas} and distance travelled in the direction of motion d_s. This quadratic equation results in two solutions for d_s. There are three possible combinations of solutions, the first is a positive and a negative solution, the second combination is two positive solutions and the third is two negative solutions. Selection of the right solution is made based on the assumption that d_s should have a positive value, as the direction of motion has already been specified through the direction vector U. In the case of a positive and negative solution, the positive solution is selected. In the case of two positive solutions, the smallest of the two solutions is used. This is because the change in time (dt) between consecutive epoches is very small ($0.02s$) and the dynamic platform is not expected to travel a long distance during that time. Finally, in the case of two negative solutions, none of the solutions is used, since d_s should always be positive. In this scenario it is assumed that for the specific epoch there is no solution for d_s and consequently no speed measurements from UWB. Having calculated the distance travelled d_s, the corresponding UWB speed measurement is defined as $s_{uwb} = d_s/d_t$.

The approach followed in method 1 assumes that UWB range measurements have no associated errors. However, it is common for measurement errors to be present. The alternative method, method 2, is hence developed, where the uncertainty in range measurements

is taken into consideration when estimating the distance travelled. The uncertainty of the measurements is related to the standard deviation (σ) of the range errors observed from a specific anchor, as illustrated in figure 119. The grey area in figure 119, corresponds to the area within which the true range measurement lies, referred to as the uncertainty area. The method implemented for estimating the distance travelled when considering measurement errors is very similar to the first method presented between equations (90) and (99). The difference lies in the calculation of d_s, where this method uses an iterative approach in estimating the current position X_{est}, by iteratively moving closer to the uncertainty area in small steps.

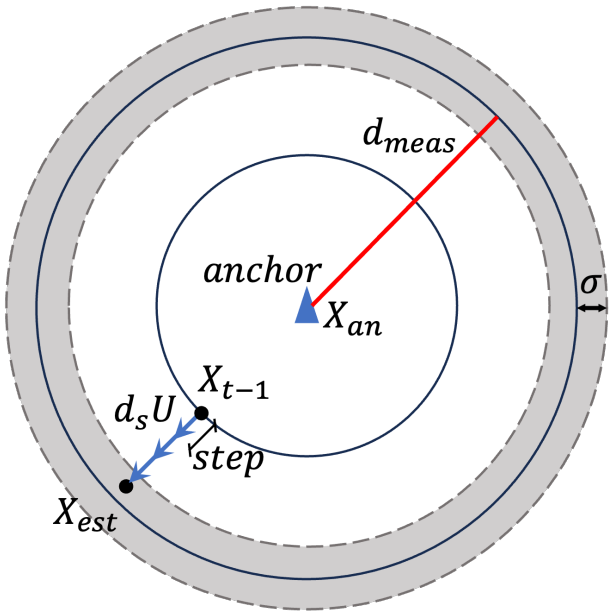


Figure 119: Displacement estimation using UWB measurements and the standard deviation of the measurement errors.

As an initial step, the direction vector of motion (U), is defined using equations (90) and (91). The current position is again estimated using equation (95). However, in this method d_s is defined as:

$$d_s^i = step * i \tag{100}$$

where, *step* is the step size and i is the iteration number.

Since this is an iterative method, the value of d_s increases with every iteration. For every iteration a new X_{est} referred to as X^i_{est} is calculate using (95). In every iteration X^i_{est} is used to calculate a provisional distance measurement (d^i_{est}) as:

$$d^i_{est}{}^2 = \left(x_{an} - x^i_{est}\right)^2 + \left(y_{an} - y^i_{est}\right)^2 + \left(z_{an} - z^i_{est}\right)^2 \tag{101}$$

An error term (ε^i_{est})is then introduced, which represents the difference between the actual range measurement (d_{meas}) and (d^i_{est}) as:

$$\varepsilon^i_{est} = |d^i_{est} - d_{meas}| \tag{102}$$

The iterative process stops once:

$$\varepsilon^i_{est} \leq \sigma \tag{103}$$

The condition in equation (103), ensures that the estimated distance falls within one standard deviation of the measurement and hence the estimated current position (X^i_{est}) is within the uncertainty area shown in figure 119. The d^i_s that satisfies condition (103), is used as the estimated distance travelled provided by the UWB system. Finally, the estimated speed is calculated as $s_{uwb} = d_s/d_t$. This method, in contrast to the first, considers the uncertainty in the range measurements when calculating the distance travelled d_s.

The estimated speed using the UWB range measurements is used as an additional measurement in the ESKF. This results in additional elements in the $\mathbf{H_x}$ matrix, since for every range measurement there are two observation functions $h()$. One observation function relating the range measurements to the state variables of position, as expressed by (78), and another one relating the UWB speed estimates to the state variables of velocity, as expressed by (93). As a result, for every UWB measurement the $\mathbf{H_x}$ matrix is:

$$\mathbf{H_x} = \begin{bmatrix} \frac{\partial d_{meas}}{\partial x_x} & \frac{\partial d_{meas}}{\partial x_y} & \frac{\partial d_{meas}}{\partial x_z} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{\partial s_{uwb}}{\partial x_{v_x}} & \frac{\partial s_{uwb}}{\partial x_{v_y}} & \frac{\partial s_{uwb}}{\partial x_{v_z}} & 0 & 0 & 0 \end{bmatrix} \tag{104}$$

where $\frac{\partial d_{meas}}{\partial x_x}$, $\frac{\partial d_{meas}}{\partial x_y}$ and $\frac{\partial d_{meas}}{\partial x_z}$ are defined as in equations (80) and $\frac{\partial s_{uwb}}{\partial x_{v_x}}$, $\frac{\partial s_{uwb}}{\partial x_{v_y}}$ and $\frac{\partial s_{uwb}}{\partial x_{v_z}}$ are defined in equations (105).

$$\frac{\partial s_{uwb}}{\partial x_{v_x}} = \frac{x_{v_x}}{\sqrt{x_{v_x}^2 + x_{v_y}^2 + x_{v_z}^2}} \tag{105a}$$

$$\frac{\partial s_{uwb}}{\partial x_{v_y}} = \frac{x_{v_y}}{\sqrt{x_{v_x}^2 + x_{v_y}^2 + x_{v_z}^2}} \tag{105b}$$

$$\frac{\partial s_{uwb}}{\partial x_{v_z}} = \frac{x_{v_z}}{\sqrt{x_{v_x}^2 + x_{v_y}^2 + x_{v_z}^2}} \tag{105c}$$

The performance of the two UWB speed estimation methods are evaluated and compared in section 9.3.1. The UWB speed estimates, are used as an additional source of measurements in the proposed integration. The UWB range measurements and speed estimations are used in the measurement update stage of the ESKF. The next section proposes a method for filtering the measurements used in the update stage of the integration. This is done to mitigate the use of significant error measurements to improve performance.

9.2.3. Measurement filtering

During the update state of the ESKF, the measurements are filtered to exclude erroneous measurements, by comparing the absolute of the innovation, $\left|\boldsymbol{y}_k - \hat{\boldsymbol{x}}_k^-\right|$ in (72), to a predefined threshold (Yao et al. 2017). Note, however, that the definition of the threshold is different from Yao et al. (2017), in which the value of the threshold is between 3 and 6 standard deviations (σ) of measurement errors. Alternatively, the threshold in this thesis is defined using the errors from both the IMU and UWB.

The idea behind the expression used to decide on the threshold, stems from the fact that the innovation is calculated using two components, 1) expected measurement based on the prediction (on the nominal state) and 2) actual measurements. In both of these components, there are errors that cause a deviation between the estimate and truth. The deviation of the predicted state from the truth is related to errors in the IMU solution, while the measurement deviation from the truth is related to the UWB measurement errors.

As a result the threshold imposed on the innovation should be related to both IMU and UWB errors.

The ESKF is a looping procedure where the agent loops between prediction and measurement update stages. During epoches where no measurements are available the filter uses the predicted state as the state estimate for the particular epoch. As a result, in between UWB measurements, the state is updated using only the prediction stage which relies on IMU measurements. During these periods the solution is assumed to drift as described by IMU drift. Once a UWB measurement is available for a measurement update, the maximum value of the innovation is a composition between the IMU drift and the maximum UWB range errors as shown in figure 120. The threshold is then defined as:

$$\sigma^d_{\text{threshold}} = d^{UWB}_{\text{error}} + d^{IMU}_{\text{drift}} \tag{106}$$

where d^{UWB}_{error} is the maximum UWB range error and d^{IMU}_{drift} is the error due to IMU drift. d^{UWB}_{error} is calculated as 3 standard deviations of UWB range measurement errors of anchor i (σ_i), as shown in (107). This assumes that the UWB range errors are normally distributed about zero and hence 3σ encloses 99.7% of the measurement errors.

Alternatively, the errors associated with the IMU drift is calculated by multiplying the IMU drift rate by the time duration for which the IMU prediction has not been corrected by a UWB measurement. This time duration is calculated using the current time t^{UWB}_j and the last time UWB measurements were used in the integration t^{UWB}_{j-1} as shown by (108).

$$d^{UWB}_{\text{error}} = 3\sigma^d_i \tag{107}$$

$$d^{IMU}_{\text{drift}} = \text{ Position drift rate } \times \left(t^{UWB}_j - t^{UWB}_{j-1}\right) \tag{108}$$

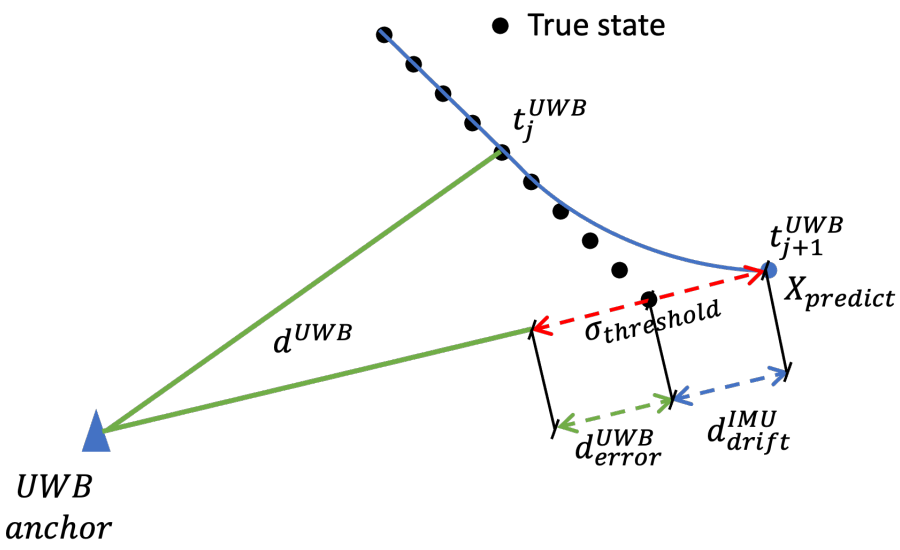


Figure 120: UWB measurement filtering.

The same principle is used to filter the UWB speed estimates. Once the UWB range measurements are filtered, the UWB speed is estimated as described in section 9.2.2. The UWB speed estimate is treated as an additional measurement, and filtering is applied to exclude UWB speed measurements with significant errors. The same principles of defining the threshold of the range measurements apply to the definition of the threshold for speed measurements. The IMU speed drift is used together with the UWB speed errors to calculate the threshold for the speed measurements as shown in equations 109 to 111

$$\sigma^s_{\text{threshold}} = s^{UWB}_{\text{error}} + s^{IMU}_{\text{drift}} \tag{109}$$

$$s^{UWB}_{\text{error}} = 3\sigma^s_i \tag{110}$$

$$s^{IMU}_{\text{drift}} = \text{ Speed drift rate } \times \left(t_j^{UWB} - t_{j-1}^{UWB}\right) \tag{111}$$

An alternative method of defining the thresholds, would be to assume that UWB range measurements, UWB speed estimates, IMU position drift and IMU speed drift are normally distributed with $d^{UWB}_{error}, s^{UWB}_{error}, d^{IMU}_{drift}, s^{IMU}_{drift}$ corresponding to their standard deviations (σ).

Under this assumptions, thresholds $\sigma^d_{threshold}, \sigma^s_{threshold}$ in equations (106) and (109) would have been calculated as $\sigma^d_{threshold} = \sqrt{d^{UWB^2}_{error} + d^{IMU^2}_{drift}}$ and $\sigma^s_{threshold} = \sqrt{s^{UWB^2}_{error} + s^{IMU^2}_{drift}}$. However, this would have resulted in a much smaller threshold than the one proposed in this section, which would have been more restrictive in the inclusion of UWB measurements. As a result, the thresholds used in this thesis are the ones presented in equations (106) to (108) and (109) to (111).

The filtering method proposed in this section, results in the UWB measurements used in the integration. The final component of implementing the ESKF is the calculation of the variances corresponding to the measurements used during the integration. The variance of the measurements are used in two main ways during the integration. The calculated measurement variance values, and the ways in which they are used in the integration are explained in the following section.

9.2.4. Covariance matrices/ variance covariance estimation

The variance of the measurements are used in the ESKF to determine the error covariance matrices which are used to weight the measurements in the integration. The measurements used in the integration are the IMU accelerations and angular velocities in the prediction stage of the ESKF and UWB-speed estimates and range measurements in the measurement update stage. The covariance matrices of the ESKF, P^- and V, presented in section 9.1, represent the error covariance of the prediction and update stages and are related to the IMU and UWB measurement variances respectively. These error covariance matrices are used to calculate the Kalaman gain K, as shown in (71), which as explained earlier, is a weighting between the two systems, representing the level of trust in each systems. This section explains how the variances of the measurements from the two systems are calculated.

The variance of the IMU measurements is calculated using measurements collected during the static parts of all the trajectories. The calculated variance values of the accelerations and angular velocities in the three orthogonal directions are shown in table 19. These are the variance values used in equation (64) to calculate the covariance matrix of

perturbations (Q_i) expressed in (69). Q_i is used to calculate the priori error covariance matrix (P^-) as shown in (66), which represents the errors in the prediction stage.

Table 19: Variance of accelerations and angular velocities ω, calculated using the static phase of the trajectories

IMU variance	
$\sigma^2_{a_x}$	0.023
$\sigma^2_{a_y}$	0.036
$\sigma^2_{a_z}$	$9.261\text{E}-03$
$\sigma^2_{\omega_x}$	$8.774\text{E}-04$
$\sigma^2_{\omega_y}$	$6.477\text{E}-05$
$\sigma^2_{\omega_z}$	$3.666\text{E}-04$

The covariance matrix V is a representation of the errors in the measurements used in the measurement update stage of the integration. The method proposed in this thesis uses the corrected range measurements of the UWB, and calculates speed estimations using the corrected UWB measurements. To ensure that every measurement has a corresponding variance, variance values of UWB speed estimates and UWB range measurements need to be established.

The variance value of the UWB speed estimates, is calculated for each anchor, based on speed estimates from UWB static data. The speed estimated using the static data s^{static}_{uwb}, is calculated using the distance travelled (d^{static}_s) over a specific time interval dt. As static UWB measurements are used and the dynamic platform does not move in a particular direction, there is no use of the direction vector U in s^{static}_{uwb} calculation. Instead, the distance travelled (d^{static}_s) is calculated as the difference between two consecutive range measurements and dt is the time elapsed between the two measurements. The calculation of the speed estimates during the static phase is shown in (112).

$$d_s^{static} = d_{meas_t} - d_{meas_{t-1}} \tag{112a}$$

$$s_{uwb}^{static} = d_s^{static}/dt \tag{112b}$$

A representative example of the distribution of s_{uwb}^{static} is shown in figure 121 for anchor 0x1102. As shown from figure 121, s_{uwb}^{static} are normally distributed around zero and their variance values are used as the variance of the UWB speed estimates in the ESKF. The variance values for the speed estimates for the eight anchors are presented in table 20.

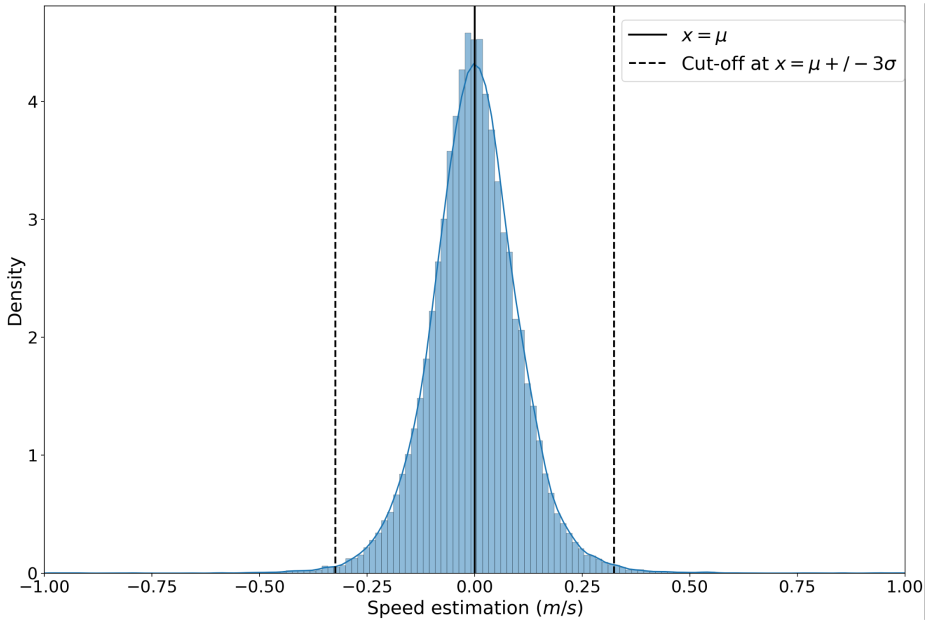


Figure 121: UWB speed estimates using static data from anchor 0x1102.

Table 20: The variance of the UWB speed estimations.

Anchor	$\sigma^2_{S_{UWB}}$
0x115b	0.009
0x113c	0.010
0x110d	0.096
0x1163	0.136
0x117f	0.034
0x1102	0.012
0x1171	0.097
0x110b	0.054

When considering the variances of the UWB range measurements, there are two ways in which they are used. The first is for the calculation of the error covariance matrix V, which is further used to calculate the Kalman gain K. Alternatively, standard deviations of the measurements are also used in the UWB speed estimation process (section 9.2.2) as well as in defining the threshold for measurement filtering (section 9.2.3) prior to the integration. These two approaches have some fundamental differences.

When considering the error covariance matrix V, it is essential to remember that V is used in the calculation of K, where K represents the weighting between the prediction and update stages of the filter. In order to have a fair weighting between the prediction and update stages, the variances used for the calculation of V should be comparable to those used to calculate P^-. Since P^- is calculated using static IMU measurements, V is also calculated using static UWB range measurements.

In contrast, when the standard deviation σ of the UWB measurements is used in UWB speed estimation and measurement filtering, it is not used for comparison with the IMU measurements. In both UWB speed estimation and measurement filtering, σ is considered to represent the error in the specific measurement. For this reason, the σ derived from the dynamic data is used in both of these cases. As discussed in section 9.2.1, the dynamic

data are classified according to the direction of motion with respect to the anchor. When calculating the UWB speed and measurement filtering threshold of a measurement, the σ corresponding to the direction of motion is used. The usage of the different variance and standard deviation values is schematically shown in figure 122, and their corresponding values are presented in table 21.

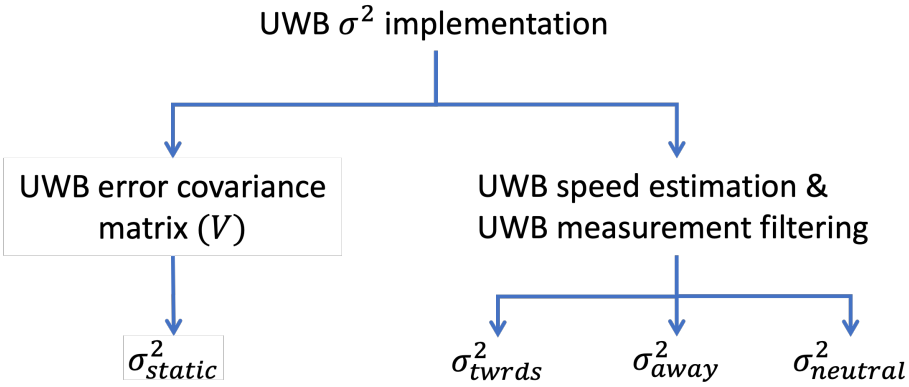


Figure 122: The usage of the different UWB σ values in the integration.

Table 21: The dynamic and static variances of UWB measurements used in the integration.

Anchor	Dynamic variance			Static variances
	σ^2_{twrds}	σ^2_{away}	$\sigma^2_{neutral}$	σ^2_{static}
0x115b	0.079	0.089	0.017	0.007
0x113c	0.097	0.086	0.023	0.014
0x110d	0.099	0.089	0.025	0.013
0x1163	0.090	0.094	0.033	0.023
0x117f	0.074	0.092	0.031	0.021
0x1102	0.090	0.132	0.045	0.032
0x1171	0.089	0.116	0.036	0.024
0x110b	0.103	0.113	0.035	0.022

This section presented the calculated variance and standard deviation values, and the way they are used in the different components of the integration. Moreover, section 9.2.1

proposed the correction method applied to the UWB range measurements, section 9.2.2 proposed the UWB speed estimation method for the creation of an additional measurement in the measurement update stage and section 9.2.3 proposed a measurement filtering method for the elimination of erroneous measurements in the integration. The next section explains the implementation of the integration, using the proposed methods and values presented in sections 9.2.1 to 9.2.4.

9.2.5. Implementation

The corrected IMU and UWB measurements, the calculated variance and standard deviation values and the UWB speed estimation and measurement filtering methods, are used to develop and compare the performance of two methods of integration. The two integration methods are referred to as Integration 1 and Integration 2 (discussed earlier), and the difference between the two methods is the use of UWB speed estimates. Both methods of integration, use the ESKF presented in section 9.1 to calculate the positioning solution. The prediction stage of the ESKF happens at the IMU data collection rate $50Hz$, while measurement update occurs if UWB measurements are available within an IMU epoch. This results in a position solution estimated at the rate of IMU data collection. The two integration methods are explained below in detail.

A schematic of Integration 1 is shown in figure 123, where the IMU data and UWB range measurements are used in the ESKF, as described in section 9.1. In addition, UWB measurement correction and measurement filtering methods explained in sections 9.2.1 and 9.2.3 respectively, are also implemented. The pre-processed IMU measurements (discussed in section 8.3) are used in the prediction stage of the filter to calculate the nominal and error state, resulting in the predicted state (X_t^-) and the priori error covariance matrix (P_t^-). When UWB measurements are available, the measurement update stage of the filter is implemented. At the measurement update stage, the UWB range measurements are corrected as explained in section 9.2.1, according to the direction of motion of the dynamic platform with respect to the anchor from which the measurement is collected. The corrected UWB measurements are then filtered as described in section 9.2.3, with

the measurements passing through the filter (y_t) being used to calculate the observation matrix H and measurement error covariance V. H, P_t^- and V are used to calculate the Kalman gain K_t, which is then used together with the innovation ($y_t - X_t^-$) to calculate the correction $\delta\hat{x}_t^+$ to the predicted state X_t^-. The predicted state is then corrected resulting in the final position estimate (X_t, P_t) for the specific epoch which is used in the next iteration as the previous epoch's position estimation.

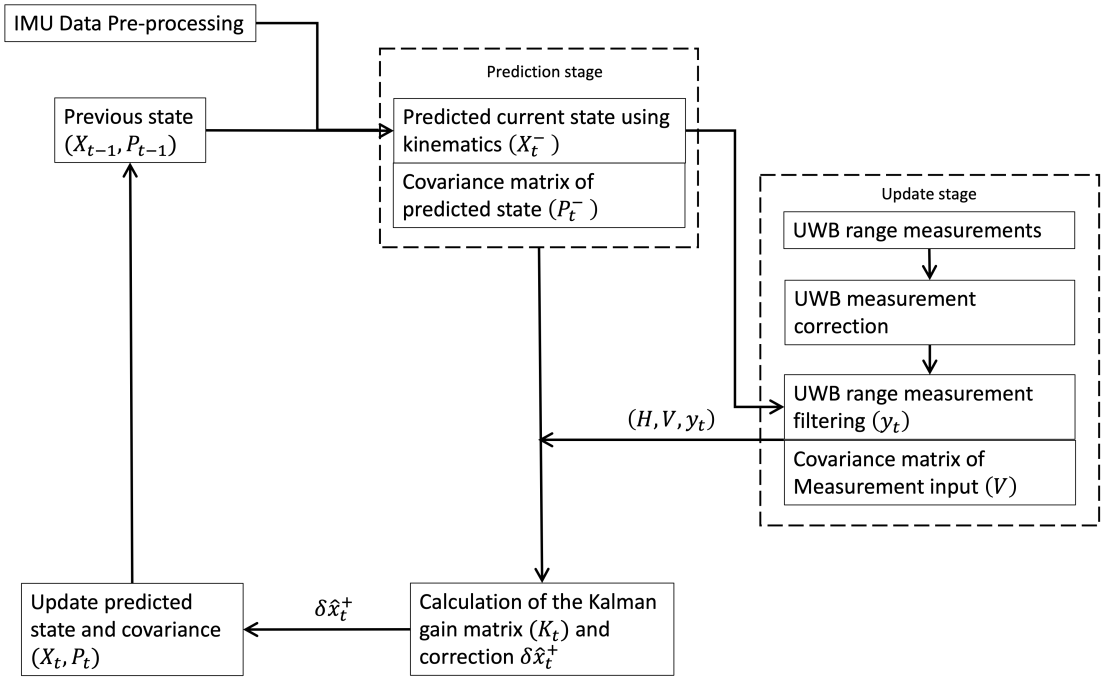


Figure 123: Integration 1.

Integration 2 presented in figure 124, is Integration 1 with the addition of the UWB-speed measurements. The UWB speeds are estimated using the filtered range measurements (y_{range}). Once the speeds are estimated, the UWB speed filtering method, presented in section 9.2.3, is applied to filter the speed estimates before using them in the integration.

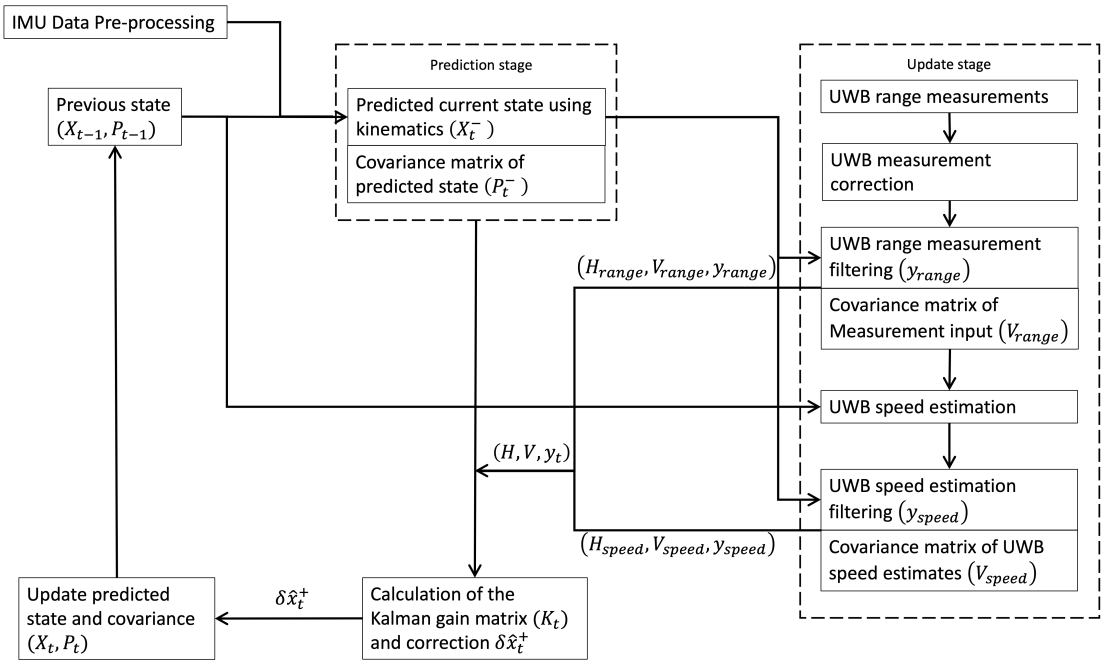


Figure 124: Integration 2.

The two implementations presented in this section are used to calculate the positioning solution when the IMU and UWB systems are integrated. The following section discusses the accuracy performance of the two implementations and compares them.

9.3. IMU/UWB integration results

Initially, the performance of the Integration 2 is evaluated by examining the methods and adjusting the parameters used for UWB speed estimation. Once the suitable method of UWB speed estimation and relevant parameters are identified, the performance of the Integration 1 and 2 are compared. To demonstrate this process, path15 was used as a test case when developing the integration methods.

In this section, performance of the two UWB speed estimation methods presented in section 9.2.2, are evaluated. The performance of the UWB speed estimation using measurement uncertainty, is then evaluated by varying the level of σ defining the uncertainty area around the range measurement. Finally, when the suitable UWB speed estimation method and corresponding level of σ is identified, the performance of Integration 1 and 2

are compared.

9.3.1. Performance of UWB speed estimation methods

As discussed in section 9.2.5 Integration 2 uses the UWB speed estimates of the range measurements. In section 9.2.2, two ways of estimating speed based on UWB data are presented. One method solves for UWB speed assuming the range measurements are accurate (method 1). Alternatively, method 2, incorporates the uncertainty in the measurements by creating an uncertainty area around the measurement defined by the σ of the measurement. The performance of the two methods is compared by calculating the errors in the estimated speeds as well as the resulting errors in the 2D positioning solution of Integration 2. The errors in speed estimates are calculated using the true speed of motion, which is considered to be the average speed of every trajectory presented in table 6.

As explained in section 9.2.5, not every range measurement and UWB-speed estimate is used in the integration as both types of measurements are filtered. To evaluate the performance of the UWB speed estimation methods, the estimated speeds used in the integration are analysed. The speed error distributions of path15 are shown in figure 125. Figure 125 shows that the distributions of the two methods are not comparable which prompts a comparison of the number of speed estimates used in the integration, by each method. The number of speeds estimates used in the integration, for some of the dynamic paths are shown in table 22, as a percentage of total number of UWB range measurements available.

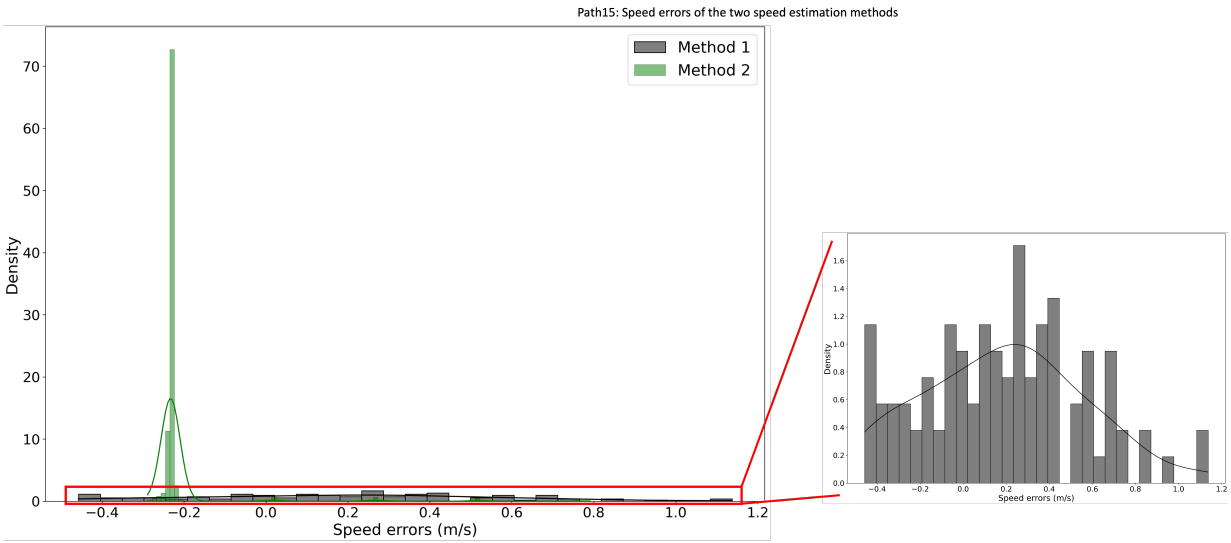


Figure 125: Estimated speed errors using the two methods of UWB speed estimation for path15. Methods 1 and 2 correspond to UWB speed estimation without and with the consideration of measurements uncertainty, respectively.

Table 22: Number of speed estimates used during the integration depending on the speed estimation method used. Methods 1 and 2 correspond to UWB speed estimation without and with the consideration of measurements uncertainty, respectively.

Path	Range Measurements used	Method 1		Method 2	
		Number speed estimates used	% speed estimates used	Number speed estimates used	% speed estimates used
path15	3286	99	3.0%	1810	55.1%
path12	3361	82	2.4%	1220	36.3%
path6	3279	75	2.3%	1943	59.3%
diag1	595	13	2.2%	259	43.5%

As can be seen from table 22, the number of UWB speed estimates used in the integration is only between $2\% - 3\%$ of the used range measurements, when method 1 is used.

Alternatively, when the method 2 is implemented, the number of speed estimates used in the integration rises to $36\% - 55\%$ of the used range measurements. Method 2 allows for more speed estimates to be used compared to method 1. This indicates that method 2 calculates more accurate speed estimates, since more measurements pass through measurement filtering of UWB speeds. This demonstrates that by incorporating the uncertainty of the range measurement in the speed estimation it is more likely to result in a more representative speed estimation.

The larger number of UWB-speed estimates used, the more frequently IMU velocities are corrected, which is expected to result in a more accurate positioning solution. The 2D position error distributions corresponding to the two speed estimation methods used in the integration are shown in figure 126 for path15. Method 2 performs better than method 1, as the position error distribution of method 2 is shifted towards smaller error values. This is also evident from the calculated RMSE and accuracy (95%) shown for some of the trajectories in table 23. As expected, method 2 performs better (up to 61%) than method 1, in terms of 2D position accuracy (95%).

Table 23: Accuracy (95%) and RMSE of the 2D positioning solution using the methods 1 and 2 of UWB speed estimation.

Path	Accuracy (95%)(m)			RMSE (m)		
	Method 1	Method 2	% difference	Method 1	Method 2	% difference
path15	0.319	0.198	61%	0.201	0.115	75%
path12	0.524	0.405	30%	0.318	0.210	51%
path6	0.329	0.316	4%	0.193	0.173	12%
diag1	0.231	0.200	15%	0.131	0.115	14%

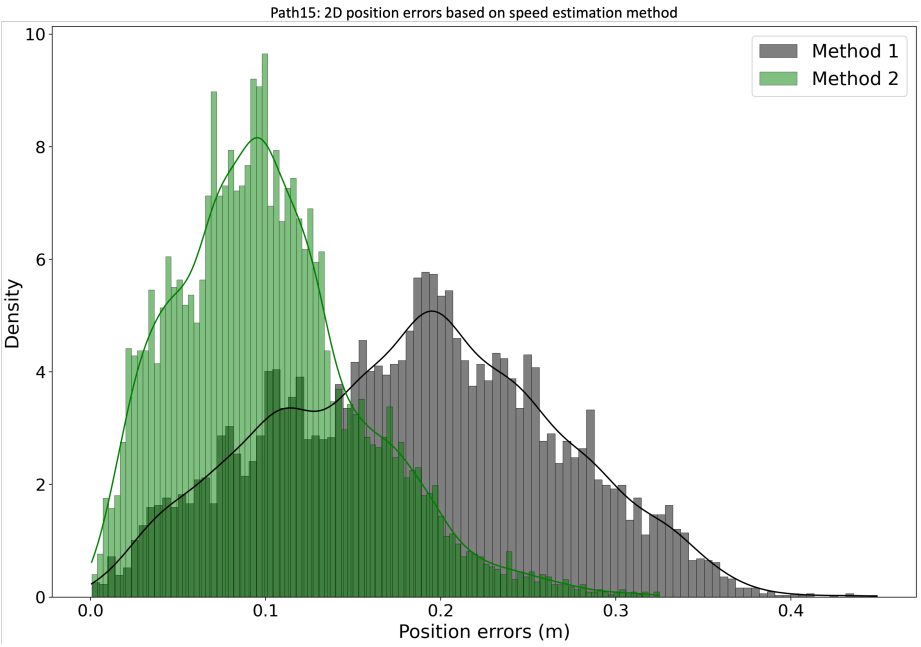


Figure 126: 2D position errors when using UWB speed estimation methods 1 and 2 for path15.

As range measurements are not error-free, method 2 performs better, as it considers the stochasticity of the range measurements, resulting in a more representative speed estimations. The uncertainty area of this method was presented in section 9.2.2, which can be defined by the σ of the range measurements. The next section investigates the sensitivity of the positioning solution on the level of σ used to represent the uncertainty area around the range measurement.

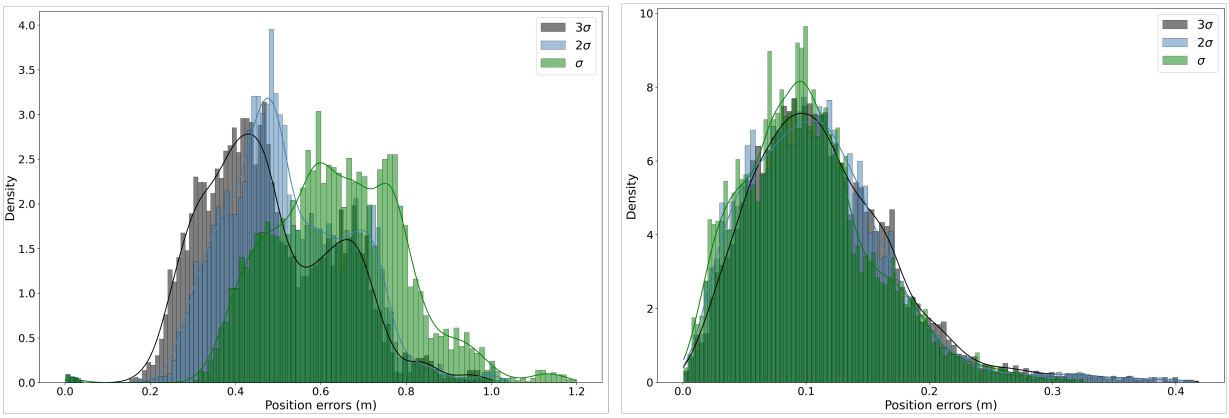
9.3.2. Sensitivity of speed estimation method on the level of standard deviation used

UWB speed estimation method 2 presented in section 9.2.2, requires the use of a value representing the error in the range measurements. This value is used to create the uncertainty area around the range measurements, defining the area within which the true range measurement lies, as shown by (103) and figure 119. In the proposed method, the uncertainty area is defined using 1σ of the range measurements.

The reason behind the choice of 1σ is that a step-wise approach is implemented to

estimate the distance travelled. As shown in the figure 119, the distance travelled is calculated by taking steps in the direction of motion. The number of steps is an iterative process that stops, once the estimated position falls within the bound of the range measurement. Therefore, the bound should be defined with a value that is representative of the errors. Assuming a normal distribution of the measurement errors, a value of 1σ encloses about 68% of the measurement errors. Alternatively, 95% and 99.7% of the measurement errors are assumed to be enclosed within 2σ and 3σ respectively. Increasing the level of σ allows for a larger proportion of measurement errors to be considered. However, by increasing the level of σ we are moving closer to the tails of the measurement error distributions, which means we are considering errors that have lower probability of occurring. As the iterative process stops once the solution falls within the bound, using high level of σ increases the likelihood of stopping at position corresponding to an error of high significance towards the tails of the distribution. For this reason, the value of 1σ was proposed in section 9.2.2.

Nonetheless, although 1σ is proposed, the sensitivity of the positioning solution to the level of σ used is evaluated. The positioning solution is calculated for a few paths, with the bound being defined by 1σ, 2σ and 3σ. The performance is evaluated by visually assessing the 2D position error distributions while also comparing the 2D positioning accuracy (95%) achieved with the different σ levels. A visual observation of the position error distributions of the different levels of σ used, shows that there are generally two groups of cases. In one group the distinction between the error distributions of 1σ, 2σ and 3σ is very clear. Alternatively, there are cases where the distinction between the three cases is not visually clear. Examples of the two cases are shown in figure 127. In figure 127a, the distinction in the distribution of position errors using 1σ, 2σ and 3σ is clear with 3σ resulting in the best accuracy performance. Alternatively, although the distinction in performance in figure 127b is not very clear, it can be seen that 1σ is characterised by a narrower distribution of errors, resulting in better accuracy (95%). Table 24 presents the positioning accuracy (95%) and RMSE obtained using 1σ, 2σ and 3σ during speed estimation and highlights the level of σ that results in better accuracy.



(a) path14 (b) path15

Figure 127: 2D Position errors with varying level of σ.

As shown in table 24, for some paths (e.g. path14), the use of 3σ results in better positioning accuracy compared to 1σ and 2σ. Alternatively, for daig1 and diag8, although 3σ performs better than 1σ, the improvement in terms of accuracy is only $0.007m$ and $0.013m$ respectively.

A possible reason 3σ performs better than 1σ is the speed of motion of the dynamic platform, due to the relationship between speed and measurement errors discussed in section 7.2.2. To investigate whether the level σ relates to the speed of motion, table 24, presents the paths in ascending order with respect to the speed of motion. As can be seen from table 24, the better accuracy performance of 3σ is seen in both lower as well as higher speeds. Consequently, the results in the table do not show any relationship between the suitable level of σ and speed of motion.

Other than the level of σ used, an additional parameter that can influence speed estimation, is the step-size. The step-size was determined using path15 as this was the test case used for the development of the method. A number of step-sizes have been examined to identify the one that results in the best position accuracy. This resulted in a step-size value of $0.005m$ and is used for all the paths in order to provide a fair comparison when evaluating the impact of varying the level of σ used.

In summary, the 2D positioning accuracy (95%) results obtained for different level of

Table 24: Accuracy (95%) of the UWB speed estimation integration with varying level of σ

Path	σ	Accuracy (95%) (m)	RMSE (m)	Selected σ	Speed (m/s)
diag1	1σ	0.200	0.115	3σ	0.183
	2σ	0.204	0.115		
	3σ	0.193	0.113		
path1	1σ	0.985	0.466	3σ	0.366
	2σ	0.543	0.270		
	3σ	0.296	0.192		
diag4	1σ	0.286	0.179	1σ	0.386
	2σ	0.309	0.191		
	3σ	0.306	0.194		
path6	1σ	0.316	0.173	1σ	0.431
	2σ	0.334	0.188		
	3σ	0.344	0.194		
path15	1σ	0.198	0.115	1σ	0.483
	2σ	0.211	0.125		
	3σ	0.221	0.130		
diag8	1σ	0.367	0.195	3σ	0.565
	2σ	0.355	0.193		
	3σ	0.354	0.192		
path10	1σ	0.475	0.281	1σ	0.665
	2σ	0.488	0.257		
	3σ	0.576	0.265		
path14	1σ	0.913	0.661	3σ	0.721
	2σ	0.754	0.543		
	3σ	0.737	0.505		
path12	1σ	0.405	0.210	1σ	0.741
	2σ	0.405	0.212		
	3σ	0.486	0.254		

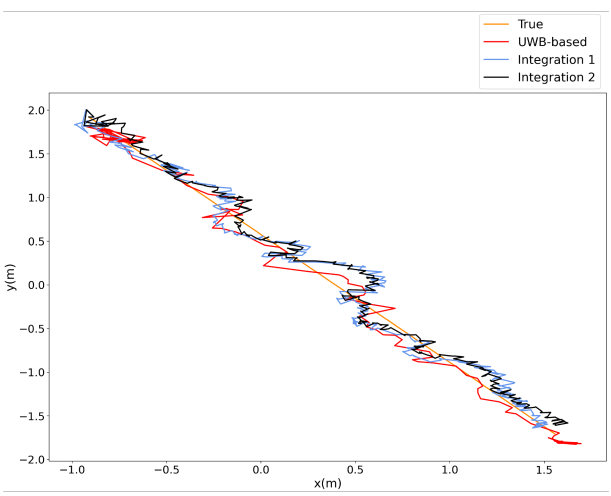
σ show that for some cases the use of 3σ, is better compared to 1σ. This suggests a dependency of the positioning accuracy on the size of the bound surrounding the range measurements, used during UWB speed estimation. A realistic bound is hence required to represent the uncertainty in the measurements. The level of σ resulting in the best performance, presented in table 24, are used in the next section. The next section compares Integrations 1 and 2 as well as the UWB-base positioning solution.

9.3.3. Comparison of different positioning methods

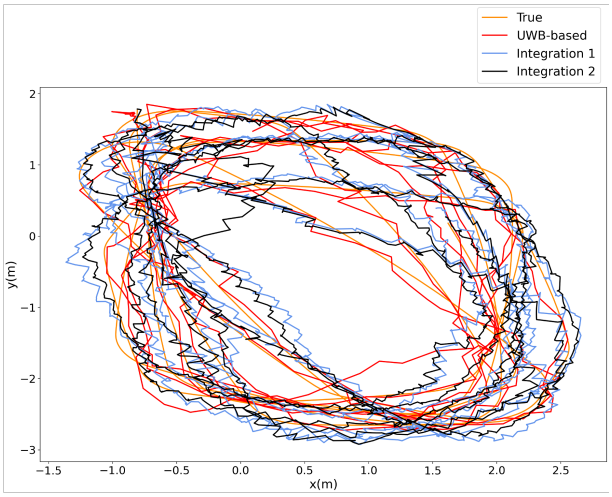
In this section the performance of Integrations 1, 2 and UWB-based multilateration method (discussed in chapter 7) are compared. The UWB-based solution relies only on the UWB range measurements, alternatively, as discussed in section 9.2.5 both integrations use data from both the IMU and UWB.

The UWB speed estimation method 2 (using the uncertainty bound) is used to calculate speed measurements in Integration 2. As explain in section 9.2.2, the σ of the range measurement is used to determine a bound around the range measurements, during the speed estimation process. As discussed in section 9.3.2, the level of σ resulting in the best performance in terms of positioning accuracy, varies between paths. The level of σ used in Integration 2 are presented in table 24.

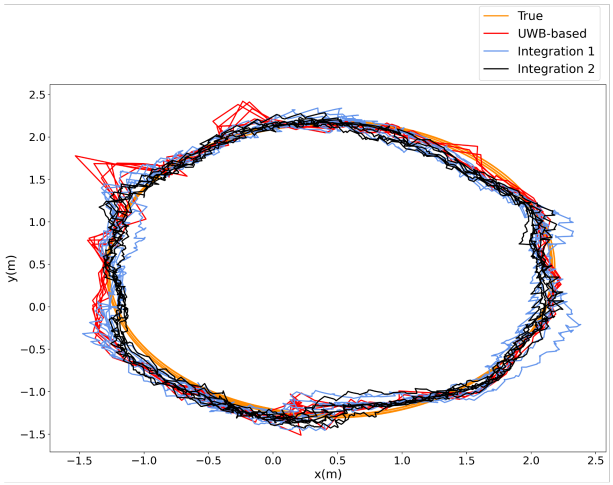
A visual representation of integrations 1, 2 and UWB-based positioning solutions is shown in figure 128. Figure 128, presents the solution for three paths, showing an example of each path type for which data were collected (a straight line path, a random path and a circular path). To evaluate and compare the performance of the three methods, their 2D position error distributions and the corresponding accuracy (95%) are further investigated.



(a) diag1: straight line path



(b) path10: random path



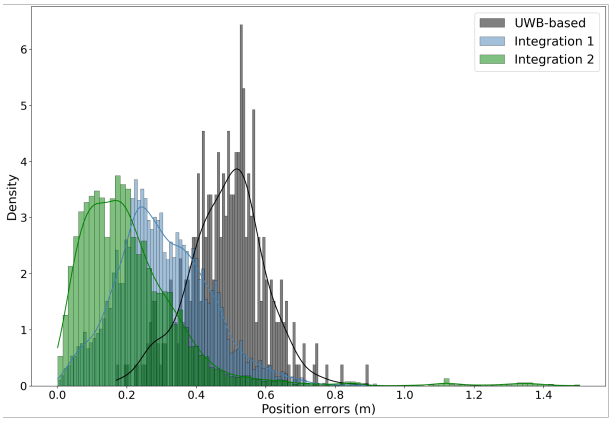
(c) path15: circular path

Figure 128: Positioning solution of straight line, random and circular path.

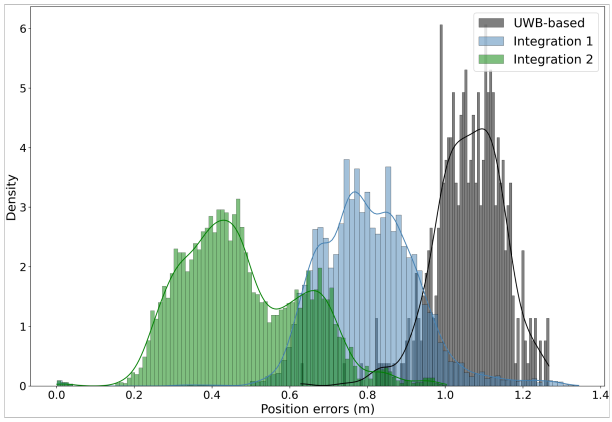
The 2D position error distributions of the UWB-based, Integrations 1 and 2, are shown in figure 129, for a selection of paths. The error distributions show that Integration 2 (with speed estimates) results in the lowest errors in the majority of the paths. This is also shown in table 25, where the accuracy (95%) of the methods is numerically evaluated. The calculated accuracy (95%) of Integration 2 is up to 68% better than Integration 1 and up to 67% better than the the UWB-based solution, as shown in table 25. Integration 2 improves the accuracy performance on average by 0.161m from Integration 1 and 0.223m

from UWB-standalone solution, both of which are significant improvements compared to the decimetric level of accuracy set in chapter 2. This illustrates the effectiveness of the additional UWB speed estimates in the IMU/UWB integration (Integration 2) in improving the accuracy performance of the system.

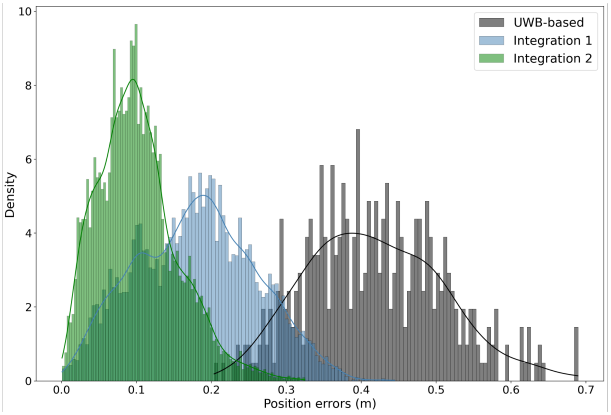
In addition, Integration 2 results in sub-meter level of accuracy (95%) (between 0.193cm - 0.737cm), well above the target maximum error of $2m$ set in chapter 2, for the examined scenarios. However, the decimeter level of accuracy (30cm) is not achieved for all the paths. This shows the effectiveness of Integration 2 in always achieving a positioning accuracy well above the maximum error target of 2m, although not always satisfying the most stringent requirement of 30cm set in chapter 2. Similar observations to accuracy (95%) are seen when considering the RMSE in table 26.



(a) path10: random path



(b) path14: circular path



(c) path15: circular path

Figure 129: 2D Position errors of random and circular paths.

In addition to analysing the improvement in performance between Integration 1, 2 and UWB-based solution, the impact of the different components presented in section 9.2 is examined. This is done by gradually adding the integration components and observing the changes in accuracy performance. The examined components are the UWB measurement corrections (discussed in section 9.2.1) and the UWB measurement filtering method (discussed in section 9.2.3). Their impact on accuracy (95%), is calculated for path15, with the results shown in table 27. The impact of the components is tested by gradually adding them in Integration 1 (without speed estimates), and comparing the accuracy performance to the UWB-based solution. As can be seen from table 27, there is about 1% improvement

Path Type	Path	Accuracy (95%)(m)			Difference (m) between Integration 2 and:		% Improvement in accuracy when Integration 2 is compared to:	
		Integration 2	Integration 1	UWB based	Integration 1	UWB based	Integration 1	UWB based
Random	path1	0.296	0.926	0.527	0.630	0.231	68.1%	43.8%
	path6	0.316	0.318	0.433	0.001	0.117	0.4%	26.9%
	path10	0.475	0.542	0.655	0.067	0.180	12.4%	27.5%
Circular	path12	0.405	0.506	0.668	0.101	0.263	20.0%	39.4%
	path14	0.737	1.005	1.199	0.267	0.462	26.6%	38.5%
	path15	0.198	0.313	0.575	0.115	0.377	36.6%	65.5%
Straight line	diag1	0.193	0.231	0.373	0.038	0.180	16.3%	48.2%
	diag4	0.286	0.332	0.463	0.046	0.176	13.8%	38.1%
	diag5	0.308	0.246	0.627	0.062	0.092	−25.3%	50.8%
	diag7	0.315	0.266	0.627	0.049	0.311	−18.4%	49.6%
	diag8	0.354	0.541	0.419	0.188	0.066	34.7%	15.6%

Table 25: Accuracy (95%) of Integrations 1, 2 and UWB based positioning solutions. Table also shows % improvement in accuracy when Integration 2 is compared to Integration 1 and UWB based solutions.

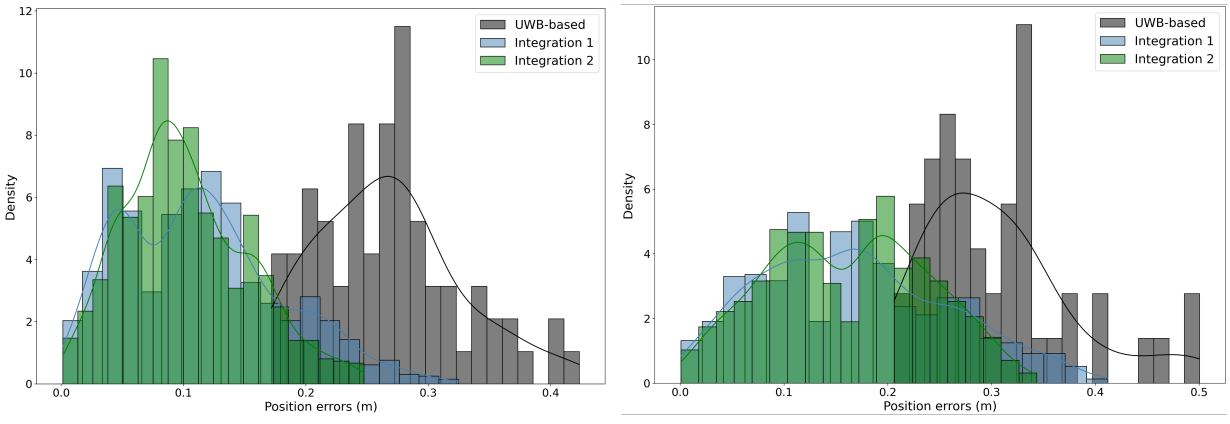
Path Type	Path	RMSE (m)			Difference (m) between Integration 2 and:		% Improvement in accuracy when Integration 2 is compared to:	
		Integration 2	Integration 1	UWB based	Integration 1	UWB based	Integration 1	UWB based
Random	path1	0.192	0.472	0.385	0.281	0.194	59.4%	50.3%
	path6	0.173	0.188	0.310	0.015	0.137	8.1%	44.2%
	path10	0.281	0.335	0.498	0.054	0.217	16.1%	43.5%
Circular	path12	0.210	0.312	0.528	0.102	0.318	32.7%	60.2%
	path14	0.505	0.812	1.065	0.308	0.560	37.9%	52.6%
	path15	0.115	0.197	0.427	0.082	0.312	41.6%	73.1%
Straight line	diag1	0.113	0.130	0.273	0.017	0.159	13.0%	58.4
	diag4	0.179	0.190	0.314	0.011	0.135	5.8%	42.9
	diag5	0.189	0.160	0.264	0.029	0.076	−17.9%	28.6%
	diag7	0.198	0.170	0.529	0.028	0.331	−16.6%	62.6
	diag8	0.192	0.274	0.326	0.082	0.134	30.0%	41.2%

Table 26: RMSE of Integrations 1, 2 and UWB based positioning solutions. Table also shows % improvement in accuracy when Integration 2 is compared to Integration 1 and UWB based solutions.

when including the filtering component (corresponding to a difference of 0.003m-0.006m: between 0.335m-0.332m or 0.319m-0.313m) and about 3% improvement when including UWB measurement corrections (corresponding to a difference of 0.016m-0.019m: between 0.335m-0.319m or 0.332m-0.313m). Both of these improvements are not considered significant at the decimeter level of accuracy. Alternatively, significant accuracy improvement of 20% (corresponding to a difference of 0.115cm: between 0.0313m-0.198m),is observed when the UWB speed estimates are included in the integration.

The small improvement (1%) resulting from the filtering component, is caused by the fact that only a few measurements are removed. This is seen in table 27, where about 99.7% of the measurements are used post measurement filtering. This might be caused by a rather ideal environment setting. In a more complex indoor environment, with increased NLOS and multipath propagation, the exclusion rate of the filter is expected to be higher.

Although Integration 2 (with speed estimates) results in the highest accuracy, this is not as clear for the straight line paths. Two examples of error distributions of straight line paths are shown in figure 130. Although, in table 25, Integration 2 seems to perform better than Integration 1, looking at their error distributions, the differences are small, of the cm level. Moreover, for some of the straight line paths (diag5 and diag7), Integration 1 performs better than Integration 2 (shown in table 25).



(a) diag1 (b) diag4

Figure 130: 2D Position errors of straight line paths.

Positioning method	Mode		Accuracy (95%)	%Improv. from UWB-standalone	Number UWB meas. available	Number UWB meas. used	% UWB meas. used
	Measurement Filtering	Measurement correction					
UWB-standalone	-	-	0.575	-	-	-	-
Integration 1	No	No	0.335	41.65%	3298	3298	100%
	Yes	No	0.332	42.19%	3298	3291	99.79%
	No	Yes	0.319	44.56%	3298	3298	100%
	Yes	Yes	0.313	45.57%	3298	3290	99.76%
Integration 2	Yes	Yes	0.198	65.50%	3298	3286	99.64%

Table 27: Accuracy (95%) performance when gradually including UWB measurement filtering and correction

The integration of IMU and UWB significantly increases the availability of the positioning system. In the case of path15, UWB-standalone results in 423 positioning solutions, while IMU/UWB integrations result in 6,837 positioning solutions, increasing the availability by 16 times. This shows that the integration improves both the accuracy performance and the availability of positioning solution.

The improvement in performance when moving from Integration 1 to Integration 2 is evidence that the introduction of UWB speed estimates in the integration is beneficial in terms of accuracy performance. The UWB speed estimates are additional sources of measurements that tackle the errors in accelerations, without the need of any additional sensors, keeping the complexity of the system to a minimum. The UWB speed estimation method can be adapted and used with any ranging system integrated with any system providing information of direction of motion.

9.4. Summary

This chapter presented the method developed for the integration of UWB and IMU systems. The chapter begun with a description of the quaternion-based ESKF which is used to integrate the two systems. The implementation of the ESKF was then described in section 9.2, where two integration implementations were proposed. The performance of the two implementations was then evaluated in section 9.3.

Implementing the two methods of UWB speed estimation (method 1: without uncertainty bound and method 2: with uncertainty bound) in the IMU/UWB integration and comparing the performance, showed that method 2 outperforms method 1 and improves positioning accuracy up to 61% and RMSE up to 75% compared to method 1. The choice of method 2, led to a sensitivity analysis of this method to varying levels of σ defining the bound around the range measurement. This analysis showed that the optimal level of σ varies across different paths, illustrating the need for a more realistic bound.

A comparison of Integration 1 (without UWB speed estimates), Integration 2 (with UWB speed estimates) and UWB-standalone solution, showed that Integration 2 outperforms Integration 1 and UWB-standalone solution. Integration 2 improves the accuracy

performance on average by 0.161m from Integration 1 and 0.223m from UWB-standalone solution, both of which are significant improvements compared to the decimetric level of accuracy set in chapter 2. This illustrates the effectiveness of the UWB speed estimation method.

The accuracy performance of the integration, with the addition of the components of: UWB measurement corrections, UWB measurement filtering and UWB speed estimation was calculated to evaluate the impact of each component. The addition of measurement filtering resulted in about 1% (0.003m-0.006m) accuracy improvement while the addition of UWB measurement corrections in about 3% (0.016m-0.019m) improvement. These two components showed insignificant improvements in accuracy, at the mm and cm level respectively.

The improved accuracy performance of the Integration 2 in comparison to Integration 1, shows the effectiveness of introducing the UWB speed estimates. The UWB range measurements reduce the impact of accelerations on position estimation, while the addition of UWB speed estimates reduce its impact on velocities. The proposed implementation results in sub-meter level of accuracy for the examined scenarios, which is always well above $2m$, although not always satisfying the most stringent requirement of 30cm set in chapter 2.

The additional UWB speed estimates do not require the need for an additional system, keeping the complexity of the system to a minimum. The UWB speed estimation method can be used with any ranging system to create additional measurements of speed.

Chapter 10

Conclusion & Further work

This chapter summarises the main findings of the research presented in this thesis. The six objectives formulated in chapter 1 were addressed in the remaining of the chapters. This chapter summarises the findings for each of the initial objectives.

10.1. Conclusions

A systematic approach is developed to identify the need of IPS. This approach starts by identifying the requirements at the service level through the location based services (LBS). In order to be exhaustive in the identification of location based functions (LBF), the functional system architecture (FSA) is developed and LBFs are identified. This leads to the final level of developing the physical level where the technologies used are integrated in a physical architecture. This framework ensures an exhaustive understanding of the operational level which leads to an optimal approach in solving the indoor positioning problem. The main conclusions of this thesis associated to each of the initial objectives are:

1. Objective 1 addressed in chapter 2: Indoor LBS are identified and safety critical applications are used to define the accuracy requirement at a decimeter level (30cm), which should not exceed 2m.

2. Objective 2 addressed in chapters 3 and 4: A FSA, transferable to any cognitive agent is developed and used to identify the LBFs.

3. The transferebility of the architecture is illustrated through its adjustment for the case of a human-driven vehicle and a VIP navigating with the aid of a cane.

4. Objective 3 addressed in chapter 5: Accounting for characteristics of UWB (long term stability/ short term instability) and IMU (short term stability/ long term instability), the two systems are complimentary and hence suitable for integration.

5. Objective 4 addressed in chapter 6: Empirical UWB and IMU data were collected through the design and execution of a data capture scheme to evaluate the performance of the two systems.

6. Objective 5 addressed in chapters 7 and 8:

   Chapter 7, detailed characterisation of UWB:

   6.1. In the static mode there are geometric dependencies of measurement errors

   6.2. In dynamic mode there is a dependency of measurement errors on the direction of motion with respect to the anchor.

   6.3. In dynamic mode there is an additional dependency of measurement errors on the speed of motion. However, since this phenomenon is not observed for all the anchors this might indicate a further dependency on the geometry between the anchor and the trajectory.

   6.4. The multilateration method outperforms the hyperbolic method, however the difference in performance is less than 5cm and is considered insignificant.

   Chapter 8, detailed characterisation of INS:

   6.1. Noisy IMU acceleration data are pre-processed, reducing their standard deviation by 80%.

   6.2. The major source of errors in the INS solution stems from the acceleration measurements.

7. Objective 6 addressed in chapter 9: Integration of UWB and IMU with the addition of UWB measurement corrections, UWB measurement filtering and UWB-derived speed measurements. The key conclusions of this chapter are:

7.1. UWB speed estimation with uncertainty bound outperforms UWB speed estimation without uncertainty bound, in terms of accuracy (95%), by 15%-60%.

7.2. There is a dependency of resulting position errors on the level of σ used to represent the uncertainty bound in UWB speed estimation method.

7.3. Significant improvement in accuracy when including UWB speed estimates in the integration. This improved the accuracy performance on average 0.161m from the integration without speed estimates, and 0.223 from UWB-standalone solution.

7.4. Insignificant improvement in accuracy when measurement filtering is used (at a mm level) and when measurement correction is applied (at a cm level)

7.5. Effectiveness of Integration 2 (IMU/UWB with speed estimations) in always achieving a positioning accuracy well above the maximum error target of 2m, although not always satisfying the most stringent requirement of 30cm set in chapter 2.

7.6. Significant improvement in positioning solution availability from UWB-stand alone to IMU/UWB integration. The number of positioning solutions increase by 16 times from 423 positioning solutions when using UWB-standalone to 6,837 when using IMU/UWB integration.

7.7. The UWB speed estimation method can be adapted and used with any ranging system to create additional measurements of speed, without the need for extra sensors.

10.2. Further Work

1. Extend IMU/UWB integration solution to 3D. The set-up of the experiment did not require the use of a 3D positioning solution, however, this is useful in situations like human navigation in multi-story buildings.

2. Geometric dependency of UWB measurement errors: Understanding any geometric dependencies of UWB measurement errors, and developing relationships reflecting

those dependencies will be beneficial is several ways. This will allow for improved corrections of UWB measurement errors and provide guidance for improved set-up of the system in complex environments.

3. The size of the uncertainty area in UWB speed estimation: Dependence of speed estimations on level of σ illustrates that a more realistic bound is required when estimating speeds. This will be achieved by further investigating the characteristics of the measurement error distributions and how those characteristics contribute to position estimation.

4. Use of a variable step size within the UWB speed estimation method: The use of a variable step size can result in a larger number of UWB-speed estimates included in the solution, potentially further improving the accuracy performance.

5. Integration error covariance matrices: In the presented integration error covariance matrices are developed using variance from static data. Monitoring the quality of the covariance matrix in every epoch and applying corrections if needed can result in improved performance.

6. UWB measurement corrections depending on direction of motion with respect to the anchor: The 0.95 threshold for determining whether the tag is moving towards or away from the anchor was set after tuning the threshold, using empirical data. Further investigation to find a scientific explanation of this relationship is needed.

References

Adams, J. A. (2007), Unmanned vehicle situation awareness: A path forward, *in* 'Human systems integration symposium', Citeseer, pp. 31–89.

Alarifi, A., Al-Salman, A., Alsaleh, M., Alnafessah, A., Al-Hadhrami, S., Al-Ammar, M. A. & Al-Khalifa, H. S. (2016), 'Ultra wideband indoor positioning technologies: Analysis and recent advances', *Sensors* **16**(5), 707.

AlHinai, N. (2020), Chapter 1 - introduction to biomedical signal processing and artificial intelligence, *in* W. Zgallai, ed., 'Biomedical Signal Processing and Artificial Intelligence in Healthcare', Developments in Biomedical Engineering and Bioelectronics, Academic Press, pp. 1–28.
URL: *https://www.sciencedirect.com/science/article/pii/B9780128189467000019*

Apple (2023), 'Ultra wideband availability', Available from: https://support.apple.com/en-gb/109512. [Accessed 29th February 2024].

Aqel, M. O., Marhaban, M. H., Saripan, M. I. & Ismail, N. B. (2016), 'Review of visual odometry: types, approaches, challenges, and applications', *SpringerPlus* **5**, 1–26.

Arents, J. & Greitans, M. (2022), 'Smart industrial robot control trends, challenges and opportunities within manufacturing', *Applied Sciences* **12**(2), 937.

Baker, C. R., Ferguson, D. I. & Dolan, J. M. (2008), 'Robust mission execution for autonomous urban driving'.

Basiri, A., Lohan, E. S., Moore, T., Winstanley, A., Peltola, P., Hill, C., Amirian, P. & e Silva, P. F. (2017), 'Indoor location based services challenges, requirements and usability of current solutions', *Computer Science Review* **24**, 1–12.

Basnyake, C., Williams, T., Alves, P. & Lachapelle, G. (2010), 'Can gnss drive v2x?', Available from: https://www.gpsworld.com/transportationroadcan-gnss-drive-v2x-10611/. [Accessed 23rd October 2023].

Behere, S. & Törngren, M. (2016), 'A functional reference architecture for autonomous driving', *Information and Software Technology* **73**, 136–150.

Berger, C. & Rumpe, B. (2014), 'Autonomous driving-5 years after the urban challenge: The anticipatory vehicle as a cyber-physical system', *arXiv preprint arXiv:1409.0413* .

Brigadnov, I., Lutonin, A. & Bogdanova, K. (2023), 'Error state extended kalman filter localization for underground mining environments', *Symmetry* **15**(2), 344.

Che, F., Ahmed, Q. Z., Lazaridis, P. I., Sureephong, P. & Alade, T. (2023), 'Indoor positioning system (ips) using ultra-wide bandwidth (uwb)—for industrial internet of things (iiot)', *Sensors* **23**(12), 5710.

Chebrolu, N. (2019), 'Extended kalman filter localization', University of Bonn. Course: Mobile Sensing and Robotics.

Chen, P.-C. (1999), A non-line-of-sight error mitigation algorithm in location estimation, *in* 'WCNC. 1999 IEEE Wireless Communications and Networking Conference (Cat. No. 99TH8466)', Vol. 1, IEEE, pp. 316–320.

Cominelli, M., Patras, P. & Gringoli, F. (2019), Dead on arrival: An empirical study of the bluetooth 5.1 positioning system, *in* 'Proceedings of the 13th international workshop on wireless network testbeds, experimental evaluation & characterization', pp. 13–20.

Cross, P. (1983), 'Working paper no. 6: Advanced least squares applied to position-fixing'.

Dardari, D., Conti, A., Ferner, U., Giorgetti, A. & Win, M. Z. (2009), 'Ranging with ultra-wide bandwidth signals in multipath environments', *Proceedings of the IEEE* **97**(2), 404–426.

Donges, E. (1999), 'A conceptual framework for active safety in road traffic', *Vehicle System Dynamics* **32**(2-3), 113–128.

Du, J., Masters, J. & Barth, M. (2004), Lane-level positioning for in-vehicle navigation and automated vehicle location (avl) systems, *in* 'Proceedings. The 7th International IEEE Conference on Intelligent Transportation Systems (IEEE Cat. No. 04TH8749)', IEEE, pp. 35–40.

D'Alfonso, L., Lucia, W., Muraca, P. & Pugliese, P. (2015), 'Mobile robot localization via ekf and ukf: A comparison based on real data', *Robotics and Autonomous Systems* **74**, 122–127.

El-Rabbany, A. (2006), *Introduction to GPS: the global positioning system 2nd Edition*, Artech house.

El-Sheimy, N. & Youssef, A. (2020), 'Inertial sensors technologies for navigation applications: State of the art and future trends', *Satellite Navigation* **1**(1), 1–21.

Elsanhoury, M., Mäkelä, P., Koljonen, J., Välisuo, P., Shamsuzzoha, A., Mantere, T., Elmusrati, M. & Kuusniemi, H. (2022), 'Precision positioning for smart logistics using ultra-wideband technology-based indoor navigation: A review', *IEEE Access* **10**, 44413–44445.

Endsley, M. R. (1988), Design and evaluation for situation awareness enhancement, *in* 'Proceedings of the Human Factors Society annual meeting', Vol. 32, SAGE Publications Sage CA: Los Angeles, CA, pp. 97–101.

Endsley, M. R. (1995), 'Toward a theory of situation awareness in dynamic systems', *Human factors* **37**(1), 32–64.

Endsley, M. R. (2016), *Designing for situation awareness: An approach to user-centered design*, CRC press.

Fan, R. & Du, X. (2022), 'Nlos error mitigation using weighted least squares and kalman filter in uwb positioning', *arXiv preprint arXiv:2205.05939* .

Feng, D., Wang, C., He, C., Zhuang, Y. & Xia, X.-G. (2020), 'Kalman-filter-based integration of imu and uwb for high-accuracy indoor positioning and navigation', *IEEE Internet of Things Journal* **7**(4), 3133–3146.

Fukuju, Y., Minami, M., Morikawa, H. & Aoyama, T. (2003), 'Dolphin: An autonomous indoor positioning system in ubiquitous computing environment.', *WSTFES* **3**, 53.

Gehrke, J. D. (2009), Evaluating situation awareness of autonomous systems, *in* 'Performance Evaluation and Benchmarking of Intelligent Systems', Springer, pp. 93–111.

GOV (2018), 'Satellite-derived time and position: Blackett review', Available from: https://www.gov.uk/government/publications/satellite-derived-time-and-position-blackett-review. [Accessed 18th January 2019].

Groves, P. D. (2013), *Principles of GNSS, inertial, and multisensor integrated navigation systems 2nd Edition*, Artech house.

Gu, Y., Lo, A. & Niemegeers, I. (2009), 'A survey of indoor positioning systems for wireless personal networks', *IEEE Communications surveys & tutorials* **11**(1), 13–32.

Guo, A., Zhou, Z., Zhu, X. & Bai, F. (2019), 'Low-cost sensors state estimation algorithm for a small hand-launched solar-powered uav', *Sensors* **19**(21), 4627.

GVR (2022), 'Indoor location-based services market size, share & trends analysis report by product, by technology, by application (monitoring, tracking, proximity, navigation, others), by end-use, by region, and segment forecasts, 2023 - 2030', Available from: https://www.grandviewresearch.com/industry-analysis/indoor-lbs-market. [Accessed 22nd December 2023].

Hol, J. D., Dijkstra, F., Luinge, H. & Schon, T. B. (2009), Tightly coupled uwb/imu pose estimation, *in* '2009 IEEE international conference on ultra-wideband', IEEE, pp. 688–692.

Huang, J., Junginger, S., Liu, H. & Thurow, K. (2023), 'Indoor positioning systems of mobile robots: A review', *Robotics* **12**(2), 47.

Jo, K., Kim, J., Kim, D., Jang, C. & Sunwoo, M. (2015), 'Development of autonomous car—part ii: A case study on the implementation of an autonomous driving system based on distributed architecture', *IEEE Transactions on Industrial Electronics* **62**(8), 5119–5132.

Kang, J. M., Yoon, T. S., Kim, E. & Park, J. B. (2020), 'Lane-level map-matching method for vehicle localization using gps and camera on a high-definition map', *Sensors* **20**(8), 2166.

Kelly, J. & Sukhatme, G. S. (2009), Visual-inertial simultaneous localization, mapping and sensor-to-sensor self-calibration, *in* '2009 IEEE International Symposium on Computational Intelligence in Robotics and Automation-(CIRA)', IEEE, pp. 360–368.

Khalajmehrabadi, A., Gatsis, N. & Akopian, D. (2017), 'Modern wlan fingerprinting indoor positioning methods and deployment challenges', *IEEE Communications Surveys & Tutorials* **19**(3), 1974–2002.

Kunz, F., Nuss, D., Wiest, J., Deusch, H., Reuter, S., Gritschneder, F., Scheel, A., Stübler, M., Bach, M., Hatzelmann, P. et al. (2015), Autonomous driving at ulm university: A modular, robust, and sensor-independent fusion approach, *in* '2015 IEEE intelligent vehicles symposium (IV)', IEEE, pp. 666–673.

Lawrence, A. (2001), *Modern inertial technology: navigation, guidance, and control*, Springer Science & Business Media.

Li, D., Landry, R. & Lavoie, P. (2008), Low-cost mems sensor-based attitude determination system by integration of magnetometers and gps: A real-data test and performance

evaluation, *in* '2008 IEEE/ION Position, Location and Navigation Symposium', IEEE, pp. 1190–1198.

Li, X., Wang, J. & Li, T. (2013), 'Seamless positioning and navigation by using geo-referenced images and multi-sensor data', *Sensors* **13**(7), 9047–9069.

Lin, S.-C., Zhang, Y., Hsu, C.-H., Skach, M., Haque, M. E., Tang, L. & Mars, J. (2018), The architectural implications of autonomous driving: Constraints and acceleration, *in* 'Proceedings of the Twenty-Third International Conference on Architectural Support for Programming Languages and Operating Systems', ACM, pp. 751–766.

Liu, H., Darabi, H., Banerjee, P. & Liu, J. (2007), 'Survey of wireless indoor positioning techniques and systems', *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)* **37**(6), 1067–1080.

Liu, W., Song, D., Wang, Z. & Fang, K. (2019), 'Comparative analysis between error-state and full-state error estimation for kf-based imu/gnss integration against imu faults', *Sensors* **19**(22), 4912.

Liu, Y. & Ozguner, U. (2007), Human driver model and driver decision making for intersection driving, *in* '2007 ieee intelligent vehicles symposium', IEEE, pp. 642–647.

Luettel, T., Himmelsbach, M. & Wuensche, H.-J. (2012), 'Autonomous ground vehicles—concepts and a path to the future', *Proceedings of the IEEE* **100**(Special Centennial Issue), 1831–1839.

Madgwick, S. et al. (2010), 'An efficient orientation filter for inertial and inertial/magnetic sensor arrays', *Report x-io and University of Bristol (UK)* **25**, 113–118.

Madyastha, V., Ravindra, V., Mallikarjunan, S. & Goyal, A. (2011), Extended kalman filter vs. error state kalman filter for aircraft attitude estimation, *in* 'AIAA Guidance, Navigation, and Control Conference', p. 6615.

Makki, A., Siddig, A., Saad, M. & Bleakley, C. (2015), 'Survey of wifi positioning using time-based techniques', *Computer Networks* **88**, 218–233.

Malajner, M., Planinšič, P. & Gleich, D. (2015), Uwb ranging accuracy, *in* '2015 International Conference on Systems, Signals and Image Processing (IWSSIP)', IEEE, pp. 61–64.

Marković, L., Kovač, M., Milijas, R., Car, M. & Bogdan, S. (2022), Error state extended kalman filter multi-sensor fusion for unmanned aerial vehicle localization in gps and magnetometer denied indoor environments, *in* '2022 International Conference on Unmanned Aircraft Systems (ICUAS)', IEEE, pp. 184–190.

Matthaei, R. & Maurer, M. (2015), 'Autonomous driving–a top-down-approach', *at-Automatisierungstechnik* **63**(3), 155–167.

Mautz, R. (2012), 'Indoor positioning technologies'.

Mazhar, F., Khan, M. G. & Sällberg, B. (2017), 'Precise indoor positioning using uwb: A review of methods, algorithms and implementations', *Wireless Personal Communications* **97**(3), 4467–4491.

Mendoza-Silva, G. M., Torres-Sospedra, J. & Huerta, J. (2019), 'A meta-review of indoor positioning systems', *Sensors* **19**(20), 4507.

Meng, X., Roberts, S., Cui, Y., Gao, Y., Chen, Q., Xu, C., He, Q., Sharples, S. & Bhatia, P. (2018), 'Required navigation performance for connected and autonomous vehicles: where are we now and where are we going?', *Transportation planning and technology* **41**(1), 104–118.

MI (2020), 'Location-based services market-growth, trends, and forecast (2020-2025)', Available from: https://www.reportlinker.com/p05877818/Location-based-Services-Market-Growth-Trends-and-Forecast.html?utm_source=PRN. [Accessed 3rd February 2023].

Mobileye (2024), 'The mobileye safety methodology', Available from: https://www.mobileye.com/technology/safety-methodology/. [Accessed 29th February 2024].

Montemerlo, M., Thrun, S., Dahlkamp, H., Stavens, D. & Strohband, S. (2006), Winning the darpa grand challenge with an ai robot, *in* 'AAAI', pp. 982–987.

Naser, R. S., Lam, M. C., Qamar, F. & Zaidan, B. (2023), 'Smartphone-based indoor localization systems: A systematic literature review', *Electronics* **12**(8), 1814.

Niloy, M. A., Shama, A., Chakrabortty, R. K., Ryan, M. J., Badal, F. R., Tasneem, Z., Ahamed, M. H., Moyeen, S. I., Das, S. K., Ali, M. F. et al. (2021), 'Critical design and control issues of indoor autonomous mobile robots: A review', *IEEE Access* **9**, 35338–35370.

Nur, K., Feng, S., Ling, C. & Ochieng, W. (2013), 'Integration of gps with a wifi high accuracy ranging functionality', *Geo-spatial Information Science* **16**(3), 155–168.

Ochieng, W., Shardlow, P. & Johnston, G. (1999), 'Advanced transport telematics positioning requirements: An assessment of gps performance in greater london', *The Journal of Navigation* **52**(3), 342–355.

Ochieng, W. Y., Sauer, K., Walsh, D., Brodin, G., Griffin, S. & Denney, M. (2003), 'Gps integrity and potential impact on aviation safety', *The journal of navigation* **56**(1), 51–65.

OptiTrack (2023*a*), 'How is 3d motion data obtained in an optical motion capture system?', Available from: https://optitrack.com/support/faq/general.html. [Accessed 12th July 2023].

OptiTrack (2023*b*), 'Motion capture cameras', Available from: https://optitrack.com/cameras/primex-13/. [Accessed 6th April 2023].

Pascacio, P., Casteleyn, S., Torres-Sospedra, J., Lohan, E. S. & Nurmi, J. (2021), 'Collaborative indoor positioning systems: A systematic review', *Sensors* **21**(3), 1002.

Pau, G., Arena, F., Gebremariam, Y. E. & You, I. (2021), 'Bluetooth 5.1: An analysis of direction finding capability for high-precision location services', *Sensors* **21**(11), 3589.

Pendleton, S., Andersen, H., Du, X., Shen, X., Meghjani, M., Eng, Y., Rus, D. & Ang, M. (2017), 'Perception, planning, control, and coordination for autonomous vehicles', *Machines* **5**(1), 6.

Perakis, H. & Gikas, V. (2018), Evaluation of range error calibration models for indoor uwb positioning applications, *in* '2018 International conference on indoor positioning and indoor navigation (IPIN)', IEEE, pp. 206–212.

Ponnan, S., Shelly, S., Hussain, M. Z., Ashraf, M., Haldorai, A. et al. (2022), 'Autonomous navigation system based on a dynamic access control architecture for the internet of vehicles', *Computers and Electrical Engineering* **101**, 108037.

Poole, D. L. & Mackworth, A. K. (2010), *Artificial Intelligence: foundations of computational agents*, Cambridge University Press.

Poulose, A., Eyobu, O. S., Kim, M. & Han, D. S. (2019), Localization error analysis of indoor positioning system based on uwb measurements, *in* '2019 Eleventh International Conference on Ubiquitous and Future Networks (ICUFN)', IEEE, pp. 84–88.

Pozyx (2022), 'Where to place the anchors', Available from: https://www.pozyx.io/pozyx-academy/where-to-place-the-anchors. [Accessed 6th March 2023].

Priyantha, N. B., Chakraborty, A. & Balakrishnan, H. (2000), The cricket location-support system, *in* 'Proceedings of the 6th annual international conference on Mobile computing and networking', pp. 32–43.

Qi, L., Liu, Y., Yu, Y., Chen, L. & Chen, R. (2024), 'Current status and future trends of meter-level indoor positioning technology: A review', *Remote Sensing* **16**(2), 398.

Quddus, M. A., Ochieng, W. Y. & Noland, R. B. (2007), 'Current map-matching algorithms for transport applications: State-of-the art and future research directions', *Transportation research part c: Emerging technologies* **15**(5), 312–328.

Randeniya, D. I., Sarkar, S. & Gunaratne, M. (2010), 'Vision–imu integration using a slow-frame-rate monocular vision system in an actual roadway setting', *IEEE Transactions on Intelligent Transportation Systems* **11**(2), 256–266.

Reddy, N. & Sujatha, B. (2011), 'Tdoa computation using multicarrier modulation for sensor networks', *Int. J. Comput. Sci. Commun. Netw* **1**(1), 85–90.

Ren, J. & Xia, D. (2023), *Autonomous driving algorithms and Its IC Design*, Springer.

Roumeliotis, S. I., Sukhatme, G. S. & Bekey, G. A. (1999), Circumventing dynamic modeling: Evaluation of the error-state kalman filter applied to mobile robot localization, *in* 'Proceedings 1999 IEEE International Conference on Robotics and Automation (Cat. No. 99CH36288C)', Vol. 2, IEEE, pp. 1656–1663.

Russell, S. J. & Norvig, P. (2010), *Artificial intelligence: a modern approach*, Pearson Education.

Sahinoglu, Z., Gezici, S. & Guvenc, I. (2008), 'Ultra-wideband positioning systems', *Cambridge, New York* .

Samama, N. (2019), *Indoor positioning: technologies and performance*, John Wiley & Sons.

Samsung (2021), 'What is ultra-wideband, and how does it work?', Available from: https://insights.samsung.com/2021/08/25/what-is-ultra-wideband-and-how-does-it-work-3/. [Accessed 29th February 2024].

Santoro, L., Nardello, M., Brunelli, D. & Fontanelli, D. (2023), 'Uwb-based indoor positioning system with infinite scalability', *IEEE Transactions on Instrumentation and Measurement* **72**, 1–11.

Schroeder, J., Galler, S., Kyamakya, K. & Jobmann, K. (2007), Nlos detection algorithms for ultra-wideband localization, *in* '2007 4th Workshop on Positioning, Navigation and Communication', IEEE, pp. 159–166.

Sciavicco, L. & Siciliano, B. (2012), *Modelling and control of robot manipulators*, Springer Science & Business Media.

Shunsuke, K., Yanlei, G. & Hsu, L.-T. (2015), Gnss/ins/on-board camera integration for vehicle self-localization in urban canyon, *in* '2015 IEEE 18th International Conference on Intelligent Transportation Systems', IEEE, pp. 2533–2538.

Singh, J., Tyagi, N., Singh, S., Ali, F. & Kwak, D. (2024), 'A systematic review of contemporary indoor positioning systems: Taxonomy, techniques, and algorithms', *IEEE Internet of Things Journal* .

Skrypchuk, L., Langdon, P., Sawyer, B. D., Mouzakitis, A. & Clarkson, P. J. (2018), 'Enabling multitasking by designing for situation awareness within the vehicle environment', *Theoretical Issues in Ergonomics Science* pp. 1–24.

Sola, J. (2017), 'Quaternion kinematics for the error-state kalman filter', *arXiv preprint arXiv:1711.02508* .

Stachniss, C. (2014*a*), 'Introduction to robot mapping', University of Freiburg. Course: Simultaneous Localization and Mapping.

Stachniss, C. (2014*b*), 'Unscented kalman filter', University of Freiburg. Course: Simultaneous Localization and Mapping.

Taş, Ö. Ş., Kuhnt, F., Zöllner, J. M. & Stiller, C. (2016), Functional system architectures towards fully automated driving, *in* '2016 IEEE Intelligent Vehicles Symposium (IV)', IEEE, pp. 304–309.

Tian, Y., Lian, Z., Wang, P., Wang, M., Yue, Z. & Chai, H. (2024), 'Application of a long short-term memory neural network algorithm fused with kalman filter in uwb indoor positioning', *Scientific reports* **14**(1), 1925.

Udacity (2020), 'Carnd-capstone project', Available from: https://github.com/48cfu/Udacity-Real-Self-Driving-Car-Carla. [Accessed 29th February 2024].

Ulbrich, S. & Maurer, M. (2015), Towards tactical lane change behavior planning for automated vehicles, *in* '2015 IEEE 18th International Conference on Intelligent Transportation Systems', IEEE, pp. 989–995.

Urmson, C., Anhalt, J., Bagnell, D., Baker, C., Bittner, R., Clark, M., Dolan, J., Duggins, D., Galatali, T., Geyer, C. et al. (2008), 'Autonomous driving in urban environments: Boss and the urban challenge', *Journal of Field Robotics* **25**(8), 425–466.

Urmson, C., Baker, C., Dolan, J., Rybski, P., Salesky, B., Whittaker, W., Ferguson, D. & Darms, M. (2009), 'Autonomous driving in traffic: Boss and the urban challenge', *AI magazine* **30**(2), 17–17.

Waymo (2024), 'Waymo driver', Available from: https://waymo.com/waymo-driver/. [Accessed 29th February 2024].

Wayve (2021), 'A new approach to self-driving: Av2.0', Available from: https://wayve.ai/thinking/a-new-approach-to-self-driving-av2-0/. [Accessed 29th February 2024].

Wayve (2024), 'Pioneering a new way to solve self-driving with embodied ai', Available from: https://wayve.ai/technology/AV2.0. [Accessed 29th February 2024].

Wei, J., Snider, J. M., Gu, T., Dolan, J. M. & Litkouhi, B. (2014), A behavioral planning framework for autonomous driving, *in* '2014 IEEE Intelligent Vehicles Symposium Proceedings', IEEE, pp. 458–464.

Wei, J., Snider, J. M., Kim, J., Dolan, J. M., Rajkumar, R. & Litkouhi, B. (2013), Towards a viable autonomous driving research platform, *in* '2013 IEEE Intelligent Vehicles Symposium (IV)', IEEE, pp. 763–770.

Werner, M. (2014), *Indoor location-based services: Prerequisites and foundations*, Springer.

WHO (2023), 'Blindness and vision impairment', Available from: https://www.who.int/news-room/fact-sheets/detail/blindness-and-visual-impairment. [Accessed 22nd March 2023].

Wirola, L., Laine, T. A. & Syrjärinne, J. (2010), Mass-market requirements for indoor positioning and indoor navigation, *in* '2010 International Conference on Indoor Positioning and Indoor Navigation', IEEE, pp. 1–7.

Wise, E., Li, B., Gallagher, T., Dempster, A. G., Rizos, C., Ramsey-Stewart, E. & Woo, D. (2012), Indoor navigation for the blind and vision impaired: Where are we and where are we going?, *in* '2012 international conference on Indoor positioning and indoor navigation (IPIN)', IEEE, pp. 1–7.

Won, S.-h., Melek, W. & Golnaraghi, F. (2008), Position and orientation estimation using kalman filtering and particle diltering with one imu and one position sensor, *in* '2008 34th Annual Conference of IEEE Industrial Electronics', IEEE, pp. 3006–3010.

Won, S.-h. P., Melek, W. W. & Golnaraghi, F. (2009), 'A kalman/particle filter-based position and orientation estimation method using a position sensor/inertial measurement unit hybrid system', *IEEE Transactions on Industrial Electronics* **57**(5), 1787–1798.

Woolley, M. (2019), 'Bluetooth direction finding', *A technical Overview* .

Wymeersch, H., Seco-Granados, G., Destino, G., Dardari, D. & Tufvesson, F. (2017), '5g mmwave positioning for vehicular networks', *IEEE Wireless Communications* **24**(6), 80–86.

x-io Technologies (2022), 'Ngimu user manual v1.6'.

Xia, J., Wang, Y., Jiang, B. & Xiong, G. (2022), Research on indoor positioning system based on ble-aoa/uwb technology, *in* '2022 41st Chinese Control Conference (CCC)', IEEE, pp. 5100–5105.

Xu, Y., Li, Q., Xu, X., Yang, J. & Chen, Y. (2023), 'Research progress of nature-inspired metaheuristic algorithms in mobile robot path planning', *Electronics* **12**(15), 3263.

Yao, L., Wu, Y.-W. A., Yao, L. & Liao, Z. Z. (2017), An integrated imu and uwb sensor based indoor positioning system, *in* '2017 International Conference on Indoor Positioning and Indoor Navigation (IPIN)', IEEE, pp. 1–8.

Yassin, A., Nasser, Y., Awad, M., Al-Dubai, A., Liu, R., Yuen, C., Raulefs, R. & Aboutanios, E. (2016), 'Recent advances in indoor localization: A survey on theoretical approaches and applications', *IEEE Communications Surveys & Tutorials* **19**(2), 1327–1346.

Youn, W. & Gadsden, S. A. (2019), 'Combined quaternion-based error state kalman filtering and smooth variable structure filtering for robust attitude estimation', *IEEE Access* **7**, 148989–149004.

Zhao, H. & Wang, Z. (2011), 'Motion measurement using inertial sensors, ultrasonic sensors, and magnetometers with extended kalman filter for data fusion', *IEEE Sensors Journal* **12**(5), 943–953.

Zhong, Y., Liu, T., Li, B., Yang, L. & Lou, L. (2018), Integration of uwb and imu for precise and continuous indoor positioning, *in* '2018 Ubiquitous Positioning, Indoor Navigation and Location-Based Services (UPINLBS)', IEEE, pp. 1–5.

Zhu, N., Marais, J., Bétaille, D. & Berbineau, M. (2018), 'Gnss position integrity in urban environments: A review of literature', *IEEE Transactions on Intelligent Transportation Systems* **19**(9), 2762–2778.

Zhuang, Y., Zhang, C., Huai, J., Li, Y., Chen, L. & Chen, R. (2022), 'Bluetooth localization technology: Principles, applications, and future trends', *IEEE Internet of Things Journal* **9**(23), 23506–23524.

Zucker, R. (2019), 'How self-driving cars think: Navigating double-parked vehicles', Available from: https://www.getcruise.com/news/blog/2019/how-self-driving-cars-think-navigating-double-parked-vehicles/?$_gl = 1 * 1kah7va *_g cl_au * MTE2NzQwMzU1Mi4xNzI3NjgwMzY5$. [$Accessed 29th February 2024$].