University of L'Aquila

Department of Information Engineering, Computer Science and Mathematics

Ph.D. Program in ICT - Systems Engineering, Telecommunications and HW/SW Platforms

XXXVI cycle

Accurate and reliable positioning solutions through multi-constellation and multi-sensor approaches

SSD - ING-INF/03

Ph.D. Student

Angel Luis Zuriarrain Sosa

Course Coordinator

Prof. Vittorio Cortellessa

Advisor

Prof. Fortunato Santucci

Co-Advisor

Ph.D. Roberto Alesii

A.A. 2022/2023

I dedicate this to my family and friends,
who were always there for me.

Abstract

With the development of advanced driver assistance systems (ADAS) and autonomous vehicles (AV), recent years have seen an increasing evolution of on-board sensors and communication systems capable of interacting with available infrastructures, including satellite constellations and other systems that provide helpful information for the localization process. Therefore, it is essential to develop solutions that employ a multi-sensor approach to ensure accurate and reliable positioning in different navigation scenarios. This research proposes positioning solutions based on the Software-Defined Radio (SDR) paradigm, utilizing the Global Navigation Satellite System (GNSS). The approach is characterized by multi-constellation, multi-frequency, and augmented capabilities and can be enhanced by integrating with other sources of information. Moreover, exploring emerging technologies within the localization process contributes to creating more resilient and robust systems. In this context, this thesis investigates the implementation of services capable of timely receiving, decoding, and processing wireless signals. Two SDR-based case studies are presented: the first focused on positioning using ADS-B signals, and the other involved a distributed network designed to offer sensing and localization services supporting next-generation mobile networks.

The thesis is structured into three main sections. The first section provides a theoretical background, with an extensive description of the technological aspects and critical issues involved. The second section includes the most relevant publications produced or presented during this doctoral program. Finally, the third section presents the conclusions and outlines directions for future research.

Initially, a study on GNSS identifies the primary limitations and physical phenomena affecting the system. This analysis includes the development of SDR-based solutions capable of simulating and receiving GNSS signals.

Detailed investigations identify the key factors influencing localization systems' accuracy, availability, continuity, and integrity. This examination involves a comparison of different SDR platforms through experimental activities conducted in controlled environments and scenarios involving the processing of real GNSS signals.

Subsequently, alternative systems capable of providing sensing and positioning services are explored. Two approaches are considered: the SDR-based receiver calculates its position using information from mobile anchors, and another where distributed and synchronized anchors detect the transmitter or interference source. The first approach is addressed as opportunistic positioning using ADS-B signals, while the second approach deals with sensing and localization using an SDR-based distributed sensor network.

This research also presents an architecture for navigation based on a multi-sensor approach. This architecture is implemented to develop an on-board unit (OBU) within the framework of the EMERGE project. This solution addresses the challenge of reducing GNSS system errors by utilizing augmentation services that provide atmospheric and clock corrections. Additionally, the process of sensor fusion using GNSS and inertial measurement data, as well as the results obtained from field tests, are presented.

Acknowledgements

I would like to thank my advisor, Prof. Fortunato Santucci, and my co-advisor, Dr. Roberto Alesii, for their support and guidance throughout this challenging journey. Their encouragement and dedication have been instrumental in shaping the course of this research.

A heartfelt thanks to my friends Amleto, Graziano, Mario and Paola, who have been there for me every step of the way.

A special note of thanks goes to Dr. Alex Piccioni and Dr. Andrea Marotta for their insightful advice and assistance, which have contributed significantly to the success of this work.

I am also grateful to my friends and colleagues at the Center of Excellence Ex-EMERGE and RadioLabs for their incredible contributions and camaraderie during this research.

I would also like to acknowledge Telespazio S.p.A. for their support in making this work possible.

Finally, I want to thank my wife for her patience, understanding, and constant support.

Thank you for your support and for making this challenging an enriching experience.

List of Publications

J: Journal publication; **C:** Conference publication.

Appended Papers:

C1 A. L. Z. Sosa, R. Alesii and F. Santucci, "Cross-platform evaluation for Software Defined Radio GNSS receiver," *2022 3rd URSI Atlantic and Asia Pacific Radio Science Meeting (AT-AP-RASC)*, 2022, pp. 1-4, DOI: 10.23919/AT-APRASC54737.2022.9814436.

C2 A. L. Z. Sosa, R. Alesii and F. Santucci, "Opportunistic RSS-based localisation using SDR and ADS-B system," *2024 4th URSI Atlantic Radio Science Meeting (AT-RASC)*, Meloneras, Spain, 2024, pp. 1-4, DOI: 10.46620/URSIATRASC24/JTMJ9870.

C3 A. Piccioni, A. L. Z. Sosa, R. Alesii and F. Graziosi, "SDR-Based Distributed System for Mobile Communication Network Monitoring and Support," ACCEPTED IN *Next-Generation Multimedia Services at the Edge: Leveraging 5G and Beyond - co-located with ISCC June 26, 2024 // Paris, France.*

J1 Zuriarrain Sosa, A.L.; Ioannucci, V.; Pratesi, M.; Alesii, R.; Albanese, C.; Valentini, F.; Cinque, E.; Martinelli, A.; Brizzi, M. OBU for Accurate Navigation through Sensor Fusion in the Framework of the EMERGE Project. *Appl. Sci. 2024, 14, 4401.* DOI: 10.3390/app14114401.

Contents

List of Figures

List of Tables

Part I

Thesis Background

Chapter 1

Introduction

The world population reached 8 billion people on 15 November 2022: a milestone in human history. The world's population has grown from 7 billion to 8 billion in only twelve years. This accelerated population growth challenges managing natural resources and emissions, especially in urban areas [1]. In fact, issues related to the trend of populations to live in large urban centres are among the most critical challenges of our time. This implies that cities are the first place where innovations must lead us towards a new model of sustainable development, i.e. development that meets the needs of the present without compromising the ability of future generations to meet their own needs. This is the key meaning of smart cities, and the entire urban ecosystem has a crucial role in achieving this sustainable future.

Sustainable development is impossible without considering social, economic and environmental issues when calling for innovation. A sustainable economy enables cities to make the long-term investments necessary to build and maintain adequate infrastructure to provide efficient services, develop a social environment open to citizens, and encourage and support business activities without compromising the natural environment.

Information and Communication Technologies (ICTs) are not the only ingredients of a smart city, but they are certainly among the enablers. By considering the city as a complex system, ICTs simplify the complexity management and allow us to address sustainable development in an integrated way. ICTs allow us to measure and analyse complex phenomena and facilitate real-time short- and long-term planning and decision-making. ICT thus acts as the foundation of a smart city. Meaningful use of data requires the existence and development of a large-scale ICT infrastructure in the urban environment. Moreover, the design of this infrastructure must itself

be smart. It is vital to add intelligence to this system and to provide scalability, robustness and flexibility.

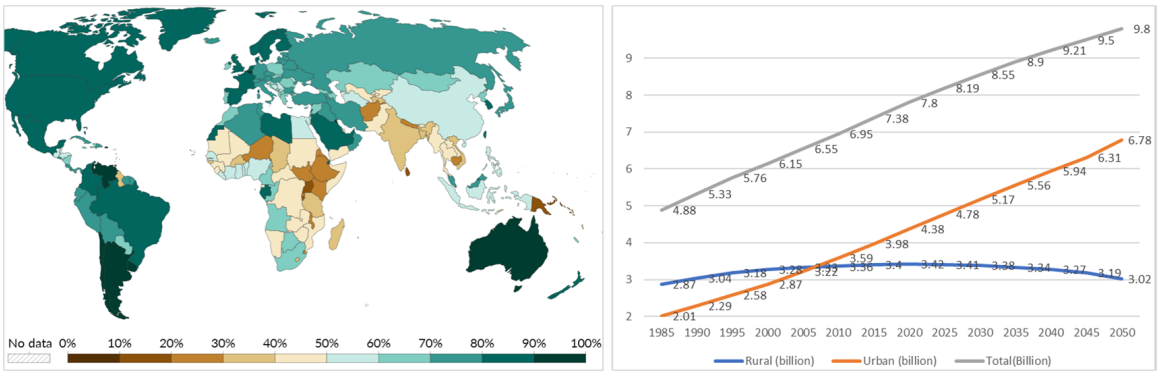

Figure 1.1: UN World Urbanization Prospects (until 2050). Source: United Nations, Department of Economic and Social Affairs, Population Division (2018)

Smart mobility constitutes a key aspect in the development of smart cities. Smart mobility includes high flexibility, scalability, integration, efficiency in energy resources, and automated vehicles (AVs) or "self-driving" cars. There are many approaches behind smart mobility: the use of technologies that provide helpful information: IoT, sensors, networks, and satellites; the creation of structures that include smart roads and monitoring platforms; and solutions that target the environment-friendly electric car and cycling path. Safely and efficiently, positioning, navigation, and timing (PNT) services support most "smart" processes in mobility. These services seem immediate when applications such as traffic management, access control, autonomous mobility, precise positioning, public health and safety, critical infrastructures, or security are some of the goals driving innovation in modern urban scenarios [2, 3].

Space vertical applications are critical in the smart cities ecosystem as technology enablers at the core of profitable business models [4]. Although spatial service offerings have expanded in many directions, the Global Navigation Satellite System (GNSS) 's global coverage and free provision of absolute positioning solutions remain fundamental. With the increasing adoption and availability of GNSS signals, frequencies and services, user technologies have evolved and become widespread in many devices and applications. In parallel, GNSS augmentation systems and high-accuracy services have mitigated major errors due to atmospheric conditions or receiver clock limitations.

However, vulnerability to signal degradation and attenuation in urban environments and dense vegetation indoors would cause GNSS to be unreliable in location

accuracy. These issues can reduce accuracy or result in total signal loss in certain situations, such as urban canyons or galleries. Furthermore, GNSS signals can be affected by atmospheric conditions, multipath interference and satellite geometry, which can introduce errors in positioning and timing. In addition, GNSS signals are susceptible to deliberate interference that can alter their availability and integrity. These challenges highlight the need for alternative solutions and technologies to improve the performance and robustness of GNSS in challenging environments.

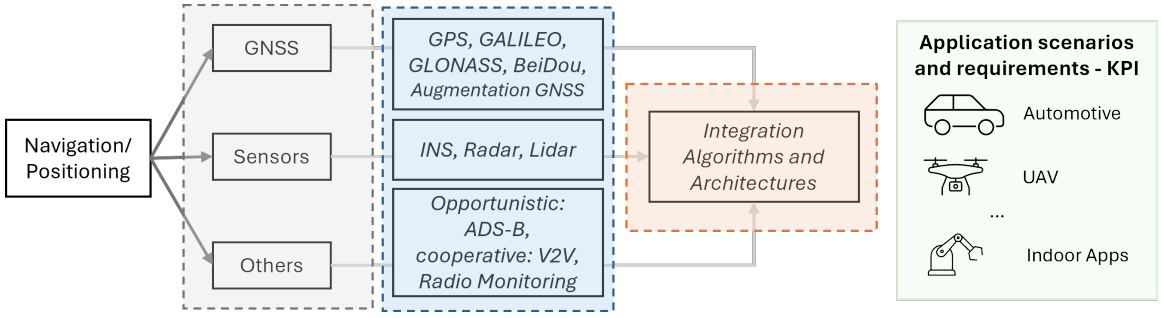

Figure 1.2: General architecture for Multi-sensor integrated navigation/positioning

Combining GNSS with other systems could compensate for the limitations and leverage their strengths to achieve continuous positioning. For this reason, the most advanced navigation/positioning systems include inertial navigation, GNSS (multi-constellation) approach and a group of systems and sensors that may include LiDAR, visible light positioning (VPL), wireless networks (WiFi, Bluetooth, 5-6G), radio frequency identification (RFID), Ultra Wide Band (UWB), among others [5].

Due to their limitations, it is difficult to use single-sensor-based navigation/positioning systems to provide robust, accurate, and seamless solutions. For instance, the Inertial Navigation System (INS) is a relative positioning technology and only provides an accurate solution for a limited time, as both inertial sensor errors and integration errors will cause the solution to diverge. Thus, other absolute positioning data sources, such as GNSS, are usually needed; however, although they are accurate in open-sky environments, they suffer from signal blockage and multipath in urban areas and other GNSS-challenging environments. Other wireless positioning systems, such as WiFi and Bluetooth, usually have limitations, such as high dependency on Access Points (APs) distribution, noisy and unstable solutions, labour costs for building databases, and fluctuating Received Signal Strengths (RSS). Magnetic positioning is usually used indoors for local, as opposed to global positioning. Vision positioning, which often uses a camera to capture an object's motion, offers accurate localization at a relatively low cost; however, it has limitations, such as privacy

issues, the extraction of features from environments, and considerable computation resources. Therefore, achieving positioning performance goals while satisfying application requirements using a single-sensor modality is difficult. Consequently, to improve positioning performance, the advantages of different kinds of sensors can be fully exploited to improve reliability, robustness, and spatial and temporal coverage. The system's general architecture is depicted in 1.2. Taking a GNSS/INS integrated system as an example, GNSS provides the position and velocity to aid inertial navigation by reducing cumulative errors, while INS fills the gap in challenging environments by filtering its noise; hence, integration reduces the limitations of both constituent parts. Thus, the integration component is presented as an important element in the localization process. There are three main approaches to systems integration: Loosely coupled (LC), Tightly Coupled (TC) and Ultra Tightly Coupled Integration (uTC). LC integrates different sources, based on position and velocity levels, while TC integrates sensor data in ranging levels between transmitters and the receiver. Unlike LC and TC, uTC integrates raw measurements from a GNSS receiver at a deeper level, such as raw carrier and code phases. Multi-sensor integrated systems can be used in many applications, such as automotive, aircraft, maritime, spacecraft, indoor mobile robots, and unmanned aerial vehicles (UAVs). A significant effort of this research has been dedicated to advancing industrial navigation solutions through a multi-sensor approach, with a primary focus on GNSS-based systems. These solutions include the development of an On-Board Unit (OBU) within the framework of the EMERGE project (Connected, geo-localized, and cybersecure vehicles), representing a practical application in the automotive domain.

The complex scenario of modern localization, the availability of emerging technologies, and the increasing demands in various fields have driven the need for efficient, adaptable, and flexible solutions. In telecommunications, Software Defined Radio (SDR) stands out as a crucial tool, enabling the implementation of traditional hardware components (such as filters, modulators, and detectors) through software. The recent advancements in integration and computing power have led to the creation of more versatile and high-performing SDR platforms. Consequently, this research heavily relies on SDR-based solutions. Specifically, SDR-based GNSS receiver solutions with multi-constellation capabilities (GPS and GALILEO) are proposed as a valuable framework for studying geostationary satellite positioning. Additionally, SDR technology has been applied in localization processes that depend on the timely reception of ADS-B signals. In this context, the entire process—ADS-B signal reception, filtering, detection, and localization—is integrated into a single SDR-based

solution. Furthermore, SDR has proven essential in controlled environments for signal studies and experimental communication channel analysis. Another section of this thesis focuses on sensing and localization via distributed sensor systems. Here, the SDR paradigm is used to develop a distributed service aligned with the International Telecommunication Union (ITU) recommendations for 5G and beyond. This involves a distributed network of SDR devices designed to support mobile networks. In this work, the above approaches and solutions are described in the next sections and investigated in the next chapter.

1.1 Software-Defined Radio

With modern advances in computing technologies, digital signal processing (DSP) and digital communication algorithms, artificial intelligence, radio frequency (RF) hardware design, and many other elements have evolved modern communication systems into complex, intelligent, high-performance platforms that can adapt to operational environments and deliver large amounts of information in real-time [6]. One of the most important milestones in the last decade in communication systems technology is the software-defined radio, or SDR, which adopts the most recent advances in all fields to yield the ultimate transmitter and receiver.

The current communications systems use a combination of digital processing and RF analogue parts. The tendency is to increase the digital parts which are implemented by software. The use of flexible RF hardware has resulted in systems able to implement different functionalities from a single front-end. This combination of flexible HW and digital signal processing is what defines SDR. This translates to supporting various features and functionalities, such as updating and upgrading through reprogramming, without the need to replace the hardware on which they are implemented. This opens the door to the possibility of realizing multi-band and multi-functional communication devices [7].

The hardware component in a SDR platform is the front-ends, based on all the analog elements like filters, amplifiers and mixers, and the block responsible for the Digital-to-Analog Conversion (DAC) in the transmitter and the Analog-to-Digital Conversion (ADC) in the receiver [8,9]; in Figure 1.3 **(b)** is represented a comparison between SDR device and the equivalent ad hoc platform in PHY and MAC ISO/OSI layers. The main difference with classic radio platforms is in the signal processing: SDR devices offer digital signal processing, enabling the flexibility to switch between different digital processing while classic devices require ad hoc development; the main

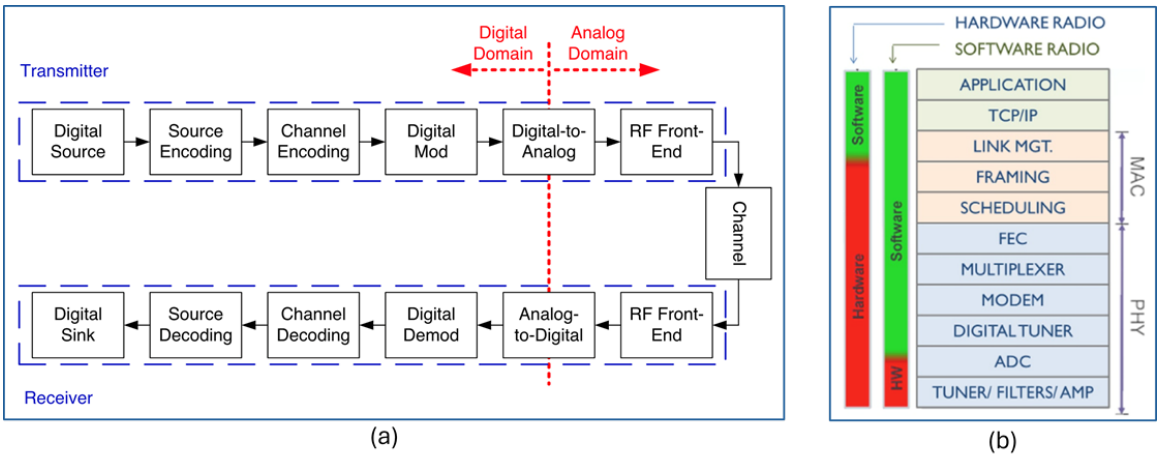

Figure 1.3: (a) Schematic of a Digital Communication System. (b) Hardware-software division of RF functionalities in SDR and traditional radio platforms

trade-off is on the performance since the software-based development is certainly slower than the hardware, but the use of application-specific integrated circuits or FPGAs allows different design methodologies with different performance, depending on the necessity [10].

1.1.1 SDR architecture

In SDR architecture we have to distinguish between RF architecture and processing architecture. The RF architecture refers directly to the front-end configuration and can generally be superheterodyne or zero-IF, while the processing architecture refers to the microprocessor system employed.

RF architecture

A superheterodyne architecture is recommended for applications that tolerate very low spurious emissions, high selectivity, and adjustable bandwidth. The action of the intermediate frequency filters and the gain distribution between the Intermediate Frequency (IF) stages allow optimisation of the noise figure and linearity. Although the performance of superheterodyne receivers has been optimised throughout the chain, elements such as filters still occupy a physically considerable space. In addition, some applications require the development of frequency-specific components.

An alternative to superheterodyne architecture, which has re-emerged as a potential solution in recent years, is zero-IF architecture (ZIF). potential solution in recent years, is the zero-IF architecture (ZIF). A ZIF receiver utilizes a single-frequency

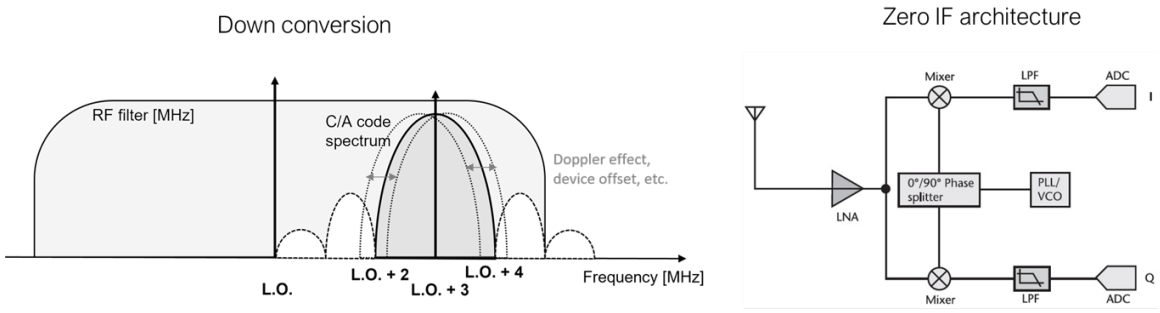

Figure 1.4: Zero IF architecture and down conversion

mixing stage with the local oscillator (LO) set directly to the frequency band of interest (see Figure 1.4), downconverting the received signal to baseband in phase (I) and quadrature (Q) components. This architecture avoids the stringent filtering requirements of the superheterodyne since all analog filtering takes place at baseband, where filters are much easier to design and less expensive than custom RF/IF filters [6]. The ADC and DAC are now operating on I/Q data at baseband, so the sample rate relative to the converted bandwidth can be reduced, saving significant power and complexity [11].

However, it is important to note that a direct conversion from frequency to baseband can increase the possibility of carrier leakage and also generates an unwanted frame rate component. Due to real-world factors, such as process variation and temperature changes in the signal chain, it is virtually impossible to maintain a perfect phase shift (90 degrees) between I and Q signals. In addition, imperfect L.O. isolation at the mixing stage introduces carrier leakage components. If not corrected, image and carrier leaks can degrade receiver sensitivity and create unwanted transmit spectral emissions. During the development of this PhD research, the advantages and limitations of SDR platforms based on Zero IF architecture have been taken into account. The higher level of Front-End integration, the low-cost character and the low power consumption are some aspects that lead us to develop strategies to mitigate the drawbacks of this architecture. The main strategies focus on digital filtering and correlation techniques in the detection of the carrier in accurate mode, therefore mitigating the L.O. instabilities in the downconversion process.

Processing architecture

When the RF information is passed to the Analogue-to-Digital conversion process (in the case of reception), the digital samples are received by the baseband processing

block. The processing of the digital samples applied one of the following methodologies or processor architectures:

- **General-purpose processor (GPP):** A GPP is a digital circuit that is clock-driven and register-based. It is capable of processing different functions and operates on data streams represented in the binary system. These GPPs can be used for several purposes, making them extremely useful for an unlimited number of applications. This eliminates the need for building application-specific circuits, reducing the overall cost of running applications. GPPs are generally a preferable hardware platform by researchers in academia due to their flexibility, abundance, and ease of programmability, which is one of the main requirements in SDR platforms.

- **Digital signal processors (DSPs):** DSP is a particular type of microprocessor that is optimized to process digital signals. To help understand how DSPs are distinguished from GPPs, we should first note that both are capable of implementing and processing complex arithmetic tasks. Tasks like modulation/demodulation, filtering, and encoding/decoding are commonly and frequently used in applications that include speech recognition, image processing, and communication systems. DSPs, however, implement them more quickly and efficiently due to their architecture (e.g., RISC-like architecture, parallel processing), which is specifically optimized to handle arithmetic operations, especially multiplications. Since DSPs are capable of delivering high-performance with lower power, they are better candidates for SDR deployment compared to GPPs [12].

- **Field programmable gate arrays (FPGAs):** Using FPGAs for custom digital signal processing applications is more efficient because they can implement fully parallel algorithms. DSP applications use many binary multipliers and accumulators that can be implemented in dedicated DSP slices. Tools like MathWorks HDL Coder are making creating new modules and targeting FPGAs easier, as it can generate portable, synthesizable Verilog and VHDL code from MATLAB functions, Simulink models, and Stateflow charts.

- **Graphics processing units (GPUs):**GPUs are processors specifically designed to handle graphics-related tasks, and they efficiently process large blocks of streaming data in parallel. SDR platforms that are comprised of both GPPs and GPUs are flexible and have a higher level of processing power. However,

this results in a lower level of power efficiency. GPUs act as co-processors to GPPs because a GPP is required to act as the control unit and transfer data from external memory. After a transfer is completed, the GPU executes signal processing algorithms [7].

- **Hybrid design:** The fourth approach towards realizing SDRs is the hybrid approach, where both hardware and software-based techniques are combined into one platform. This is commonly referred to as the co-design or hybrid approach.

1.1.2 SDR frameworks

The development environments and platforms most commonly used in the software development of SDR applications are: MATLAB & Simulink, Vivado HLS & SDSoC, GNU Radio, CUDA and LabVIEW. In this work we will concentrate on the most widely used in support of the solutions presented in Part II.

- **MATLAB & Simulink:** Most designers start with modeling and simulating the system using Mathworks MATLAB & Simulink [13]. With the availability of a wide range of built-in functions and toolboxes, especially for signal processing and communication, developing and testing applications became very common and widely adopted. However, in order to use these models for different platforms, developers would need to use MATLAB Coder and Simulink Coder to generate C/C++ codes. The generated codes can be used with Embedded Coder to optimize them and generate software interfaces with AXI drivers for the sake of running on embedded processors and microprocessors. Alternatively, developers can use the HDL Coder to generate low-level code (Verilog, VHDL) for FPGAs or other embedded processors.

- **GNU Radio:** GNU Radio is an open-source software development tool for the implementation of Software Defined Radios on Linux, MacOS or Windows platforms. Basically, it is used for all kinds of ISAC (Integrated Sensing and Communications) and GNSS applications. Some of the main hardware blocks available in GNU Radio are filters, modulators, demodulators, encoders, decoders, synchronisation layers, templates of transmitters and receivers, and other elements that make up a radio system. The code in GNU Radio is mainly developed in Python and C++ programming languages. It is also possible to

develop applications using the graphical user interface, which allows the implementation of communications systems from blocks that perform basic signal processing functions [14].

- **LabVIEW:** LabVIEW is a graphical programming environment that provides unique productivity accelerators for test system development, such as an intuitive approach to programming, connectivity to any instrument, and fully integrated user interfaces. It is similar to GNU Radio and Simulink, where the design can be constructed schematically by connecting a chain of various blocks together, each of which performs a certain function. It also offers complete support for Ettus devices (USRP) to enable rapid prototyping of communications systems. Designing different blocks of the system can be achieved using high-level languages, such as C or MATLAB, or using a graphical dataflow.

During the period of this research, the three software environments described above have been used. Matlab & Simulink is one of the most widespread cross-platform tools with excellent functionalities that have been added to the dedicated toolboxes. In fact, all the studies, signal analysis, filtering and development presented in this document, of GNSS receivers and simulators are based on this environment. LabVIEW has been used for real-time acquisition applications using USRP devices at high sample rates and integrated with Matlab script. GNU Radio has also been used in the development of solutions that allow the reception of signals that allow the development of solutions for opportunistic positioning.

1.1.3 Advantages, challenges and applications

The adoption of Software Defined Radio (SDR) technology has revolutionized the field of Information and Communication Technologies (ICTs) by introducing significant advancements in flexibility, performance, and cost-effectiveness. Unlike traditional hardware-based systems, SDR-based solutions can be easily updated and reprogrammed via software, allowing for rapid adaptation to new signals, standards, and protocols. This makes them highly versatile and future-proof, capable of accommodating evolving technologies without the need for extensive hardware modifications. Cost-effectiveness is another major benefit of SDR technology. By consolidating multiple functions onto a single hardware platform through software, the need for multiple dedicated hardware units is eliminated, leading to reduced overall costs. This not only simplifies the design and maintenance of communication systems but also makes advanced capabilities more accessible.

Advanced signal processing techniques such as adaptive filtering, multipath mitigation, and interference cancellation significantly drive the use of SDR. These techniques can be implemented more efficiently in software, improving the sensitivity, accuracy, and reliability of the receivers. Moreover, SDR-based systems can support multiple communication standards and protocols by simply implementing new software modes. This multi-standard support provides better coverage, redundancy, and performance, which are critical for future applications. However, integrating SDR technology is not without challenges. The high computational requirements for real-time signal processing necessitate powerful processors or FPGAs, which can increase power consumption and complexity. Due to the intensive processing demands, latency issues may also arise, potentially affecting the systems' performance.

Integrating SDR-based solutions with existing legacy systems poses another challenge due to compatibility issues. Ensuring seamless operation and interoperability with traditional hardware may require additional effort and resources. Despite these challenges, the advantages and potential applications of SDR-based systems are vast and varied. SDR technology finds applications in various fields, such as navigation and positioning, telecommunications, scientific research, public safety, and defence. In navigation and positioning, SDR-based systems are widely used in vehicles, aircraft, and maritime vessels for accurate and reliable positioning.

A prominent application of SDR technology is in Global Navigation Satellite Systems (GNSS). SDR-based GNSS receivers [15] and simulators offer enhanced flexibility, cost-effectiveness, and performance by enabling advanced signal processing techniques and multi-constellation support. These systems are used in high-precision applications such as navigation and localization, geospatial surveying, autonomous vehicles, and scientific research. The application of SDR technology in GNSS significantly improves their accuracy, reliability, and adaptability to different signal environments, making them a critical component in modern navigation systems.

1.2 Global Navigation Satelite System

Global Navigation Satellite Systems are the most commonly used resource in the localization process. GNSS satellites continuously transmit radio signals in the L frequency band (1.2 to 1.6 GHz). A receiver interprets the ranging codes and navigation data included in such signals, allowing the identification of the transmitting satellites and their positions, as well as computing the travelling time through space and the consequent range information. The localization is based on solving geometric

problems involving the distances of a user to at least four GNSS satellites with known coordinates. The trilateration in radio navigation uses a physical principle that, in a vacuum, the propagation of an electromagnetic wave has a constant and known speed, that is, the speed of light c = 299792458 m/s. Therefore, if the time of flight or the Time Of Arrival (TOA) of the signal from known locations to the user can be measured, the distances can be calculated by multiplying that time by the speed of light. The main GNSS systems, like GPS or Galileo, are TOA systems. Furthermore, as said, the distances are given by time measurements. This leads to the assumption of the perfect synchronization between the satellite's clock and the user's clock. To solve the system correctly, it is necessary to have an additional equation to consider the unknown, referred to as the clock difference.

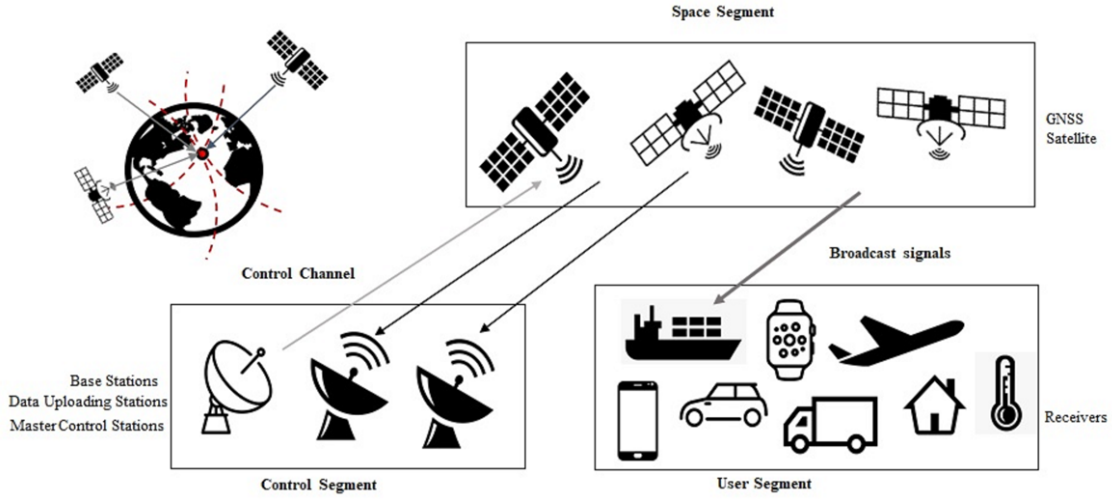

Figure 1.5: GNSS-based localization and architecture

The architecture of the GNSS system basically consists of three main segments: the space segment, which comprises the satellites; the control segment, which is responsible for the proper operation of the system; and the user segment, which includes the GNSS receivers [16].

Space segment

The space segment is composed by the satellites positioned in different orbital planes. The satellites are devoted to transmitting signals, storing and broadcasting the navigation messages kept updated by the control segment. In the case of GPS, the nominal constellation comprises 24 satellites, as depicted in 1.1. The satellites are arranged in six orbital planes equally spaced, and they are placed in a Medium Earth Orbit (MEO) orbit at an altitude of 20200 km and an inclination of 55° with respect

to the equator. The satellites have nearly circular orbits and a period of 11 hours, 58 minutes and 2 seconds. The MEO orbit gives the possibility to have one satellite visible for several hours in each pass, and the constellation has an adequate number of satellites to have full coverage of the Earth. Galileo, in the Full Operational Capability (FOC) phase of the project, consists of 24 satellites and 6 in-orbit spares intended to prevent any interruption in service. The satellites are in MEO orbit at an altitude of 23222 km. Galileo has three orbital planes inclined at 56◦ with respect to the equator. The period is about 14 hours, 4 minutes and 45 seconds and guarantees at least six satellites in view from any point on the Earth. The nominal Glonass constellation consists of 24 MEO satellites deployed in three orbital planes, with eight equally spaced in each plane. The orbits are roughly circular and at an altitude of 19 100km with a nominal period of 11 hours, 15 minutes and 44 seconds, repeating the geometry every eight sidereal days. The Beidou constellation (Phase III) consists of 35 satellites, including 5 Geostationary Orbit (GEO) satellites and 30 non-GEO satellites in a nearly circular orbit. The non-GEO satellites include 3 Inclined Geosynchronous Satellite Orbit (IGSO) ones, and 27 MEO satellites orbiting at an altitude of 21 528km in three orbital planes with an inclination of about 55 degree and with an orbital period of about 12 hours and 53 minutes, repeating the ground track every seven sidereal days.

There are two other systems under development by Japan and India: Quasi-Zenith Satellite System (QZSS) and Indian Regional Navigation Satellite System (IRNSS) respectively. Currently, QZSS and IRNSS are regional systems, but a further global expansion is foreseen in the coming years.

Table 1.1: GNSS Open Signal characteristics

Constellation	Availability and Coverage	Carrier Frequency Center band[GHz]	Open Service Signals and Spreading Modulation	Code[Mcps]/ Data[bps] Rate	Minimum Bandwith [MHz]	Received Power [dBW]
GPS (US)	- 31 Satellites on sky - Global Coverage - Fully Operational	L1: 1.57542	- C/A: BPSK(1) - L1C: MBOC(6,1,1/11) (non-fully operational)	1.023/50	2.046 4.092	-158.5 -157
		L2: 1.22760	- L2C: BPSK(1) (pre-operational)	1.023/25	2.046	-161.5
		L5: 1.17645	- L5: BPSK(10) (pre-operational)	10.23/50	20.46	-157.9
GALILEO (EU)	- 24 Satellites on sky - 22 Satellites in usable condition - Global Coverage	E1: 1.57542	- E1 OS: MBOC(6,1,1/11)	1.023/125	8.184	-157
		E5a: 1.17645 E5b: 1.20714	- E5a-b: BPSK(10)	10.23/25-125	20.46	-155
BeiDou Phase III (CHN)	- 35 Satellites on sky - Global Coverage - Fully Operational	B1C: 1.57542	- B1-C: MBOC(6,1,1/11)	1.023/50	32.736	-159/-161
		B2: 1.17645/1.20714	- B2-a,b: AltBOC(15,10)	10.23/50	20.46	-163
GLONASS (RUS)	- 24 Satellites on sky - Global Coverage - Fully Operational	G1: 1.59806-1.60931 G2: 1.24293-1.25168	- C/A: BPSK(0.511) (FDMA)	0.511/50	15	-161 -167
		G3: 1.202025	- BPSK(10) (CDMA)	10.23/100	20.46	-161

Control segment

The control segment (also called the ground segment) is responsible for properly operating the GNSS. Its primary functions are to control and maintain the satellite constellation's status and configuration, predict ephemeris and satellite clock evolution, keep the corresponding GNSS time scale (through atomic clocks), and update the navigation messages for all the satellites.

For example, in case of GPS, the control segment is a network of Monitoring Station (MS), a Master Control Station (MCS) and the Ground Antennas (GA). The MS are stations spread worldwide which collect GPS data from all the satellites. The MCS is the core of the network, it collects all the data coming from the MS to estimate the ephemeris and clock errors. The Alternate Master Control Station (AMCS) is a functional backup station for the MCS. The transmitted data includes ephemerides and clock correction, so the navigation message is updated.

The Galileo ground segment controls the entire satellite constellation and involves two Ground Control Centre (GCC), Telemetry, Tracking and Control (TT&C) stations, nine Mission Uplink Stations (ULS) and a worldwide network of Galileo Sensor Stations (GSS). The Ground Control Segment (GCS) is responsible for the constellation control and management of Galileo satellites. It provides the TT&C function for the whole satellite constellation. Its functional elements are deployed within the GCCs and the globally distributed TT&C stations. The TT&C stations use S-band frequency antennas 13m in diameter to secure data exchange between the control centres and satellites. The Ground Mission Segment (GMS) is responsible for the determination and uplink of the navigation and integrity data messages needed to provide the navigation and UTC time transfer service. The GMS includes a worldwide network of GSS, continuously collecting data to be processed by GCC for determining Galileo navigation and integrity data messages. Each GSS has three parallel reception channels: one channel for determining orbit data and clock synchronisation, a second for integrity determination, and a third for redundant channel. The global geographical distribution of such stations has been selected to ensure permanent access to any constellation satellite at any time. The two GCCs constitute the core of the ground segment. Two redundant elements are located in Fucino (Italy) and Oberpfaffenhofen (Germany). Some of their main functions are:

- Orbit determination and synchronisation;
- Control of all Galileo satellites and uploading navigation data messages;

- Monitoring and control, performance monitoring, performance prediction and maintenance management functions of the ground segment elements;
- Generation of navigation messages;
- Computation of Galileo System Time (GST) and provision of a reliable and stable coordinated time reference for the Galileo system.

User segment

The user segment consists of GNSS receivers. Their main function is to receive GNSS signals, determine pseudoranges (and other observables) and solve the navigation equations to obtain the coordinates and provide a very accurate time. The basic elements of a generic GNSS receiver are an antenna with pre-amplification, a radio frequency section, a microprocessor, an intermediate-precision oscillator, a feeding source, some memory for data storage and an interface with the user [17].

1.2.1 GNSS signals characteristics

GNSS satellites continuously transmit navigation signals at two or more frequencies in the L band. These signals contain ranging codes and navigation data to allow users to compute both the travel time from the satellite to the receiver and the satellite coordinates at any epoch. The main signal components are described as follows:

- Carrier: Radio frequency sinusoidal signal at a given frequency.
- Ranging code: Sequences of zeros and ones which allow the receiver to determine the travel time of the radio signal from the satellite to the receiver. They are called PRN sequences or PRN codes.
- Navigation data: A binary-coded message providing information on the satellite ephemeris (pseudo-Keplerian elements or satellite position and velocity), clock bias parameters, almanacs (with a reduced-accuracy ephemeris data set), satellite health status and other complementary information.

Global Positioning System (GPS)

GPS was the first to provide global coverage and currently has 31 satellites in orbit. It offers four "open" or "civil" type signals distributed in the L1, L2 and L5 bands: L1-C/A, L1C, L2C and L5C (see Figure 1.6). The GPS uses the CDMA technique to send different signals on the same radio frequency, and the modulation method used

is Binary Phase Shift Keying (BPSK).[1] The L1-C/A signal is the most diffused signal for market applications. The Pseudorandom Noise Code (PRN) is an unique Gold code, of 1 millisecond in length at a chipping rate of 1.023 Mbps. Although the coding frequency is the same, the modulation determines a minimum receive bandwidth of 2.046 MHz for L1-C/A and 4.092 MHz for L1C. The minimum receive power is higher for the new civil signal L1C. L2C signal is composed by two different PRN codes to provide ranging information; the civil-moderate code (called CM), and the civil-long length code (called CL). The CM code is 10230 bits long, repeating every 20 ms. The CL code is 767250 bits long, repeating every 1500 ms. Each signal is transmitted at 0.511 Mbits per second (Mbit/s); however, they are multiplexed together to form a 1.023 Mbit/s signal. The L5C signal was designed for users requiring Safety of Life (SoL) applications. There are two signal components: the in-phase component (L5I) with data and ranging code, both modulated via BPSK onto the carrier; and the quadrature component (L5Q), with no data but also having a ranging code BPSK modulated onto the carrier. This signal has an improved code/carrier tracking loop, and its high power and design provide robustness against interference. It is important to note that L1C, L2C and L5C will be of limited use until they are broadcast from 18 to 24 satellites [18].

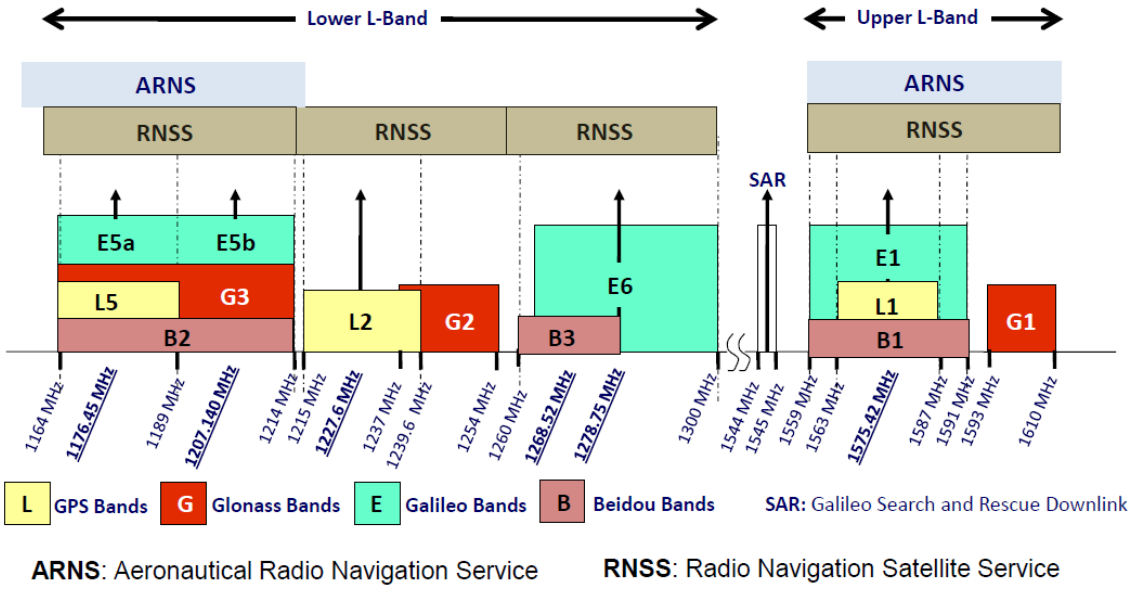

Figure 1.6: GNSS frequency allocation

[1]The L1C signal is designed to enable interoperability between GPS and other international satellite navigation systems. Multiplexed Binary Offset Carrier (MBOC) modulation improves mobile reception in cities and other challenging environments. L1C comprises the L1C-I data channel and L1C-Q pilot channel.

EU´s Global Navigation Satellite System (GALILEO)

Galileo satellites permanently transmit three independent Code Division Multiple Access signals, E1, E5 and E6. In Table 1.1, the characteristics of the Open Services (OS) offered in the E1 and E5 bands are described. E1 supports the OS, CS, SoL and PRS services. It contains three navigation signal components in the L1 band, and two components, E1-B and E1-C, are open-access signals with unencrypted ranging codes accessible to all users. E1-B is a data channel and E1-C a pilot channel. The E1-B data stream, at 125 bps of navigation data, also contains unencrypted integrity messages and encrypted commercial data. The MBOC modulation is used for the E1-B and E1-C signals, implemented by the Composite Binary Offset Carrier (CBOC). Also in this case the modulation scheme impact in the bandwidth therefore 4 MHz are needed centered in 1.57542 GHz. The E5 signal is sub-divided into signals denoted E5a and E5b. The E5a and E5b signal components are modulated onto a single E5 carrier frequency at 1.191795 GHz using a technique known as Alternate Binary Offset Carrier (AltBOC). The composite signal E5 can be processed as a single large-bandwidth signal or as two different signals. In the receiver implementation the impact of bandwidth is very important, an approach that includes both components (E5a + E5b) requires more than 50 MHz, while a separate treatment implies a bandwidth of 20.46 MHz for each component.

BeiDou

In phase III, the Beidou Navigation Satellite System provides global navigation coverage through 35 satellites which support open services SPS (Standard Accuracy Signal Service). In this case, and with the objective of non-interfering frequency band allocation, MBOC and AltBOC spreading modulation are used for the B1C and B2 signals, respectively. This directly impacts the minimum bandwidth necessary to receive the signals coming from the BeiDou constellation. The bandwidth required for the B1C signal, centred at 1.57542 GHz, is 32.736 MHz and for the B2 signal, centred at 1.17645/1.20714 GHz, is 20.46 MHz.

GLONASS

In contrast to the other constellations, each GLONASS satellite broadcasts at a particular frequency within the band. This frequency determines the frequency channel number of the satellite and allows receivers to identify the satellites (with the Frequency-division multiple access technique). The CDMA Open Service Navigation

Signal in L3 frequency band is called L3OC and consists of two BPSK(10) components: data and pilot. These components are in phase quadrature with each other, and L3OCd is delayed 90° [19].

The integration of new open signals in GNSS systems with global coverage offers new opportunities in the design of positioning systems. The trend shows a homogenization of medium access techniques (CDMA) and modulation schemes. There is also an increase in bandwidth requirements due to frequency relocation. The expansion of GNSS systems and signals promises excellent performance in the localization process based on an approach capable of using all available resources simultaneously.

1.2.2 Enhance GNSS

GNSS enhancement refers to techniques used to improve the accuracy of positioning information provided by the Global Positioning System or other global navigation satellite systems, generally a network of satellites used for navigation. These techniques include Real-time kinematic positioning, Carrier-phase tracking, or GNSS augmentation services. Enhancement methods of improving accuracy rely on external information being integrated into the calculation process. Many such systems are in place, and they are generally named or described based on how the GNSS sensor receives the information. Some systems transmit additional information about sources of error (such as clock drift, ephemeris, or ionospheric delay), others provide direct measurements of how much the signal was off in the past, while a third group provides additional navigational or vehicle information to be integrated into the calculation process. In this thesis, the PointPerfect GNSS correction service [20] is used.

Corrections of a global navigation satellite system (GNSS) improve navigation system attributes, such as accuracy, reliability and availability, by integrating information into the PVT calculation process. There are many such systems, and they are generally named or described according to the way in which external information is received by the GNSS sensor. Some systems transmit additional information on error sources (such as clock drift, ephemeris or ionospheric delay), others directly measure how much the signal has drifted in the past, while a third group provides additional vehicle information for integration into the calculation process.

Another technique to improve the reliability of the navigation is the multi-sensor integration. In 1.5, the EMERGE project [21] use case that integrates GNSS/INS information [22] is presented.

1.3 GNSS SDR-based solutions

1.3.1 GNSS SDR receiver

Nowadays, SDR development has earned the interest of the research community. In particular, in the last decades, SDR has become an RF platform capable of being reconfigured in real-time and a paradigm that can redefine modern communication systems. This is due to the variety of advantages SDR offers, like the reduced development costs with a single platform or the replacement of ad hoc systems with SDR devices controlled by software, which are easy to handle for updates compared with classic devices where physical replacements are required. Moreover, SDR devices offer the versatility to implement every specific task through a single device, the interoperability to develop multi-standard and multi-technology systems able to operate simultaneously [23, 24].

During this thesis, the SDR paradigm has been applied to different case studies: in the analysis of GNSS signals and in the implementation of other systems related to localization. Specifically, an SDR-based GNSS receiver can be defined as a generic GNSS receiver that has been designed and implemented using SDR [11]. Figure 1.7 (based on: [25] and [26]) depicted a generic GNSS receiver.

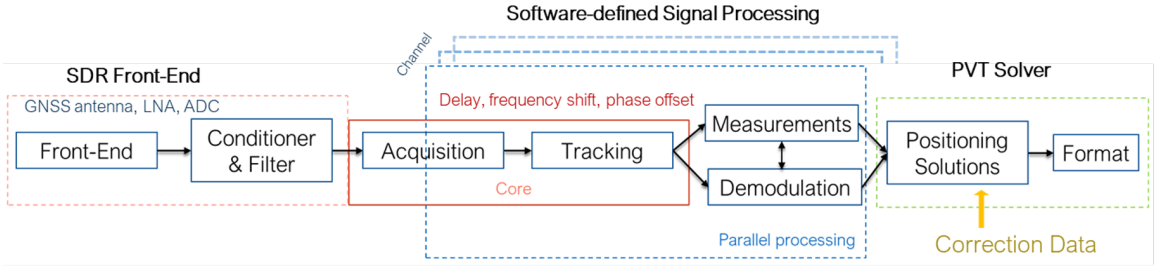

Figure 1.7: Block diagram of a generic GNSS receiver architecture and its functional blocks

The Front-End, consist in an antenna and an SDR device that implements the down-conversion process directly in base-band (BB) or intermediate frequency (IF) in two components [2]: in-phase and quadrature. Subsequently, both components are digitally converted, filtered and sent to the signal processing block. The signal processing block is the core of the system. The core comprises the GNSS signal acquisition and

[2]Many SDR systems present a Homodyne architecture (Zero-IF), but in fact, the flexibility of these systems allows us to configure the HW to obtain intermediate frequencies to be processed in digital processing subsequently. In the case of GNSS signals, the term IF indicates that each satellite's signal must be subjected to a further carrier search and synchronisation process.

tracking block and can be performed in parallel. The acquisition process is implemented to identify the available satellites and a first estimation of the frequency and phase of the carrier [27, 28]. After verification of positive acquisitions, the tracking block follows the evolution of the signal carrier frequency and the Pseudorandom Noise (PRN) code phase. When the detected signals are correctly tracked, it moves on to the demodulation of the navigation information and the measurement process which results in the position calculation. Finally, the complete process results are sent to the PVT [3] solver for position calculation. However, the GNSS receiver scheme can interact with other localization techniques or Augmentation systems, i.e. correction from dedicated systems. The location and navigation data provided by other technologies can be added directly into the PVT solver block or at a subsequent stage, from the receiver output.

The function of the acquisition block is to determine visible satellites and coarse values of carrier frequency and code phase of the satellite signals. The satellite identification process is performed by the autocorrelation process. Specific codes have high autocorrelation and low crosscorrelation values, so they can be identified in noisy environments (see Figure 1.8). To obtain a high autocorrelation value, aligning the locally generated code with the received code is necessary. In fact, an essential element is the code phase, which is also used in the distance measurements between the satellite and the receiver. The other element is the carrier frequency, which corresponds to the Intermediate Frequency (IF) in case of downconversion. The IF should be known from the L1 carrier frequency of 1.57542 GHz and the mixers in the downconverter. However, the frequency can deviate from the expected value. The relative velocity of the satellite causes a Doppler effect, resulting in a higher or lower frequency. In the worst case, the frequency can deviate up to ± 5 kHz (stationary receiver).

The method used in the acquisition process is Parallel Code Phase Search Acquisition. The idea is to perform a correlation with the incoming signal and a locally generated code in the frequency domain. The in-phase and quadrature digital components from the front-end (SDR platform) are multiplied by a locally generated carrier. The result corresponds to the C/A component when compensating for the Doppler effect suffered by the signal and other errors introduced in the downconversion process. 41 iterations with 250 KHz steps are performed to sweep the frequency range. Each carrier multiplication process is converted in the frequency domain using the Fast Fourier Transform and multiplied by the conjugate of the local PRN code. This

[3]PVT is refer to the user Position, Velocity, and Time. The role of a PVT solver block is to calculate navigation solutions and deliver information in adequate formats or data representation.

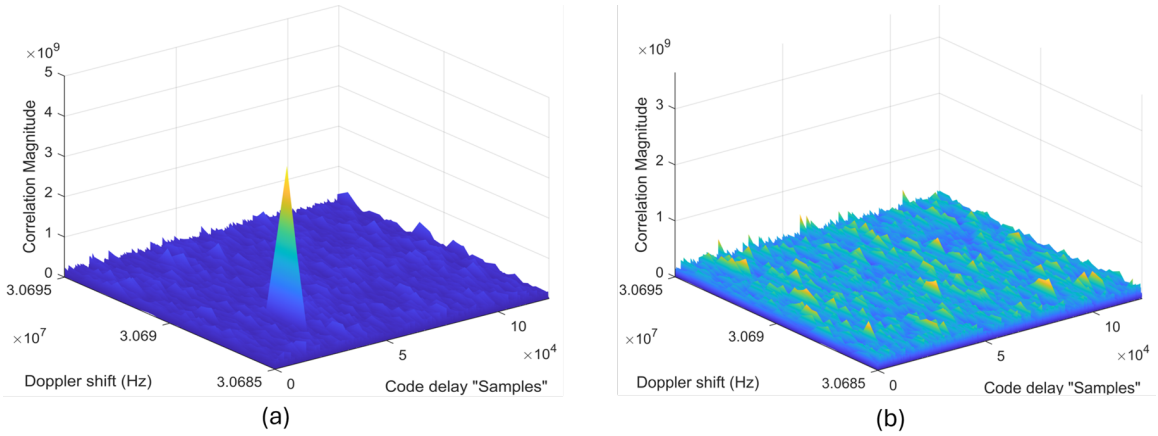

Figure 1.8: Output from acquisition block. (a) A PRN code is visible so a significant peak is present. (b) No PRN code detected

multiplication process in the frequency domain corresponds to the circular crosscorrelation operation in the time domain. The multiplication result is transformed into the time domain by an inverse Fourier transform. The absolute value of the output of the inverse Fourier transform represents the correlation between the input and the PRN code. If a peak is present in the correlation, the index of this peak marks the PRN code phase of the incoming signal. This algorithm is depicted in Figure 1.9 (a).

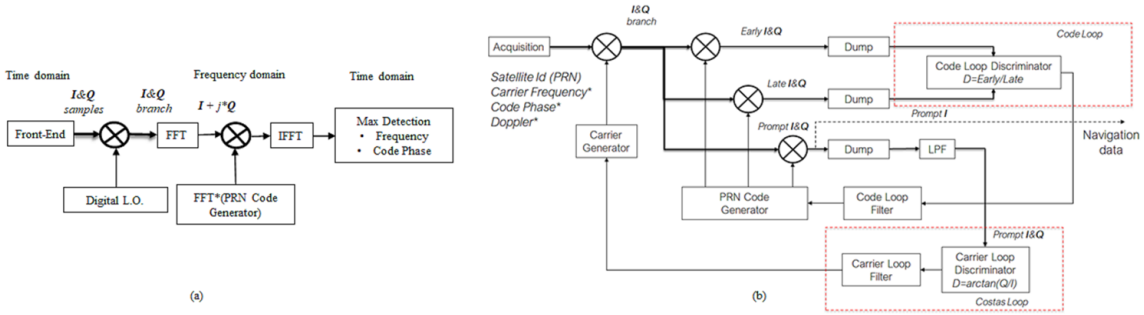

Figure 1.9: Core of the GNSS receiver. (a) Parallel Code Phase Search Acquisition diagram. (b) Tracking block schematics

The Parallel Code Phase Search Acquisition algorithm has the better performances in term of processing time and number of iterations. Although it presents a major complexity due to the transformations between the time and frequency domain (FFT and IFFT). However, many mathematical tools have addressed the complexity problem in the FFT transform. That is why our proposal initially proposes the use of Matlab in the elaboration of digital signals. The acquisition provides only rough estimates of the frequency and code phase parameters. The main purpose of tracking

is to refine these values, follow the signal evolution, and demodulate the navigation data from the specific satellite.

The first step in the tracking process is to correlate the signal with a locally generated C/A code from the given satellite. This correlation removes the C/A code and produces the navigation message signal. However, the samples to be correlated must be consistent with the phase calculated in the acquisition process. Frequently, due to disturbances, the phase can shift in a few places either to the right or to the left. This phase shift must, therefore, be compensated for in the code. Two additional PRN codes are introduced to determine the direction of the shift. The original PRN code is called prompt and is the one we try to keep aligned, while the new codes are called early code and late code, shifted to the right and left, respectively. In our proposal, the code tracking process is performed for each millisecond, and although it may appear inefficient, it actually unifies the working frequency of the entire block and reduces the complexity of the process. Figure 1.10 shows how the signal is followed in terms of code and frequency.

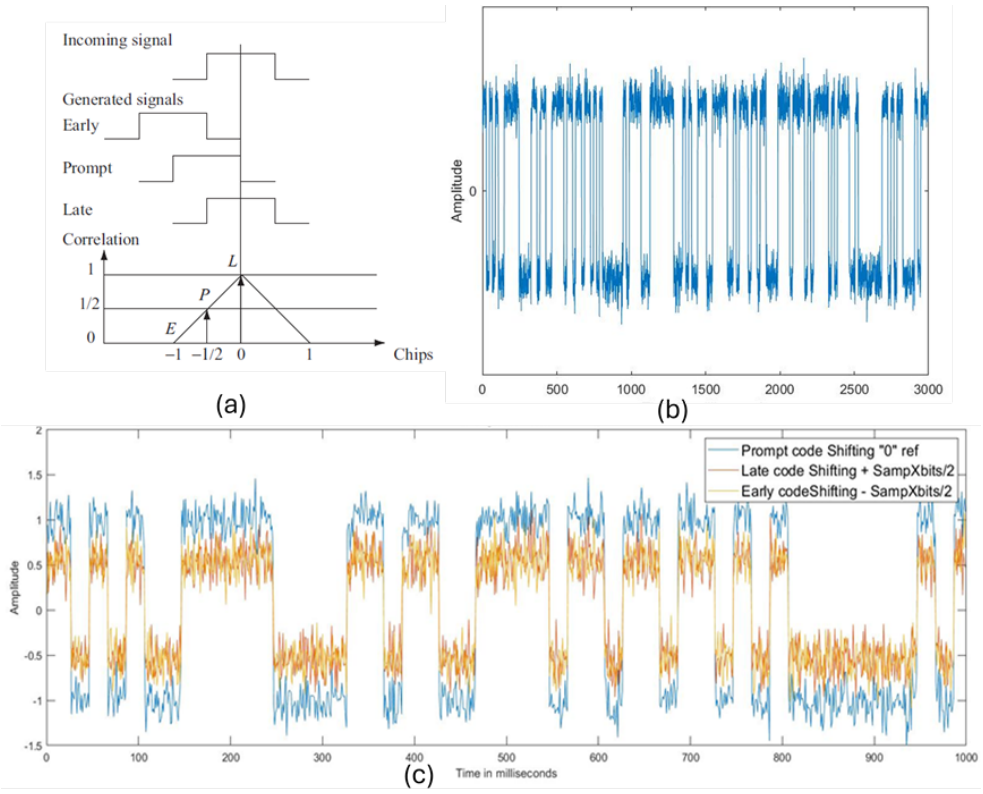

Figure 1.10: Code tracking. (a) Three local codes are generated and correlated with the incoming signal: Early, Prompt, and Late. (b) A portion (in the time domain) of the tracked navigation bits. (c) Code tracking result for each code: Early, Prompt, and Late

The variation of the reference velocity between the transmitter and receiver causes the Doppler effect to be inconsistent. In addition, SDR platforms can exhibit instability in the local oscillator. These phenomena support the use of a carrier tracking loop. The carrier loop discriminator block finds the phase error on the local carrier wave replica. The output of the discriminator is then filtered and used as feedback to the Carrier Generator, which adjusts the frequency of the local carrier wave. In this way, the local carrier wave could be an almost precise replica of the input signal. The problem with using an ordinary Phase Lock Loop (PLL) is that it is sensitive to 180° phase shifts. Therefore, our proposal consists of a Costas Loop with a tangential discriminator, insensitive for 180° phase shifts due to navigation bits.

The role of a PVT Solver block is to compute navigation solutions and deliver information in suitable formats for further processing or data representation. The position estimation process is far from simple because all the errors that occur in the system cannot be modelled. Therefore, integrating techniques that contribute to the system's accuracy is necessary. This is where GNSS augmentation systems play a key role. Augmentation of a global navigation satellite system (GNSS) improves navigation system attributes, such as accuracy, reliability and availability, by integrating information into the PVT calculation process. In this work, we developed localization [22] applications that use real-time corrections and exploit the diversity in geometry (multi-constellations) and in frequency (multi-frequency). The next section describes another GNSS SDR-based application: the emulation and simulation of GPS and GALILEO signals.

1.3.2 SDR-based GNSS simulator

When developing and studying the signal processing components of GNSS applications, it is crucial to have access to representative data for testing their functionality and assessing the primary physical phenomena that impact system performance. The objective is to implement a system capable of real-time operation using data captured from a GNSS antenna via an RF front end and an ADC. However, during the solution development phase, relying on real sampled data is not optimal. The primary reason for this is the inherent difficulty in controlling the properties of the entire system and the characteristics of the sampled GNSS signals. Not only is it challenging to manage the received signals, but it is also difficult to fully ascertain their properties.

An additional factor supporting the development of SDR-based GNSS simulators is the ease with which digital signals can be converted to the analogue domain. The inherent flexibility of SDR and the simplicity of its configuration make SDR-based

platforms highly advantageous for GNSS applications, particularly in baseband processing, which can then be seamlessly integrated with the front end provided by an SDR platform.

This section introduces the simulation of GPS and GALILEO signals for Software-in-the-Loop (SiL) and Hardware-in-the-loop (HiL) applications. The Key parameters for these simulations include Doppler frequency, which accounts for the relative motion between the satellite and receiver, impacting the observed frequency of signals. Intermediate frequency (IF) is another critical parameter, as it represents the frequency to which a GNSS signal is down-converted after being received by the antenna, making signal processing more manageable. The carrier-to-noise ratio (C/N0) is also essential, as it measures signal strength relative to background noise, indicating signal quality under various conditions. Additionally, frequency bands differ between systems, with GPS primarily using the L1, L2, and L5 bands, while Galileo utilizes the E1, E5, and E6 bands. This selection ensures the correct signal environment is simulated for each constellation. The constellation itself, referring to the arrangement and number of satellites in orbit, is crucial for evaluating receiver performance in terms of satellite visibility and geometry.

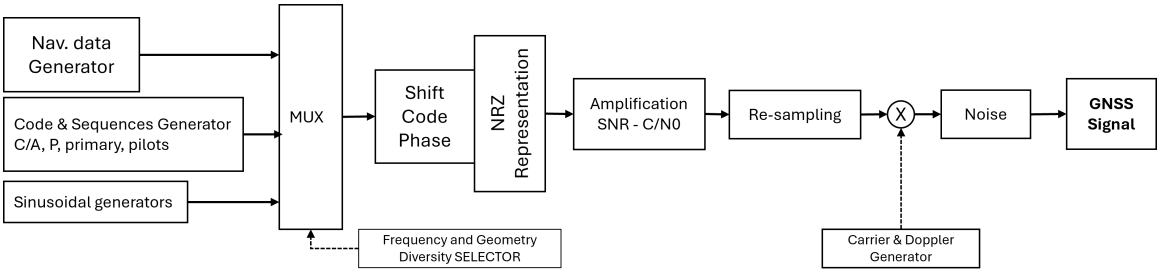

Figure 1.11: Simulator for GPS and GALILEO signals

Figure 1.11 shows the diagram model of the GNSS signal simulator for GPS and GALILEO signal generation. The scheme can be divided into three main parts: the first part is dedicated to the numerical data and sequence generation, the second part emulates the code delay and the amplitude of the signal, and the last blocks are dedicated to the frequency shift due to the Doppler effect and L.O. errors.

The first part consists of the generators of C/A, P-code, BOC signal, and random (or not) bits of navigation messages. This is the left part of the scheme. It makes all three navigation signals at the baseband: data channel, pilot channel, and restricted access channel. For example, in the case of the generation of the GPS L1 signal, the C/A code was primarily thought for the acquisition of the P (or Y) code (See Figure 1.12). The navigation message consists of 30-second frames that are 1,500 bits long

and divided into five 6-second sections. For the GALILEO signal, we apply BOC modulation using a different PRN code, and this allows the spectrum sharing in the L1 band.

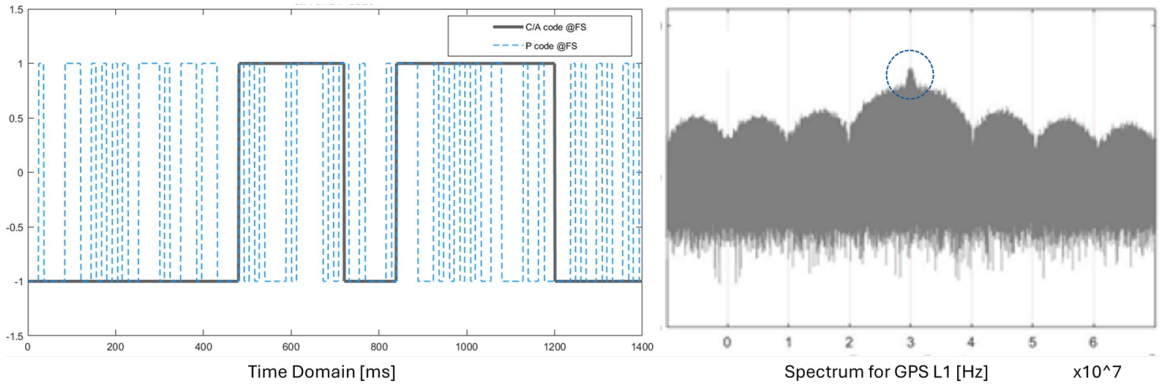

Figure 1.12: C/A and P GPS Codes. Spectrum for generated GPS L1 signal

There are some differences between the GPS and GALILEO constellations regarding the codes used. In terms of the length (in time) of the pseudo-random codes in GALILEO, the codes have a temporal length of 4 milliseconds, which presents some benefits in terms of the correlation peak result (See Figure 1.13). GPS and Galileo differ in their signal types and spreading codes. GPS provides one public signal and one encrypted signal, while Galileo offers three signals: two public Open Service (OS) signals and one encrypted Public Regulated Service (PRS) signal, with only the OS signals considered here. The data channel in Galileo's OS signals contains navigation data, while the pilot channel carries a secondary code sequence. GPS uses a 1023-chip spreading code, whereas Galileo employs a longer 4096-chip code, both with a chipping rate of 1.023 MHz. Additionally, Galileo codes on the L1 band are combined with a subcarrier signal, enhancing signal tracking performance. While GPS signals are bandwidth-limited to 20 MHz, Galileo's L1 signals are limited to 40 MHz.

Regarding data structure, the Galileo system will use a superframe, frame, and subframe construction similar to GPS. Subframes will have a unique word to facilitate synchronization to the start of the subframe (similar to the preamble in GPS). The unique synchronization word is followed by the data part, a checksum field, and tail bits. It is expected that the construction of the data part will be different from that of the GPS messages. The satellite orbit parameters have the same field size and scale in both systems. The time parameters have different field size and scales (except clock correction coefficients in the almanac). Synchronization Word (Preamble) GPS is using an 8-bit (symbol) pattern. Galileo is likely to use a 10-symbol pattern. Error

Detection Galileo will use cyclic redundancy check (CRC) to detect data corruptions inside subframes. Channel Coding In addition to CRC, Galileo will use forward error correction (FEC) to detect data corruption and correct corruption to a certain extent. This will facilitate the correction of a much larger amount of corruption compared to GPS, where only one bit per subframe can be corrected. Block interleaving will be used to make the Galileo data even more corruption-resistant.

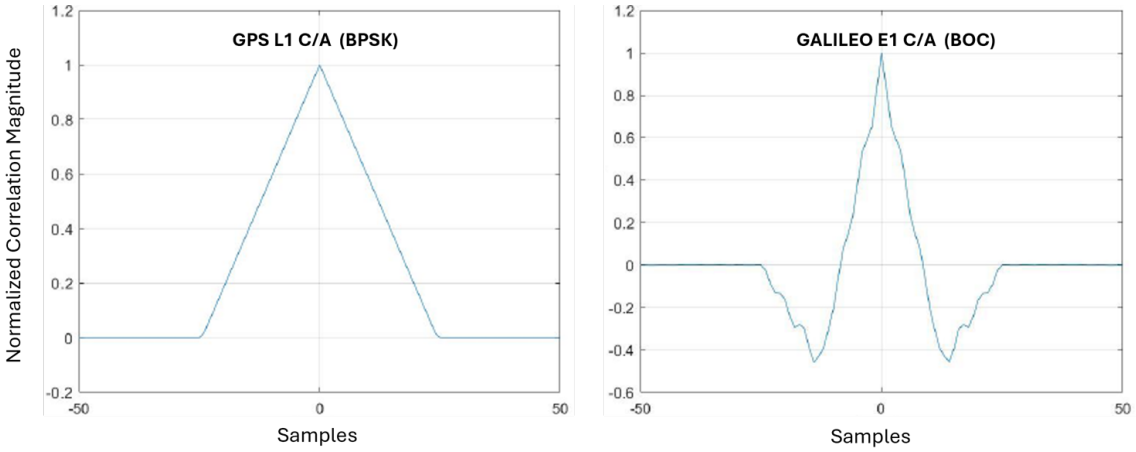

Figure 1.13: Normalized absolute correlation function of GPS L1 C/A (BPSK) and GALILEO E1 (BOC)

While both GPS and Galileo share fundamental principles in signal processing, significant differences exist that necessitate careful consideration in receiver design and implementation. Galileo's Open Service signals require a broader bandwidth than GPS, which may require updates to existing GPS front ends to ensure optimal signal reception. The longer spreading codes used by Galileo also lead to extended correlation times during signal acquisition and tracking, introducing additional complexity, particularly with the BOC modulation that is characteristic of Galileo. Furthermore, the signal processing chain in Galileo involves more sophisticated techniques for PRN code generation and handling, requiring careful attention to avoid issues like false locks. Data demodulation and decoding in Galileo are more intricate due to the specific error correction and data structures employed, which differ from those in GPS. Despite these complexities, the position computation process remains largely similar, with only minor adjustments needed to account for differences in coordinate systems. These variations underscore the need for tailored approaches in developing SDR-based receivers that can effectively handle the characteristics of both GPS and Galileo signals.

The development of an SDR-based GNSS simulator during this research becomes crucial in light of the aforementioned differences between GPS and Galileo signal processing. Such a simulator allows analysis and models of the main errors and variations, providing a controlled environment where different signal characteristics, such as bandwidth, spreading codes, and modulation techniques, can be replicated and studied. This level of control is invaluable for testing and validating new GNSS algorithms, especially in the context of multi-constellation systems where interoperability and performance optimization are critical. By leveraging the flexibility and adaptability of SDR technology, GNSS simulators can simulate real-world conditions with high precision, facilitating the development of robust localization solutions that can effectively operate across both GPS and Galileo systems. Consequently, an SDR-based GNSS simulator not only enhances our understanding of GNSS signal processing but also plays a pivotal role in advancing the field of localization by enabling the design and testing of next-generation positioning systems.

1.4 SDR-based integrated sensing and localization

In a localization system, we distinguish two types of devices: anchors and targets. The anchors refer to nodes with a fixed known location, while the latter refers to nodes whose position is yet to be determined [29]. An anchor can be a base station in a cellular network, an access point in a local area network, or, in some application scenarios, a node located via GNSS. A target, instead, can be any other device equipped with a receiver. The objective of a positioning algorithm is to estimate the unknown target locations. If a target node can communicate and acquire information only from the anchors, then the positioning method is considered non-cooperative. In turn, if all nodes can communicate and exchange information with each other, then such a system is referred to as cooperative [30].

Non-cooperative and cooperative positioning can be either distributed (self-positioning) or centralised (network-centralised positioning). The advantages of distributed methods are essentially scalability and low complexity. However, they may not achieve optimality in a global sense [31], are sensitive to error propagation due to imperfect information exchange and may require a long convergence time [32]. In contrast, centralised methods are fundamentally optimal and stable, but the computational complexity can grow with the number of nodes. Therefore, centralised

or distributed usually depends on the application and application scenario and the trade-off between complexity and performance.

Accuracy is one of the main challenges that has motivated research into new positioning systems. The inherent uncertainty in localization is due to the small number of nodes with known locations (anchors), the large number of nodes with unknown locations (targets), their limited connectivity and the difficulty of modelling the radio propagation channel. Positioning is done based on wireless distance, angle or power profile measurements. For example, the angle between a transmitter and a receiver can be estimated from the angle of arrival (AoA) of a signal, the distance between two nodes can be obtained from received power (RSS, link quality) or time-of-flight estimates, and the power profile from measurements of the impact of the communications channel. Under radio propagation conditions [33], distance, angle and power profile measurements are affected by errors which, in the case of distance and angle-angle, manifest themselves in the form of noise and bias of unknown statistics.

In this section, we explore two applications of Software-Defined Radio technology with a focus on the localization process. The first subsection details the development of a comprehensive localization system utilizing an SDR-based Automatic Dependent Surveillance-Broadcast (ADS-B) receiver. Here, the goal is to accurately determine position by capturing real-time status information, such as location and speed, broadcasted by aircraft via secondary surveillance radar (SSR). The second subsection presents the outcomes of a radio monitoring and localization service, which leverages a distributed network architecture powered by SDR technology.

1.4.1 SDR-based localization using ADS-B signals

From 2020, the availability of aircraft with ADS-B Mode S compliant transponders is almost 100% according to Regulation (EU) No. 1207/2011. The ADS-B technology can be used to localise and track devices that can receive the aircraft's transmitted signal. This localization approach is opportunistic because it uses RF signals and information available in the ether but is not initially designed for the receiver's localization purpose. The algorithms proposed in this scenario are mainly based on multilateration (MLAT) and use time difference measurements (TDOA) [34], time of arrival (TOA), or distance measurements from RSS Vs. Distance models [35, 36].

This section introduces the challenge of determining receiver position from the opportune decoding of ADS-B signals using SDR. The MLAT algorithm uses the distance determined from the RSS-based measurements of the received signal. This approach is based on an experimental-achieve RSS-Distance model.

The idea of determining the spatial coordinates of an object using information from the received ADS-B signal is addressed in three fundamental steps. The first step is to use power measurements of the received signal to generate a model to determine the distance between the transmitter (aircraft) and the receiver. The next phase consists of solving a system of equations, one equation for each aircraft available in a given time window, using MLAT algorithms. The last step is the development of the SDR-based receiver to detect, decode and extract radio signal information of ADS-B messages.

RSS Channel Model

The most widespread model used to describe radio signal propagation is the Log-Normal Shadowing model, a generalisation of the Friis free space equation. Equation (1.1) shows the simplicity of the model, where RSS_0 is a constant term which takes one of the three possible values of the transmission power of airborne transponder, d is the distance between the transmitter and receiver, β is the path loss exponent, η is a zero-mean Gaussian random variable, and N indicates the number of messages considered for average calculation.

$$RSS(dBm) = RSS_O(dBm) - 10\beta log(d) + \eta \tag{1.1}$$

$$\hat{d} = 10^{\frac{RSS_O(dBm) - RSS(dBm)}{10\beta}} \tag{1.2}$$

$$RSS(dBm) = \frac{1}{N}\sum_{n=1}^{N} RSS_{n,t} \tag{1.3}$$

The channel model assumes that the noise has a Gaussian distribution. When a variable has a Gaussian distribution, its mean value equals its average value. However, in practical conditions, where some outliers may exist, it is better to use the mean value because it is more robust to outliers. In Equation (1.2), the value of $\hat{d}$ is estimated from the mean RSS measured on a set of messages from the same airborne in a given time window.

Multilateration algorithm

The MLAT is based on the solution of a system of equations where the solution corresponds to the receiver coordinate. In our case, the estimated distance between the transmitter and receiver is determined based on a model relating to the RSS measured

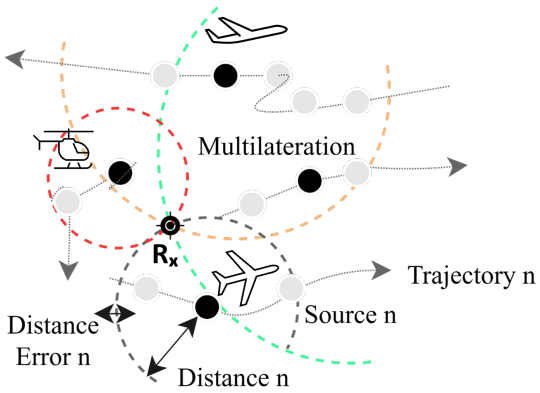

Figure 1.14: MLAT in the ADS-B scenario

at the receiver. The application of a statistical model leads to errors in distance estimation. These errors in distance estimation lead to the use of optimisation methods in selecting the correct solution.

Given the n reference points $P1(X_1, Y_1, Z_1)$, $P2(X_2, Y_2, Z_2), ...Pn(X_n, Y_n, Z_n)$ and the range measurements $d_1, d_2, ..., d_n$. The solution of the coordinates (x, y, z) using n points is equivalent to solving a quadratic system of equations with $n-th$ expression:

$$(x - X_n)^2 + (y - Y_n)^2 + (y - Y_n)^2 = d_n^2 \tag{1.4}$$

Manipulating (1.4) we can write:

$$(x^2 + y^2 + z^2) - (2xX_n + 2yY_n + 2zZ_n) = d_n^2 - (x_n^2 + y_n^2 + z_n^2) \tag{1.5}$$

Equation (1.5) allows to compose a linear system equation with the form $\mathbf{A}x = \mathbf{b}$. Applying Least Squares (LS) methods with the constraint $x \in \{(X_0, X_1, X_2, X_3)^T \in \mathbb{R}^4 / X_0 = X_1^2 + X_2^2 + X_3^2\}$, the solution can be written as follows: $\hat{x} = (A^T A)^{-1} A^T b$. However, solutions offered by this method depend only on the matrix $\mathbf{A}$ and the vector $\mathbf{b}$, which depends only on the coordinate values received in the ADS-B messages that are opportunely decoded by the receiver. Finally, the selected solution is the one that minimises the sum of squared distance error.

SDR-based ADS-B receiver

There are two types of ADS-B downlink signals (centred at 1090 MHz) in mode S, the short response (56 bits) and the extended response (112 bits), which correspond to the short and long interrogations of the secondary surveillance radar (SSR). The bit duration is 1 microsecond and uses Pulse-Position Modulation (PPM). All Mode S responses start with a fixed preamble of 8 symbols (duration 8 microseconds) and

continue with a long or short bit sequence (payload). Civil Aeronautics ADS-B messages after the 8-microsecond preamble continues with the binary sequence 10001, corresponding to the Downlink Format (DF) field. The next three bits correspond to the Transponder Capability field, followed by the ICAO code, which identifies the aircraft and is 24 bits long. The short messages continue sequentially with the 24-bit Parity/Interrogator (PI) field, completing the 56 bits. However, before concluding with the PI, the long messages contain a block message (ME) that contains information concerning the aircraft's position, altitude, speed, heading and status. In particular, the extended messages are helpful for localization because a stream of information regarding the aircraft's position/speed/altitude is available. Since the distance estimation method uses power-averaged values, a more accurate value will depend on analysing a significant number of messages from the same aircraft. It is essential to detect as many messages as possible to reduce the analysis time window. Theoretically, the frequency of extended position and velocity messages is 2 Hz, and the total number of messages (on average) exceeds six messages per second.

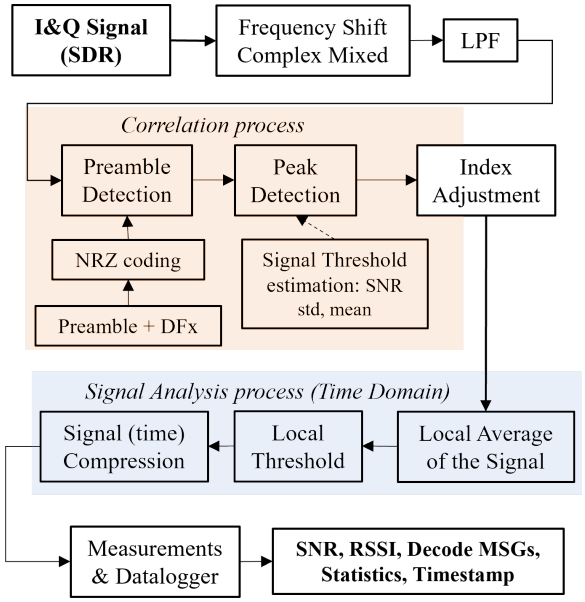

Figure 1.15: SDR-based ADS-B receiver

The received data is split into In-phase and Quadrature digital components (I&Q) when using SDR platforms with Zero-IF architectures as Front-End. Although the frequency is centred at 1090 MHz, frequency shifts may be experienced due to errors in the SDR device's local oscillator (LO) [11], the airborne transponder or the Doppler effect introduced by the speed difference between transmitter and receiver. Therefore, the first blocks of our receiver are dedicated to the correct centring of the band,

the elimination of the DC component and the filtering of the signal to increase the Signal to Noise Ratio (SNR). The receiver processes the baseband samples to detect preambles that indicate the start of an ADS-B message. The correlation process uses pre-coded NRZ sequences to obtain a higher sensitivity in preamble detection [37].

Detecting a maximum exceeding a specific threshold means, with a high probability, the presence of an ADS-B message. The threshold is calculated based on living metrics, where the (time) segment or signal frame to be analysed is statistically evaluated.

Figure 1.15 highlights the blocks working in the frequency and time domains. Once the preambles are identified, their position in the time domain is determined, and false positives are discharged. After identifying the message positions of the messages in the time domain, a signal analysis is performed to validate the results obtained in the frequency domain. The validation also confirms that the preamble's "high" pulses are within 3dB of the reference power. The parser block (*MSGs Decode*) decodes the information. At the end, the data is stored in a structure that includes the following fields: RSS (before filtering and after filtering), decoded message information (Speed, Position, Altitude), internal buffer index, timestamp, SNR and the identification of the aircraft through the ICAO code.

1.4.2 SDR-based radio monitoring and support services

In this SDR-based application, the system is a synchronised network dedicated to spectral monitoring using SDR. One of the applications developed in this research is the detection of interfering sources, which is also applied to the case of the detection of an emitter. In this context, the system acquires the signals in the spectrum under analysis, and the samples are encapsulated and transferred to the centralised system controller. The controller works under the assumption of RF synchronization between the devices; considering that the level of synchronization depends on requirements needed for the specific service. The first analysis executed by the controller is spectrum monitoring, where the Power Spectrum (PS) is processed for every sample frame acquired by each SDR device; the maximum, average, and minimum PS are computed for each SDR device with multiple acquisitions. This first stage enables the analysis of the spectrum status to identify the signals that are always present through the minimum PS; the maximum PS allows the identification of all the frequency bands, even the sporadic ones, that are present in the maximum PS but not in the minimum. Then, the average PS can be an indication of how much a signal is present in the acquired frames: if it is closer to the minimum PS, it can be a sporadic

signal, while if it is closer to the maximum, the occurrences are more frequent, as long as its amplitude has low variability.

The detection algorithm successfully identifies the interfering signals in the selected spectrum, based on the z-score. In general, for a data set x, its z-score is computed as:

$$z = \frac{x - \mu}{\sigma} \tag{1.6}$$

where μ is the data set mean value and σ its standard deviation. The z-score standardizes the data set distribution making it with zero mean and standard deviation equal to 1. Taking advantage of equation 1.6, the prototype detection algorithm evaluates the differences between the actual PS with the average one so that every new signal will be detected through the comparison of its z-score

$$z^* = \frac{|PS - PS_{avg}| - \mu}{\sigma} \tag{1.7}$$

with a threshold λ [38]. Then, the threshold is used as a comparison enabling two possible situations:

$$\begin{cases} H_0: & |z^*| < \lambda \\ H_1: & |z^*| \geq \lambda \end{cases} \tag{1.8}$$

If the absolute value of the z-score is equal or greater than λ (i.e., case H_1), it corresponds to a point distant at least λ standard deviations from the mean value, and a signal has been detected in that frequency; otherwise there is no detection (H_0). A trade-off is required to reduce the false alarm rate; it can be done by choosing a higher λ, with the drawback of a higher miss rate in the detection. In this step, the focus is on narrowband signals since the detection algorithm based on the z-score cannot guarantee the detection of wideband signals since a wideband signal will affect the PS mean, reducing its effect on the z-score. The detection of wideband signals will be taken into account in further development.

The detected signals of each SDR device are compared so that only the merged detection is taken into account, focusing only on those signals detected by all the devices simultaneously. This operation can be used to localise a static signal source. This prototype has been considered a range-based localization algorithm to estimate the distances between the source and the receivers; in general, localization is performed in two steps: the distance calculation and the position solution. The main results are presented in C3.

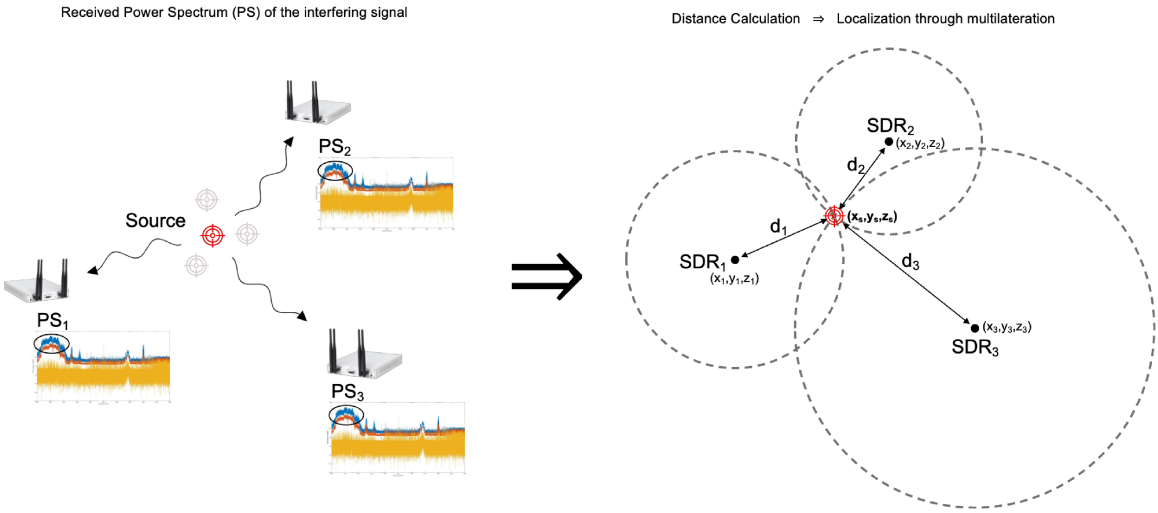

Figure 1.16: Default service representation: from spectral analysis with interference detection to position solution through multilateration (MLAT)

1.5 Multi-constellation and multi-sensor solution for reliable positioning. The EMERGE project use case

An important part of this research has focused on industrial navigation solutions using a multi-constellation and multi-sensor approach. The case study presented in this section is part of a project that has led the way in terms of the automotive paradigm in Italy. The EMERGE project (Connected, geo-localized and cyber-secure vehicles) is co-funded by the European Union and spearheaded by RadioLabs in collaboration with esteemed partners such as the University of L'Aquila, Telespazio, Leonardo, and Elital. The project aims to design and develop a novel Onboard Unit (OBU) to provide dual-use capabilities to commercial vehicles (i.e., last-mile delivery and emergency operations) enabled by dedicated cloud-edge services. The OBU is equipped with vehicle-to-everything (V2X) connectivity, procedures for cyber-secure operations and an accurate geo-localization platform. The role of this latter is to guarantee real-time, precise positioning of all nodes and their relative positions, accomplished through innovative fusion techniques that leverage data from multi-constellation GNSS sensors and inertial sensors.

1.5.1 On-Board Unit implementation

This section presents the proposed architecture for implementing the EMERGE On-board Unit (OBU), with an emphasis on creating a versatile and easily configurable system. The architecture is designed to be compatible with various hardware platforms and operating systems while ensuring efficiency, performance, portability, and scalability. The challenge is to establish a level of abstraction that meets these critical functionalities, ensuring that the system can take advantage of existing HW and SW components (COTS), operate effectively in real-time or post-processing modes, and easily add or remove sensors, and algorithms as needed. The architecture is organized

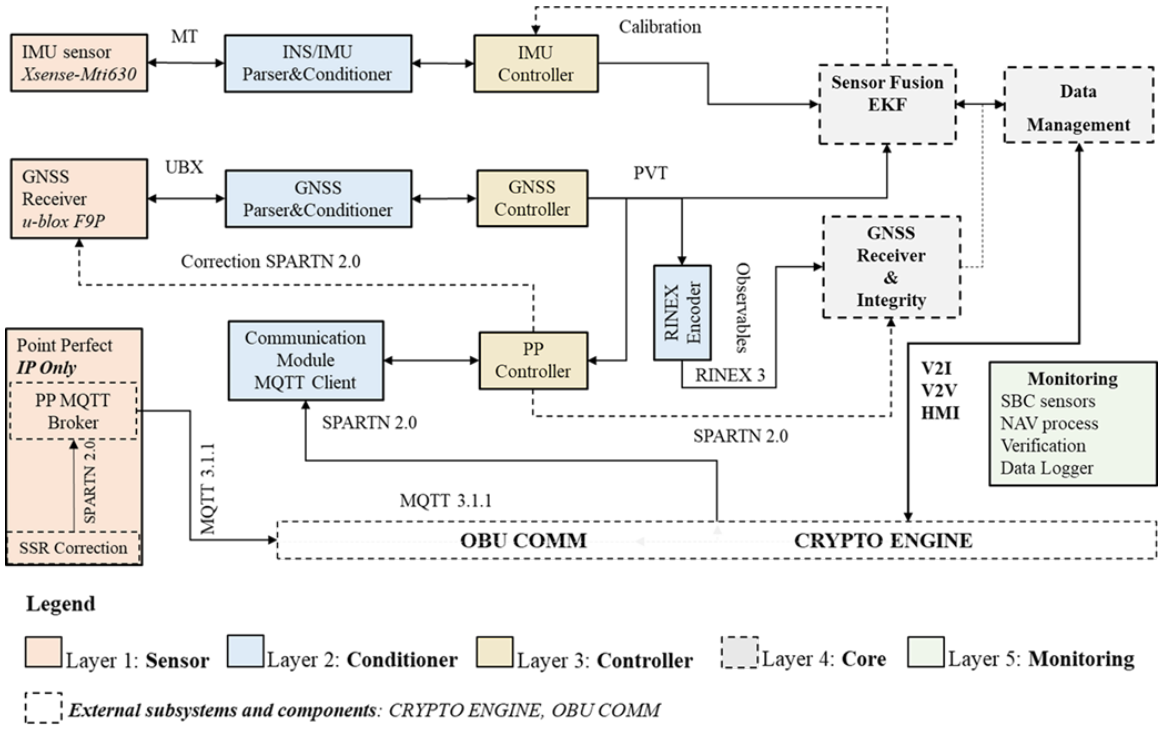

Figure 1.17: On-board system architecture

into five functional layers (See Figure 1.17), each playing a specific role in the navigation system. The Sensor layer comprises all sensors that provide critical information, such as GNSS receivers, inertial sensors, and external clocks. The Parser and Conditioner layer is responsible for decoding, parsing, and filtering sensor data, as well as handling communication with the sensors. The Controller layer manages system start-up, interruptions, and parameter adjustments, particularly in dynamic scenarios, while also overseeing the information flow between the Core and Sensor layers. The Core layer is the system's central hub, housing all navigation and information management algorithms, including Sensor Fusion and Integrity Software components.

Finally, the Monitoring layer tracks the system's performance during navigation, manages hardware resources, and supports remote accessibility and alert notifications. In addition to these layers, the OBU includes external subsystems like the COMM and CRYPTO ENGINE blocks, which ensure secure remote communication and access to IP-based services. This connectivity is crucial for receiving information from Augmentation GNSS (AGNSS) systems and exchanging data with traffic control elements such as control centres, other vehicles, or early warning systems. The system's architecture supports a loosely coupled integration of inertial sensor data with corrected GNSS navigation solutions, enhancing accuracy and integrity in navigation processes.

For the EMERGE project, the onboard navigation system's hardware, software, and service components are selected based on specific functional requirements. The primary components include a GNSS receiver, the u-blox ZED-F9P, which provides multi-band GNSS with centimetre-level accuracy, and an inertial sensor, the Xsens-MTi-630 AHRS, known for its high-speed dead-reckoning capabilities and low in-run bias stability. These sensors work together with a GNSS augmentation service, Point-Perfect, which delivers high-precision positioning corrections through an IP connection, ensuring efficient bandwidth use and reduced power consumption. The integration of these components into the OBU allows for a robust, adaptable navigation system capable of meeting the diverse demands of the EMERGE project.

1.5.2 Sensor Fusion: Loosely Coupled Integration

The primary obstacle in enabling AV navigation is establishing a consistently accurate and dependable navigation solution across all landscapes. GNSS stands out as the most prevalent source of navigation solutions due to its enduring stability in offering long-term solutions [39]. However, GNSS encounters limitations in providing accurate or navigation solutions in certain circumstances, notably within urban settings that include tunnels, underground parking areas, and high-rise buildings [40]. Satellite navigation in the urban environment is subject to some objectively critical issues, mainly related to:

- Availability/Visibility of satellites: the number of satellites available/visible, which allows obtaining a precise and punctual position calculation, is scarce or even absent;

- Multipath: the typicality and singularity of the urban context, in which the vehicles carry out their service, produces a non-trivial multipath error that seriously compromises the performance of the GNSS signal;

- Other kinds of interference that can have a significant impact in an urban context.

These criticalities result in considerable variability in the reception of the GNSS signal, ranging from optimal conditions to complete absence. On the other hand, inertial navigation systems (INS) possess several advantages. They operate continuously, exhibit low short-term noise, and resist jamming and interference. However, INS accuracy degrades over time due to integration errors, and maintaining effective sole-means navigation is costly. While GNSS ensures high long-term accuracy, its output rate is lower, and signal obstruction leads to discontinuous navigation. By integrating INS and GNSS, their advantages complement each other. GNSS measurements prevent inertial solution drift, while INS smoothes GNSS output and bridges signal outages. This integration results in a continuous, high-bandwidth navigation solution with long and short-term accuracy [41].

The primary advantage of a loosely coupled integration architecture lies in its simplicity. This architecture is versatile and compatible with any INS and GNSS user equipment, making it ideal for retrofit applications. In a loosely coupled INS/GNSS system, the integration algorithm utilizes GNSS position and velocity solutions as measurement inputs, regardless of the specific type of INS correction or GNSS aiding employed.

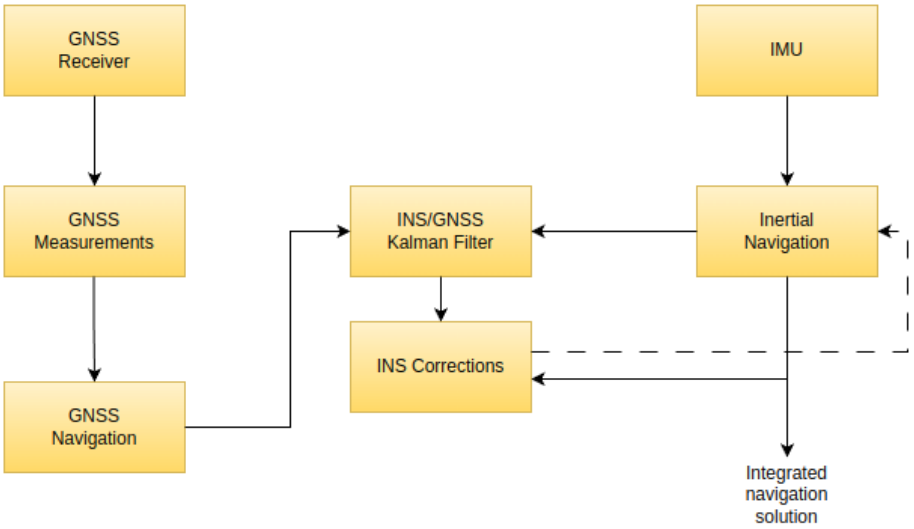

Figure 1.18: Loosely coupled INS/GNSS coupling scheme

In the cascaded operation of this architecture, the GNSS user equipment, which incorporates a navigation filter [41], utilizes the Extended Kalman Filter (EKF) as the

integration solution. The EKF is a recursive filter specifically designed to estimate the state of a dynamic system from multiple noisy measurements. It has evolved from the Standard Kalman Filter to address the complexities associated with nonlinear dynamic systems. The resulting integrated navigation solution consists of the INS navigation solution refined by the Kalman filter's error estimates. Specifically, within the loosely coupled integration architecture, the sequential processes can be delineated as follows:

1. Calculation of position and velocity with GNSS;
2. Calculation of the difference between the estimated position and velocity from the GNSS and INS solutions to assess IMU errors by integrating the estimated differences in a Kalman filter;
3. Correction of the INS solution using variational equations.

The tests presented in J1 were carried out in a confined area and, therefore, to simplify the spatial representation in a local context, the North East Down (NED) coordinate frame was chosen.
Figure 1.18 illustrates the functional diagram of the GNSS-INS loose coupling process. It is a closed-loop correction architecture; consequently, the estimated errors in the NED reference frame, $\delta\hat{\mathbf{L}}_b$ for the latitude, $\delta\hat{\boldsymbol{\lambda}}_b$ for the longitude, $\delta\hat{\mathbf{h}}_b$ for the height, $\delta\hat{\mathbf{v}}^n_{eb}$ for the velocity and $\delta\hat{\mathbf{C}}^n_b$ for the attitude are fed back to the inertial navigation processor, where they are used to correct the inertial navigation solution. In this way, the integrated navigation solution of the navigation system is the inertial navigation solution itself, and is obtained, respectively, for the orientation,$\hat{\mathbf{C}}^n_b$, the velocity, $\hat{\mathbf{v}}^n_{eb}$, and the position, as follows [41]:

$$\hat{\mathbf{C}}^{n+}_b = (\mathbf{I} - [\delta\hat{\boldsymbol{\psi}}^n_{nb}]_{\mathrm{x}})\hat{\mathbf{C}}^{n-}_b \tag{1.9}$$

$$\hat{\mathbf{v}}^{n+}_{eb} = \hat{\mathbf{v}}^{n-}_{eb} - \delta\hat{\mathbf{v}}^n_{eb} \tag{1.10}$$

$$\hat{\mathbf{L}}^+_b = \hat{\mathbf{L}}^-_b - \delta\hat{\mathbf{L}}_b \tag{1.11}$$

$$\hat{\boldsymbol{\lambda}}^+_b = \hat{\boldsymbol{\lambda}}^-_b - \delta\hat{\boldsymbol{\lambda}}_b \tag{1.12}$$

$$\hat{\mathbf{h}}^+_b = \hat{\mathbf{h}}^-_b - \delta\hat{\mathbf{h}}_b \tag{1.13}$$

where superscripts - and + are used to indicate the solution before and after correction, "e" denotes the Earth-Centered Earth-Fixed (ECEF) reference frame, while "n" is used because the solution is resolved with respect to the axes of the NED

coordinate frame, and b represents the inertial platform coordinate frame. $\delta\hat{\boldsymbol{\psi}}^n_{nb}$ is the vector of Euler angles of the correction, and $[\delta\hat{\boldsymbol{\psi}}^n_{nb}]_{\times}$ represents the antisymmetric matrix

$$[\delta\hat{\boldsymbol{\psi}}^n_{nb}]_{\times} = \begin{bmatrix} 0 & -\delta\hat{\psi}^n_{nb,z} & \delta\hat{\psi}^n_{nb,y} \\ \delta\hat{\psi}^n_{nb,z} & 0 & -\delta\hat{\psi}^n_{nb,x} \\ -\delta\hat{\psi}^n_{nb,y} & \delta\hat{\psi}^n_{nb,x} & 0 \end{bmatrix}$$

The pseudocode (1) outlines the procedural steps of the implemented INS-GNSS loosely coupled algorithm. This algorithm's initial phase involves setting crucial parameters to account for IMU errors.

Algorithm 1 Loosely coupled algorithm

```
Set initialisation parameters for IMU errors;
f_IMU, init_time, init_imu_samples(init_time * f_IMU) and level_time;
for i = 1 to init_imu_samples do
    Initialisation loop
end for
for i = init_imu_samples + 1 to end do
    Specific force f^b_ib ← fb_IMU(i);
    Angular velocity ω^b_ib ← wb_IMU(i);
    Correct f and w using estimated biases;
    Update navigation solution with mechanisation equations;
    Apply ZUPT detection algorithm;
    if There are new r, v solution to measure then
        Apply Kalman Filter;
        Update the navigation solution with Kalman filter estimates;
    end if
end for
```

Setting initialisation parameters for IMU errors

Inertial navigation systems are renowned for delivering highly accurate position, velocity, and attitude information, particularly over short time spans. However, this precision degrades significantly over extended periods due to inherent error sources within the sensors. To address this challenge, the algorithm's first step focuses on precisely defining the initialisation parameters related to IMU errors. This meticulous calibration lays the groundwork for mitigating sensor inaccuracies and ensures the subsequent integration of INS-GNSS data yields precise and reliable navigation estimates.

The Allan variance method characterises various error types present in inertial sensor data. This method represents root mean square (RMS) random drift error as

a function of averaging time [42]. Inertial measurements were analysed using the Allan variance method; precisely, specific force and angular velocity measurements collected from the IMU during a stationary five-hour test were utilized. These measurements were crucial in deriving parameters such as velocity random walk, angle random walk, angle rate random walk, bias instability, correlation times, and dynamic bias root-PSD for the accelerometer and gyroscope.

The Initialisation Loop

As a systematic iteration over IMU samples, this loop (2) calculates essential parameters and initializes the navigation solution when the system is stationary. Its execution sets the stage for optimal accuracy and performance of the Kalman filter during subsequent integration, establishing a robust foundation for a precise and reliable navigation solution.

Within each iteration over the IMU measurement taken in the first seconds, the initialisation loop calculates the IMU sampling interval (tor_i) considering the IMU frequency, facilitating precise temporal alignment. Specific force (f_{ib}^b) and angular velocity (ω_{ib}^b) measurements are extracted from the IMU, providing essential motion-related data. The algorithm (2) checks for the availability of new GNSS measurements within the current Inertial Navigation System (INS) time. If available, it updates the GNSS position. IMU measurements are accumulated for levelling purposes and also GNSS data, when available, is accumulated to initialize position parameters. Roll and pitch are computed based on the averaged specific force measurements through the levelling process. The estimated position is initialized using the averaged GNSS positions.

Algorithm 2 Initialisation Loop

gnss_eps= ε; %duration in seconds
for $i = 1$ to init_imu_samples **do**
  if $i == 1$ **then**
    $tor_i \leftarrow \frac{1}{frequency_{IMU}}$;
  else
    $tor_i \leftarrow t_{IMU}(i) - t_{IMU}(i-1)$;
  end if
  $f^b_{ib} \leftarrow fb_{IMU}(i)$;
  $\omega^b_{ib} \leftarrow wb_{IMU}(i)$;
  $is\_gnss\_available \leftarrow$ false;
  $gdx \leftarrow$ find($t_{GNSS} \geq (t_{IMU}(i) -$ gnss_eps) and $t_{GNSS} < (t_{IMU}(i) +$ gnss_eps));
  %Check if there is a new GNSS measurement to process at current INS time
  if (!isempty(gdx) and $gdx > 1$) **then**
    $is\_gnss\_available \leftarrow$ true;
    $last\_gdx \leftarrow gdx$;
    gnss position $GNSS\_r \leftarrow$ [gnss_lat(gdx); gnss_lon(gdx); gnss_h(gdx)];
  end if
  $f\_leveling^b_{ib} \leftarrow [f\_leveling^b_{ib}, f^b_{ib}]$;
  $\omega\_leveling^b_{ib} \leftarrow [\omega\_leveling^b_{ib}, \omega^b_{ib}]$;
  % accumulate GNSS solutions to initialize position
  if $is\_gnss\_available$ **then**
    $GNSS\_r\_init = [GNSS\_r\_init, GNSS\_r]$;
  end if
  if ($i >$ init_imu_samples_level and !isempty($GNSS\_r\_init$)) **then**
    $ave\_f^b_{ib} \leftarrow$ mean($f\_leveling^b_{ib}$, 2);
    $roll \leftarrow$ atan2($-ave\_f^b_{ib}(2), -ave\_f^b_{ib}(3)$);
    $pitch \leftarrow$ atan($ave\_f^b_{ib}(1)/\sqrt{ave\_f^b_{ib}(2)^2 + ave\_f^b_{ib}(3)^2}$);
    $est\_r^n_{eb} \leftarrow$ mean($GNSS\_r\_init$, 2); %Initializes the estimated position
  end if
end for

Specific force and angular velocity error model

The primary sources of error are modelled as follows:

$$\tilde{\mathbf{f}}^b_{ib} = \mathbf{f}^b_{ib} + \mathbf{b}_a + \mathbf{n}_a \tag{1.14}$$

$$\tilde{\boldsymbol{\omega}}^b_{ib} = \boldsymbol{\omega}^b_{ib} + \mathbf{b}_g + \mathbf{n}_g \tag{1.15}$$

where $\mathbf{b}_a$ and $\mathbf{b}_g$ are biases, $\mathbf{n}_a$ and $\mathbf{n}_g$ are random noises.

Navigation solution update

Mechanisation equations (in discrete form) [43] are used to update the solution of inertial navigation at the next time instant, using measurements of angular velocity $\boldsymbol{\omega}_{ib}^{b}$ and acceleration $\mathbf{f}_{ib}^{b}$ from the IMU sensors. An optimised version was used since mechanisation equations result from approximations and may introduce errors.

Orientation update: if the angular velocity remains constant, i.e., if the integration interval τ is sufficiently small,

$$\mathbf{C}_{b}^{n}(t+\tau) \approx \mathbf{C}_{b}^{n}(t)(\mathbf{I} + [\boldsymbol{\omega}_{ib}^{b}]_{\mathrm{X}}\tau) - (\boldsymbol{\Omega}_{ie}^{n} + \boldsymbol{\Omega}_{en}^{n})\mathbf{C}_{b}^{n}(t)\tau \tag{1.16}$$

where:

$$\boldsymbol{\Omega}_{ie}^{n} = [\boldsymbol{\omega}_{ie}^{n}]_{\mathrm{X}} = \boldsymbol{\omega}_{ie} \begin{pmatrix} 0 & \sin(L_b) & 0 \\ -\sin(L_b) & 0 & -\cos(L_b) \\ 0 & \cos(L_b) & 0 \end{pmatrix} \tag{1.17}$$

$$\boldsymbol{\Omega}_{en}^{n} = [\boldsymbol{\omega}_{en}^{n}]_{\mathrm{X}} \Leftarrow \boldsymbol{\omega}_{en}^{n} \begin{pmatrix} v_{eb,E}^{n}/R_E(L_b) + h_b \\ -v_{eb,E}^{n}/R_N(L_b) + h_b \\ v_{eb,E}^{n}\tan(L_b)/R_E(L_b) + h_b \end{pmatrix} \tag{1.18}$$

R_E and R_N are respectively the radius of transverse curvature and the radius of curvature of the meridian at that point. $\boldsymbol{\Omega}_{ie}^{n}$ and $\boldsymbol{\Omega}_{en}^{n}$ are skew-symmetric matrices of respective rotation rates. To have the optimisation, we define the attitude axis update matrix as the coordinate transformation matrix from the body reference frame at the end of the update ($b+$) to the body reference system at the beginning ($b-$)

$$\mathbf{C}_{b+}^{b-} = exp[\boldsymbol{\alpha}_{ib}^{b}]_{\mathrm{X}} = \sum_{r=0}^{\infty} \frac{[\boldsymbol{\alpha}_{ib}^{b}]_{\mathrm{X}}^{r}}{r!} \tag{1.19}$$

Where $\boldsymbol{\alpha}_{ib}^{b}$ is the attitude increment.
Truncating the formula to the fourth order gives the Rodrigues formula, used to calculate:

$$\mathbf{C}_{b}^{n}(t+\tau) \approx \left[\mathbf{I} - (\boldsymbol{\Omega}_{ie}^{n} + \frac{1}{2}\boldsymbol{\Omega}_{en}^{n}(t) + \frac{1}{2}\boldsymbol{\Omega}_{en}^{n}(t+\tau))\tau\right] \mathbf{C}_{b}^{n}(t)\mathbf{C}_{b+}^{b-} \tag{1.20}$$

where $\boldsymbol{\Omega}_{en}^{n}(t+\tau)$ is calculated from $\mathbf{L}_b(t+\tau), \boldsymbol{\lambda}(t+\tau), \mathbf{h}_b(t+\tau)$.

Velocity update:

$$\mathbf{v}_{eb}^{n}(t+\tau) = \mathbf{v}_{eb}^{n}(t) + [\mathbf{f}_{ib}^{n} + \mathbf{g}_{b}^{n}(L_b, h_b) - (\boldsymbol{\Omega}_{en}^{n} + 2\boldsymbol{\Omega}_{ie}^{n})\mathbf{v}_{eb}^{n}(t)]\tau \tag{1.21}$$

where the acceleration due to gravity, $\mathbf{g}$, is modeled as a function of latitude and height.

Position update:

$$\mathbf{h}_b(t+\tau) \approx \mathbf{h}_b(t) - \frac{1}{2}[\mathbf{v}^n_{eb,D}(t) + \mathbf{v}^n_{eb,D}(t+\tau)] \tag{1.22}$$

$$\mathbf{L}_b(t+\tau) \approx \mathbf{L}_b(t) + \frac{1}{2}\left[\frac{\mathbf{v}^n_{eb,N}(t)}{R_N(\mathbf{L}_b(t)) + \mathbf{h}_b(t)} + \frac{\mathbf{v}^n_{eb,N}(t+\tau)}{R_N(\mathbf{L}_b(t)) + \mathbf{h}_b(t+\tau)}\right]\tau \tag{1.23}$$

$$\boldsymbol{\lambda}_b(t+\tau) \approx \boldsymbol{\lambda}_b(t) + \tfrac{1}{2}\left[\frac{\mathbf{v}^n_{eb,E}(t)}{(R_N(\mathbf{L}_b(t))+\mathbf{h}_b(t))\cos\mathbf{L}_b(t)} + \frac{\mathbf{v}^n_{eb,E}(t+\tau)}{(R_N(\mathbf{L}_b(t+\tau))+\mathbf{h}_b(t+\tau))\cos\mathbf{L}_b(t+\tau)}\right]\tau \tag{1.24}$$

In order to have the velocity and position accuracy update, we considered the use of the mean transformation matrix $\bar{\mathbf{C}}^n_b$ in the transformation of the specific force in the NED coordinate system:

$$\bar{\mathbf{C}}^n_b = \mathbf{C}^{n-}_b \mathbf{C}^{b-}_{\bar{b}} \tag{1.25}$$

where

$$\mathbf{C}^{b-}_{\bar{b}} = \mathbf{I} + \frac{1-\cos\left|\boldsymbol{\alpha}^b_{ib}\right|}{\left|\boldsymbol{\alpha}^b_{ib}\right|^2}[\boldsymbol{\alpha}^b_{ib}]_{\mathrm{x}} + \frac{1}{\left|\boldsymbol{\alpha}^b_{ib}\right|^2}\frac{1-\sin\left|\boldsymbol{\alpha}^b_{ib}\right|}{\left|\boldsymbol{\alpha}^b_{ib}\right|}[\boldsymbol{\alpha}^b_{ib}]^2_{\mathrm{x}} \tag{1.26}$$

obtaining

$$\mathbf{f}^n_{ib} = \bar{\mathbf{C}}^n_b \mathbf{f}^b_{ib}, \qquad \bar{\mathbf{C}}^n_b = \mathbf{C}^{n-}_b \mathbf{C}^{b-}_{\bar{b}} - \frac{1}{2}(\boldsymbol{\Omega}^n_{ie} + \boldsymbol{\Omega}^n_{en})\mathbf{C}^{n-}_b \tau \tag{1.27}$$

ZUPT detection algorithm

To make the algorithm (3) more efficient, the zero velocity update (ZUPT) algorithm was used to correct errors accumulated in the inertial navigation data when the system is stationary [44]. Thus, when the average velocity is below a fixed threshold (empirical estimated) for a fixed time period, the algorithm assumes the system is stationary and updates the current attitude and position based on the averages of the previous values.

1.5.3 Kalman Filter

Whenever corrections from the integration filter are applied, the value of the corresponding state of the Kalman filter is reset to zero. The algorithm implemented for the Kalman filter is as follows.

1. Calculation of the covariance matrix of the prediction error at instant k:

$$\hat{\mathbf{x}}^-_k = 0 \tag{1.28}$$

$$\mathbf{P}^-_k = \boldsymbol{\Phi}_{k-1}\mathbf{P}^+_{k-1}\boldsymbol{\Phi}^T_{k-1} + \mathbf{Q}_{k-1} \tag{1.29}$$

2. Calculation of the Kalman gain matrix:

$$\mathbf{K}_k = \mathbf{P}_k^- \mathbf{H}_k^T (\mathbf{H}_k \mathbf{P}_k^- \mathbf{H}_k^T + \mathbf{R}_k)^{-1} \tag{1.30}$$

3. Filtering of the system state at instant k based on measurements, and matrix of covariance of the filtered value estimation error:

$$\hat{\mathbf{x}}_k^+ = \mathbf{K}_k \delta \mathbf{z}_k^- \tag{1.31}$$

$$\mathbf{P}_k^+ = (\mathbf{I} - \mathbf{K}_k \mathbf{H}_k) \mathbf{P}_k^- \tag{1.32}$$

The state vector includes orientation, velocity and position errors, as well as the biases of the accelerometer and gyroscope, and is given by:

$$\mathbf{x}^n = [\delta \boldsymbol{\psi}_{eb}^n \quad \delta \mathbf{v}_{eb}^n \quad \delta \mathbf{p}_b \quad \mathbf{b}_a \quad \mathbf{b}_g]^T \tag{1.33}$$

where

$$\delta \mathbf{p}_b = [\delta \hat{\mathbf{L}}_b \quad \delta \hat{\boldsymbol{\lambda}}_b \quad \delta \hat{\mathbf{h}}_b]^T \tag{1.34}$$

The transition matrix $\boldsymbol{\Phi}$ is obtained by computing the expected value of the time derivative for each state. It's important to note that all estimated quantities are derived based on preceding correction. $\mathbf{Q}$ is the system noise covariance matrix, $\mathbf{H}$ is the measurement matrix, $\mathbf{R}$ is the measurement noise covariance matrix, and $\delta \mathbf{z}_k^-$ is the vector of observations; the matrices used in this study are resolved in a Local Navigation Frame and can be found in Appendix J1.7 of J1.

Chapter 2

Thesis contribution

Referring to the context defined in chapter 1, this thesis investigates the leading localization technologies, offering innovative solutions and methodologies to meet the evolving challenges and demands of modern navigation systems. In particular, it has focused on development through SDR technology, taking advantage of the flexibility and capability of these software-based RF devices. The application and solution development process has concentrated on three main areas: the development of SDR-based GNSS solutions (multi-constellation Receiver and Simulator), the development of monitoring and location services based on SDR technologies, and industrial applications for the navigation through a multi-sensor approach.

The first aspect addressed through study, analysis, and implementation is GNSS receivers based on SDR technology. The starting point involved studying GPS signals in RF and baseband to propose a GNSS signal generator to allow us to model the main physical phenomena affecting these signals. In this case, an SDR approach was also used to generate RF signals. The development tools (GNU Radio) and the HW components studied in this research phase consent to efficiently validate implementation proposals in the GNSS field. Specifically, SDR platforms such as USRP, LimeSDR, ADALM-PLUTO, and RTL-Dongle were evaluated. This analysis of HW platforms and solutions constitutes a contribution when facing advanced and secure solutions in the GNSS field.

Additionally, the proposal of implementation solutions based on SDR ensures a framework usable in future contexts by exploiting this technology's flexibility and modular nature. The preliminary studies, which included the implementation of the GPS receiver, were extended to a multi-constellation and multi-frequency approach. The laboratory-developed elements for simulating GNSS signals constitute a framework for experimental activity.

Another significant contribution is the integration of augmentation services into the localization process. By leveraging these services to correct atmospheric and clock errors in GNSS signals, the thesis addresses key sources of inaccuracies in positioning, resulting in improved overall performance and increased reliability of localization systems. Specifically, IP-based services offering Atmospheric and Clock corrections are studied and experimentally evaluated. This aspect allows for developing positioning solutions that combine PPP-RTK, thus enhancing the main navigation KPIs: accuracy, availability, continuity, and integrity.

Starting from an advanced GNSS system, the thesis establishes a comprehensive theoretical framework encompassing fundamental navigation principles, sensor fusion techniques, and critical issues in localization technology. This theoretical foundation provides valuable insights into the complexities of designing localization systems and is a basis for developing innovative methodologies. Indeed, one of the most significant contributions is proposing a multisensor integration-based architecture within the EMERGE project framework [21].

Creating an On-Board Unit (OBU) for navigation presents a platform that constitutes a valuable test facility. The system developed, part of the instrumentation incorporated into the vehicles used in the EMERGE project, ensures the real-time operation of services dedicated to decoding, communicating, and integrating data flows from remote or local sensors. Additionally, it can be used as a tool for validating and testing innovative solutions in sensor fusion. The infrastructure developed can also monitor or evaluate innovative functionalities in the integrity field.

The experimental activities carried out within this thesis evaluate not only the SDR-based GNSS and integrated navigation solution but also aspects related to SDR-based monitoring and localization services. This thesis presents a methodology for positioning using Software-Defined Radio (SDR) and the ADS-B signals. By leveraging the flexibility of SDR, the research introduces novel techniques for filtering and discriminating Mode S messages, employing advanced temporal and spatial statistical methods. These techniques significantly enhance the accuracy and reliability of ADS-B-based localization, providing a robust framework for integrating these signals into the general SDR-based localization process. Although the channel model has potential for further refinement, the methodology developed offers a solid foundation for future advancements in ADS-B signal processing within the SDR paradigm. In addition to improving ADS-B signal processing, this thesis extends the application of SDR technology to detect and localise interference sources, applying spectrum analysis, channel modelling, and MLAT. By implementing sophisticated signal processing

techniques and deploying distributed sensor networks via SDR, we introduce novel methodologies for monitoring and localizing interference sources. These advancements are particularly valuable in some specific scenarios and challenging environments.

This thesis significantly advances SDR-based localization technology, offering innovative solutions and methodologies that improve positioning solutions' accuracy, reliability, and resilience. These contributions broadly affect various applications, including autonomous navigation, precision agriculture, urban mobility, and more. They contribute to advancing localization technology and its widespread adoption in real-world scenarios. Table 2.1 summarizes the activities considered in the works appended to this thesis.

Table 2.1: Key aspect addressed in this thesis. The checkmarks are used to indicate which functionality is described in the corresponding paper

	SDR-based GNSS solutions	SDR-based monitoring and localization service	Multi-sensor Localization	
C1	✓			SDR-based GNSS receiver and simulator
C2		✓		SDR-based ADS-B receiver for localization
C3		✓		SDR-based monitoring and localization
J1			✓	Navigation using multi-sensor approach

2.1 Publications appended to the thesis

The chapters presented in the second part are based on the following publications or accepted manuscripts. Below, one briefly presents the main contributions of each chapter (publication or manuscript)[1].

C1 A. L. Z. Sosa, R. Alesii and F. Santucci, "Cross-platform evaluation for Software Defined Radio GNSS receiver," *2022 3rd URSI Atlantic and Asia Pacific Radio Science Meeting (AT-AP-RASC)*, 2022, pp. 1-4, DOI: 10.23919/AT-APRASC54737.2022.9814436.

This paper describes the main operational signals in GNSS systems that can be used in a multi-constellation, multi-frequency receiver. Also, it proposes the reception of GNSS signals using a modular architecture based on SDR. The SDR GNSS receiver is confronted with two platforms operating as Front-Ends: ADALM-PLUTO and Ettus USRP X310 - UBX160.The results, corresponding to the capture of GPS signals in the L1 band, show the impact of the platform's performances in the satellite acquisition signal process. The results and methodologies applied in this work were used in the multi-frequency and multi-constellation use cases. Furthermore, the limits identified in using SDR regarding the Local Oscillator stability become a starting point for selecting the GNSS platform.

Author's contribution: The author contributed to the idea formulation, developed a prototype of the proposed SDR GNSS receiver, performed the experimental activities to generate the results and wrote the manuscript.

C2 A. L. Z. Sosa, R. Alesii and F. Santucci, "Opportunistic RSS-based localisation using SDR and ADS-B system," *2024 4th URSI Atlantic Radio Science Meeting (AT-RASC)*, Meloneras, Spain, 2024, pp. 1-4, DOI: 10.46620/URSIATRASC24/JTMJ9870.

This activity presents a novel SDR-based solution for opportunistic localization using ADS-B Mode S signals. The methodological approach based on experimentation to describe the channel can be extended to other application fields, exploiting the SDR paradigm's modularity and flexibility. From the point of view of the results in terms of errors in the localization domain, it constitutes a starting point for future solutions closer to a real-time approach. Regarding the

[1] **J:** Journal, **C:**Conference.

power analysis, some outliers are registered and must be filtered out before the distance calculation. The solution proposed in this work consists of a time-based filtering analysis. Both aircraft interfaces (Top/Bottom) transmit messages in alternate mode every 0.5 seconds, with a maximum error of 0.1 seconds. This feature allows us to correctly group the information from each interface and promptly discard the values that introduce a significant error. Another element that has been detected and is proposed to be addressed in future work is the error introduced by the antenna radiation pattern itself, which can be attenuated by using gain masks depending on the aircraft's orientation. The proposed techniques to improve localization by timely identifying the phenomena affecting the ADS-B signals are excellent, showing a considerable improvement in the results found in the literature.

Author's contribution: The author contributed to the idea formulation, developed the SDR-based ADS-B receiver, performed the simulation with the analysis of the results and wrote the manuscript.

C3 A. Piccioni, A. L. Z. Sosa, R. Alesii and F. Graziosi, "SDR-Based Distributed System for Mobile Communication Network Monitoring and Support," ACCEPTED IN *Next-Generation Multimedia Services at the Edge: Leveraging 5G and Beyond - co-located with ISCC June 26, 2024 // Paris, France.*

This work presents a novel distributed system capable of supporting modern mobile communication networks through the advantages and flexibility of SDR technology. In the idea behind this work, the proposed system can execute different services depending on the requirements. These can be classified as default services like power emissions analysis and spectrum monitoring or critical services in uncommon scenarios where all the fundamental services cannot be guaranteed. A prototype of the proposed systems has been developed through the SDR platform NI USRP 2954R; in the first stage of this research activity, the prototype is capable of monitoring the spectrum through spectral analysis, signal detection and localization techniques. The prototype has been tested in a limited environment through an interfering source to evaluate the system's validity rather than verify each technique's performance since they depend on the adopted parameters and the environment itself.

Author's contribution: The author contributed to the idea formulation regarding the localization service, developed a prototype of the proposed SDR-

based system and tested it, analysed the results and wrote some sections of the manuscripts.

J1 Zuriarrain Sosa, A.L.; Ioannucci, V.; Pratesi, M.; Alesii, R.; Albanese, C.; Valentini, F.; Cinque, E.; Martinelli, A.; Brizzi, M. OBU for Accurate Navigation through Sensor Fusion in the Framework of the EMERGE Project. *Appl. Sci. 2024, 14, 4401.* DOI: 10.3390/app14114401.

This work presents a novel architecture for localization using a multi-sensor approach. The architecture uses communication interfaces with available infrastructures, including satellite constellations, road structures, and adjacent vehicles. This work was motivated by the EMERGE project and describes the realisation of an On Board Unit (OBU) dedicated to the navigation process. The OBU is equipped with the Xsens MTi-630 AHRS inertial sensor, a multi-constellation/multi-frequency GNSS receiver with the ublox ZED-F9P module and communication interfaces that provide access to the PointPerfect augmentation service. Experimental results support using GNSS corrections to achieve centimetre accuracy in the satellite positioning with a Time To First Fix (TTFF) of 30 seconds. However, in dynamic scenarios, the GNSS correction rate affects the GNSS results(the subscription to the MQTT root topic provides corrections every 30 seconds). The on-road test collects the Sensor Fusion filter's successful response, the monitoring module's correct functioning, and an OBU power consumption under 5 W at a high operating regime.

Author's contribution: The author contributed to the idea formulation, methodology, investigation, writing—original draft preparation, formal analysis and validation. A prototype was also developed to test the complete OBU in a real environment.

Part II

Appended Papers

Conference C1

Cross-platform evaluation for Software Defined Radio GNSS receiver

Angel Luis Zuriarrain Sosa, Roberto Alesii, and Fortunato Santucci

In proceedings of
2022 URSI AT-AP-RASC, Gran Canaria, ES

Abstract

The present work is part of a strategy that aims at a complete navigation system that offers solutions compatible with the autonomous vehicle and smart city paradigm. Nowadays, Global Navigation Satellite Systems (GNSS) and complementary positioning technologies are expanding, offering new resources to the localization process. Software Defined Radio (SDR) is emerging as an alternative to developing flexible and multi-technology solutions in a dynamic environment. This paper describes the main operational signals in GNSS systems that can be used in a multi-constellation, multi-frequency receiver. Also, it proposes the reception of GNSS signals using a modular architecture based on SDR. The SDR GNSS receiver is confronted with two platforms operating as Front-Ends: ADALM-PLUTO and Ettus USRP X310 - UBX160.The results, corresponding to the capture of GPS signals in the L1 band, show the impact of the platform's performances in the satellite acquisition signal process.

C1.1 Introduction

Demographic trends indicate a displacement of the rural population to urban areas. It is expected that almost 70% of the population will be concentrated in big cities [45]. For this reason, managing existing resources and infrastructure to meet the growing needs of the urban population in sustainable living conditions has become a significant challenge. Smart mobility constitutes a key aspect in the development of smart cities. Smart mobility includes high flexibility, scalability, integration, efficiency in energy resources, and automated vehicles (AVs) or "self-driving" cars. There are many approaches behind smart mobility: use of technologies that provide helpful information: IoT, sensors, networks, and satellites; the creation of structures that include smart roads and monitoring platforms; also, solutions that target the environment-friendly: electric car, cyclin path. Safely and efficiently, positioning, navigation, and timing (PNT) services support most "smart" processes in mobility. These services seem immediate when applications such as traffic management, access control, autonomous mobility, precise positioning, public health and safety, critical infrastructures, or security are some of the goals driving innovation in modern urban scenarios [2, 3].
The most widely used and widespread navigation system is the Global Navigation Satellite System (GNSS), thanks to its worldwide coverage and free provision of absolute positioning solutions. With the increasing adoption and availability of GNSS signals, frequencies, and services, user technologies have evolved and spread across many devices and applications. In Section C1.2 of this paper, a description of the open-civil signals and services offered by the GNSS system is carried out.
The fact that GNSS technology is one of the most scalable and frequently used implies the design of solutions that offer modularity and flexibility [46]. Therefore, the goal of the present work includes a modular approach for the localization process, starting from a GPS receiver. Software Defined Radio (SDR) is an excellent solution in developing systems that integrate different technologies [47]. In addition, it allows adapting to the technological development of new devices and the availability of new GNSS satellite services. Although the SDR concept is not new, the evolution of integrated circuits has made possible the implementation of hardware with an acceptable level of performance in the satellite environment.
Our SDR GNSS receiver scheme incorporates a first Front-End stage and then the digital processing chain. The Front-End performs three fundamental functions: filtering, downconversion of the analog signal, and digital conversion. SDR platforms

can implement two different receiving topologies: heterodyne or homodyne. A heterodyne receiver has multiple downconversion stages where amplification and filtering are applied, increasing complexity and costs and eliminating the image band [48]. In the homodyne case, the signal is converted directly to baseband and although it is less robust than the heterodyne is the simplest and lowest cost solution. Another problem related to the topology used in reception is that the heterodyne performs the conversion to an intermediate frequency (IF), thus the conversion to baseband is performed digitally within the receiver processing. In a single-stage, direct conversion to baseband implies less stability in the RF amplifiers and a less accurate filtering process. In addition, the use of a direct baseband conversion in GNSS systems can verify aliasing when performing Doppler compensation or detection.

This work includes the evaluation of two different SDR platforms in the satellite identification process. ADALM-PLUTO and Ettus USRP with UBX-160 daughterboard, implement direct downconversion in the L1 frequency band. The use of the USRP platform is widely spread in SDR GNSS receivers [49–53] and the good accuracy levels in the local oscillator (LO) guarantee the correct satellite identification. In the case of the ADALM-PLUTO platform, it is little used as Front-End because the LO accuracy levels exceed the expected threshold of the Doppler effect [ref]. Recently, some works have focused on error correction using a previous calibration process. In [54] a calibration process using the bladeRF 2.0 platform as the transmitter is proposed. Although the works consulted do not make an extended analysis on the limitations of SDR platforms in GNSS environment, it is important to highlight the open-source solution proposed in [26]. The GNSS-SDR project[sdrgnss] is based on the architecture proposed by Fernandes-Prades et.al that includes the generic use of a Front-End and a software receiver inside a Linux or MacOS environment. The receiver is highly configurable, but at the same time has many Software dependencies (including GNU Radio). The receiver's performance depends on the host PC's computational capabilities. Another limitation of GNSS-SDR is that it does not allow interaction with other localization techniques. In addition, it is difficult to modify all the configuration aspects of the acquisition process. The calibration functionalities for frequency corrections are limited and not functional.

This paper describes the main operational signals in GNSS systems that can be used in a multi-constellation, multi-frequency receiver. Our receiver includes a scalable modular architecture that uses, as Front-Ends, SDR platforms. The GNSS receiver scheme is evaluated for the specific case of GPS signals in L1 band. Subsequently, the

performance of two SDR platforms is analyzed from the point of view of signal identification capability, local oscillator accuracy, and carrier frequency detection ability. The acquisition is performed in a controlled environment, simultaneously and using a scheme where the antenna is shared from a power divider.
The paper is organized as follows: Section C1.2 resume the GNSS signal specifications, Section C1.3 describes the block functionalities of the SDR GNSS receiver, Section C1.4 presents the performances of the two SDR platform used, and finally, Section C1.5 concludes the paper.

C1.2 GNSS-based localization

Global Navigation Satellite Systems are the most commonly used resource in the localization process. The localization is based on solving geometric problem, involving the distances of a user to at least four GNSS satellites with known coordinates. Therefore, the observable in a GNSS is the time required for a signal to travel from the satellite to the receiver. The architecture basically consists of three main segments: the space segment, which comprises the satellites; the control segment, which is responsible for the proper operation of the system; and the user segment, which includes the GNSS receivers [16].

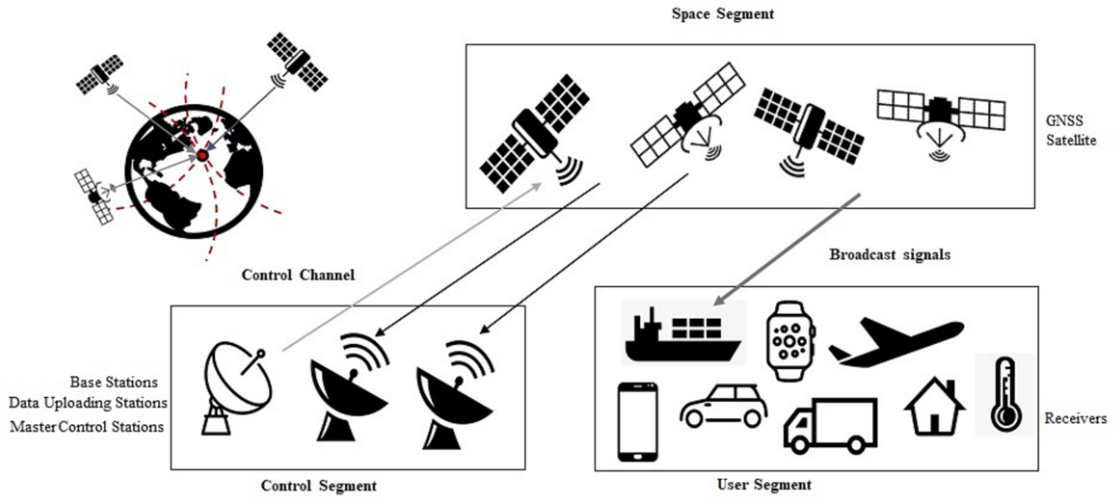

Figure C1.1: GNSS-based localization and architecture

Since the end of 2020 the Global Positioning System (GPS), the Russian Global Navigation Satellite System (GLONASS), the Chinese Navigation Satellite System

(BDS) and the European constellation, GALILEO are declared GNSS operational.[1] There are two other systems under development by Japan and India: Quasi-Zenith Satellite System (QZSS) and Indian Regional Navigation Satellite System (IRNSS) respectively. At the moment QZSS and IRNSS are regional systems, but a further global expansion is foreseen in the coming years.

C1.2.1 GNSS signals characteristics

Satellites, continuously transmit signals in L band (1-2 GHz). These signals contain codes and navigation data to allow users to compute the position. The data sent by the GNSS satellites include the high accuracy clock and ephemeris data as well as several other parameters related with the satellite health, orbit, tropospheric and ionospheric information. This information collected by the GNSS receivers is used to calculate the position, time and other necessary information. Table C1.1 shows the main features by constellation to be considered in the implementation of a GNSS receiver.

GPS was the first to provide global coverage and currently has 31 satellites in orbit. It offers four "open" or "civil" type signals distributed in the L1, L2 and L5 bands: L1-C/A, L1C, L2C and L5C. The GPS uses the CDMA technique to send different signals on the same radio frequency, and the modulation method used is Binary Phase Shift Keying (BPSK).[2] The L1-C/A signal is the most diffused signal for market applications. The Pseudorandom Noise Code (PRN) is an unique Gold code, of 1 millisecond in length at a chipping rate of 1.023 Mbps. Although the coding frequency is the same, the modulation used determines a minimum receive bandwidth of 2.046 MHz for L1-C/A and 4.092 MHz for L1C. The minimum receive power is higher for the new civil signal L1C. L2C signal is composed by two different PRN codes to provide ranging information; the civil-moderate code (called CM), and the civil-long length code (called CL). The CM code is 10230 bits long, repeating every 20 ms. The CL code is 767250 bits long, repeating every 1500 ms. Each signal is transmitted at 0.511 Mbits per second (bit/s); however, they are multiplexed together to form a 1.023 Mbit/s signal. The L5C signal was designed for users requiring Safety

[1] In the GALILEO space segment, a total of 30 satellites (24 active plus 6 spares) are planned. At present, it has 22 usable, 4 not available, 3 not usable and 2 under commissioning (01/2022) [55]

[2] The L1C signal is designed to enable interoperability between GPS and others international satellite navigation systems. Multiplexed Binary offset Carrier (MBOC) modulation is used to improve mobile reception in cities and other challenging environments. L1C comprises the L1C-I data channel and L1C-Q pilot channel.

of Life (SoL) applications. There are two signal components: the in-phase component (L5I) with data and ranging code, both modulated via BPSK onto the carrier; and the quadrature component (L5Q), with no data but also having a ranging code BPSK modulated onto the carrier. This signal has an improved code/carrier tracking loop and its high power and signal design provide robustness against interference. It is important to note that L1C, L2C and L5C will be of limited use until they are broadcast from 18 to 24 satellites [18].

Galileo satellites transmit permanently three independent Code Division Multiple Access signals, named E1, E5 and E6. In Table C1.1, the characteristics of the Open Services (OS) offered in the E1 and E5 band are described. E1 supports the OS, CS, SoL and PRS services. It contains three navigation signal components in the L1 band and two components, E1-B and E1-C, are open access signals with unencrypted ranging codes accessible to all users. E1-B is a data channel and E1-C a pilot channel. The E1-B data stream, at 125 bps of navigation data, also contains unencrypted integrity messages and encrypted commercial data. The MBOC modulation is used for the E1-B and E1-C signals, which is implemented by the Composite Binary Oset Carrier (CBOC). Also in this case the modulation scheme impact in the bandwidth therefore 4 MHz are needed centered in 1.57542 GHz. The E5 signal is sub-divided into signals denoted E5a and E5b. The E5a and E5b signal components are modulated onto a single E5 carrier frequency at 1.191795 GHz using a technique known as Alternate Binary Offset Carrier (AltBOC). The composite signal E5 can be processed as a single large-bandwidth signal or as two different signals. In the receiver implementation the impact of bandwidth is very important, an approach that includes both components (E5a + E5b) requires more than 50 MHz, while a separate treatment implies a bandwidth of 20.46 MHz for each of the components.

Table C1.1: GNSS Open Signal characteristics

Constellation	Availability and Coverage	Carrier Frequency Center band[GHz]	Open Service Signals and Spreading Modulation	Code[Mcps]/ Data[bps] Rate	Minimum Bandwith [MHz]	Received Power [dBW]
GPS (US)	- 31 Satellites on sky - Global Coverage - Fully Operational	L1: 1.57542	- C/A: BPSK(1) - L1C: MBOC(6,1,1/11) (non-fully operational)	1.023/50	2.046 4.092	-158.5 -157
		L2: 1.22760	- L2C: BPSK(1) (pre-operational)	1.023/25	2.046	-161.5
		L5: 1.17645	- L5: BPSK(10) (pre-operational)	10.23/50	20.46	-157.9
GALILEO (EU)	- 24 Satellites on sky - 22 Satellites in usable condition - Global Coverage	E1: 1.57542	- E1 OS: MBOC(6,1,1/11)	1.023/125	8.184	-157
		E5a: 1.17645 E5b: 1.20714	- E5a-b: BPSK(10)	10.23/25-125	20.46	-155
BeiDou Phase III (CHN)	- 35 Satellites on sky - Global Coverage - Fully Operational	B1C: 1.57542	- B1-C: MBOC(6,1,1/11)	1.023/50	32.736	-159/-161
		B2: 1.17645/1.20714	- B2-a,b: AltBOC(15,10)	10.23/50	20.46	-163
GLONASS (RUS)	- 24 Satellites on sky - Global Coverage - Fully Operational	G1: 1.59806-1.60931 G2: 1.24293-1.25168	- C/A: BPSK(0.511) (FDMA)	0.511/50	15	-161 -167
		G3: 1.202025	- BPSK(10) (CDMA)	10.23/100	20.46	-161

The Beidou Navigation Satellite System, in phase III, provides global coverage for navigation through 35 satellites which support open services SPS (Standard Accuracy Signal Service). Also in this case, and with the objective of non-interfering frequency band allocation, MBOC and AltBOC spreading modulation are used for the B1C and B2 signal respectively. This has a direct impact on the minimum bandwidth necessary to receive the signals coming from the BeiDou constellation. The bandwidth required for the B1C signal, centred at 1.57542 GHz, is 32.736 MHz and for the B2 signal, centred at 1.17645/1.20714 GHz, is 20.46 MHz.
In contrast to the other constellations, each Glonass satellite broadcasts at a particular frequency within the band. This frequency determines the frequency channel number of the satellite and allows receivers to identify the satellites (with the Frequency-division multiple access technique). The CDMA Open Service Navigation Signal in L3 frequency band is called L3OC and consists of two BPSK(10) components: data and pilot.These components are in phase quadrature with each other and L3OCd is delayed 90° [19].
The integration of new open signals in GNSS systems with global coverage offers new opportunities in the design of positioning systems. The trend shows a homogenization of medium access techniques (CDMA) and modulation schemes. There is also an increase in bandwidth requirements due to frequency relocation. The expansion of GNSS systems and signals promises excellent performance in the localization process based on an approach capable of using all available resources simultaneously.

C1.3 Software Defined Radio GNSS receiver

A GNSS SDR receiver is a satellite receiver that has been designed and implemented using software-defined radio. This section proposes a modular architecture for GNSS signal reception and processing compatible with SDR Front-End devices. The Front-End, consist in an antenna and an SDR device that implements the down-conversion process directly in base-band in two components: in-phase and quadrature. Subsequently, both components are digital-converted, filtered and sent to the core of the software receiver.
The core is composed of the GNSS signal acquisition and tracking block and can be performed in parallel. The acquisition process is implemented to identify the available satellites and a first estimation of the frequency and phase of the carrier [27, 28]. After verification of positive acquisitions, the tracking block follows the evolution of

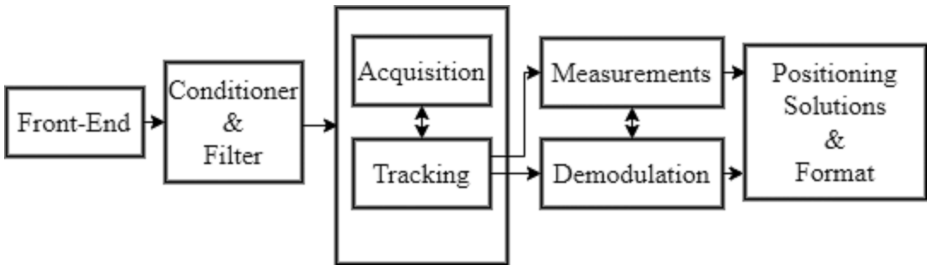

Figure C1.2: GNSS SDR receiver block diagram

the signal. When the detected signals are correctly tracked, it moves on to the demodulation of the navigation information and the measurement process which results in the calculation of the position. Finally, the results of the complete process are encoded in the compatible formats: RINEX, NMEA or KLM. Below, the GNSS SDR receiver scheme is applied from the implementation in GPS environment. However, the GNSS receiver scheme can interact with other localization techniques, i.e. it can be extended to hybrid localization. The location and navigation data provided by other technologies can be added directly into the "measurements" block or at a subsequent stage, from the receiver output file.

C1.3.1 SDR GPS receiver

The best known navigation system is the Global Positioning System (GPS), which was developed by the U.S. and has been operational since 1994. The most used open GPS service is the Legacy signal in L1 band (centered at 1.57542 GHz). The L1 signal is composed of two BPSK (binary phase shift keying) modulated signals, orthogonal to each other, i.e., with a phase offset of 90°, also known as the In-phase and Quadrature components. They contain the precision code (P) and Coarse/Acquisition (C/A) codes phase-locked together [18]. The C/A Code for Space Vehicle (SV) ID number is an unique Gold code, of 1 millisecond in length at a chipping rate of 1.023 Mbps. The Gold sequence is a linear pattern generated by the Modulo-2 addition of two subsequences: G1 and G2, each of them being a 1023 chip long linear pattern. The P code for SV number is a ranging code, P, seven days long at a chipping rate of 10.23 Mbps. Both codes are classified as pseudorandom noise code (PRN) and they have a similar spectrum to a random bit sequence.
For the implementation of the first stage of the GPS SDR receiver we use two different devices: USRP X310 and ADALM-PLUTO. Both devices have a Zero-IF or direct downconvertion architecture, which decomposes the signal in phase and quadrature.

Subsequently, both components are digitized and sent to the receiver core. Considering that the C/A code is present in the quadrature component after downconvertion process, equation C1.1 represents the superposition of all the components captured by the antenna in L1 band.

$$S_{L1} = \sum_{k=1}^{N} P_{CA-k} CA_k(t)\, N_{AV}(t) \sin\left[2\pi\left(f_{L1} + f_{D_k}\right) + \varphi_{i-k}\right] \; + \quad n(t) \tag{C1.1}$$

K, N, P_{CA-k}, $CA_k(t)$, $N_{AV}(t)$, f_{D_k} and φ_{i-k} represent the received satellite sequence number, the total received satellite number, the $k-th$ satellite power, the C/A code, the received ephemeris data $k-th$ satellite, the $k-th$ Doppler frequency and the initial frequency phase, respectively. The $n(t)$ component represent the noise. To simplify, the multipath components and other phenomena are modeled in the noise representation.

Acquisition

The function of acquisition block is to determine visible satellites and coarse values of carrier frequency and code phase of the satellite signals. The satellite identification process is performed by the autocorrelation process. PRN C/A codes have a high autocorrelation value and a low croscorrelation value, so they can be identified in noisy environments. To obtain a high autocorrelation value, it is necessary to align the locally generated code with the received code. In fact, an important element is the phase of the code that is also used in the distance measurements between the satellite and the receiver. The other element is the carrier frequency, which in case of downconversion corresponds to the Intermediate Frequency (IF). The IF should be known from the L1 carrier frequency of 1.57542 GHz and from the mixers in the downconverter. However, the frequency can deviate from the expected value. The relative velocity of the satellite causes a Doppler effect resulting in a higher or lower frequency. In the worst case, the frequency can deviate up to ± 5 kHz (stationary receiver).

The method used in the acquisition process is Parallel Code Phase Search Acquisition. The idea is to perform a correlation with the incoming signal and a local PRN in frequency domain. The in-phase and quadrature digital components coming from the front-end are multiplied by a locally generated carrier. The result corresponds to the C/A component when compensating for the Doppler effect suffered by the signal and other errors introduced in the downconvertion process. In order to sweep

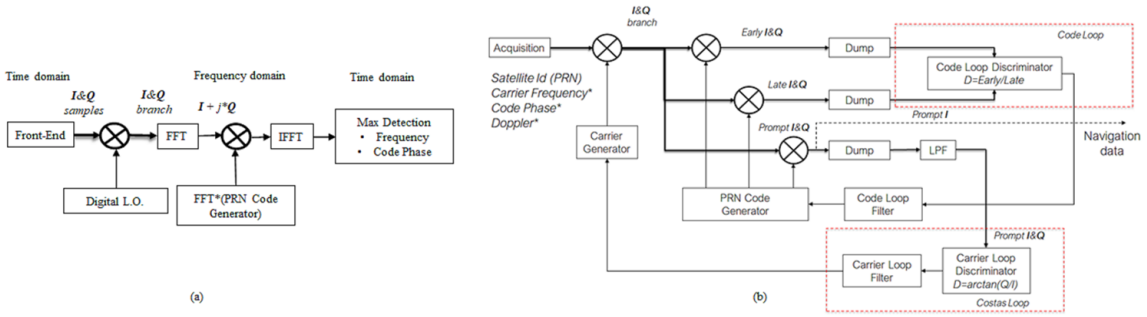

Figure C1.3: Core of the GNSS receiver. (a) Parallel Code Phase Search Acquisition diagram. (b) Tracking block schematics

the frequency range, 41 iterations with 250 KHz steps are performed. Each carrier multiplication process is converted in the frequency domain using the Fast Fourier Transform and multiplied by the conjugate of the local PRN code. This multiplication process in the frequency domain corresponds to the circular crosscorrelation operation in the time domain. The result of the multiplication is transformed into the time domain by an inverse Fourier transform. The absolute value of the output of the inverse Fourier transform represents the correlation between the input and the PRN code. If a peak is present in the correlation, the index of this peak marks the PRN code phase of the incoming signal. This algorithm is depicted in Figure C1.3 (a).

The Parallel Code Phase Search Acquisition algorithm has the better performances in term of processing time and number of iterations. Although it presents a major complexity due to the transformations between the time and frequency domain (FFT and IFFT). However, many mathematical tools have addressed the problem of complexity in the FFT transform. However, many mathematical tools have tackled the problem of complexity in the FFT transform. That is why our proposal, initially, proposes the use of Matlab in the elaboration of digital signals.

Tracking

The acquisition provides only rough estimates of the frequency and code phase parameters. The main purpose of tracking is to refine these values, follow the signal evolution, and demodulate the navigation data from the specific satellite.

The first step in the tracking process is to correlate the signal with a locally generated C/A code from the given satellite. This correlation removes the C/A code and produces the navigation message signal. But the samples to be correlated must be consistent with the phase calculated in the acquisition process. Frequently, due to

disturbances, the phase can shift a few places either to the right or to the left. This phase shift must therefore be compensated for in the code. In order to determine the direction of the shift, two additional PRN codes are introduced. The original PRN code is called prompt and is the one we try to keep aligned, while the new codes are called: early code and late code, shifted to the right and left respectively. In our proposal, the code tracking process is performed for each millisecond, and although it may appear inefficient, it actually unifies the work frequency of the entire block and reduces the complexity of the process.
The variation of the reference velocity between transmitter and receiver causes the Doppler effect to be inconsistent. In addition, SDR platforms can exhibit instability in the local oscillator. These phenomena support the use of a carrier tracking loop. The carrier loop discriminator block is used to find the phase error on the local carrier wave replica. The output of the discriminator, is then filtered and used as a feedback to the Carrier Generator, which adjusts the frequency of the local carrier wave. In this way the local carrier wave could be an almost precise replica of the input signal. The problem with using an ordinary Phase Lock Loop (PLL) is that it is sensitive to 180° phase shifts. Therefore, our proposal consists of a Costas Loop with tangential discriminator, insensitive for 180° phase shifts due to navigation bits.

C1.4 Results

In this section, the performance of the SDR GPS receiver is tested using synthetically generated signals and real signals acquired with two different devices. The interaction of the receiver with locally generated numerical signals allows us to characterize its response in a controlled environment. After verifying the software receiver's correct operation, we proceed to the realization of the acquisition with SDR devices at the same time and using identical instrumentation. The final goal is to characterize and evaluate the ADALM-PLUTO and USRP X310 platforms in the GPS satellite acquisition process.

Table C1.2: Acquisition process from numerically generated signals

Simulated Satellites	Carrier frequency error [Hz]	Doppler Frequency error [Hz]	Code Phase error [Samples]	Peak Metric
PRN 1	2	3	1	10.2
PRN 6	2	1	0	7.6
PRN 11	1	1	3	9.4
PRN 16	2	0	1	8.8

The GPS synthetic signal generator is developed using Matlab and performs a numerical simulation of the main physical phenomena affecting the signal: Doppler effect,

attenuation, multipath, and the noise introduced by the antenna at the receiver. The synthetic generator randomly generates an indicated number of Gold codes corresponding to a specific satellite, then encoded NRZ and mixes with the carrier. Once the components for each simulated satellite have been obtained, the superposition is carried out, taking into account the free-space path attenuation, the satellite orbit of 20.0 km, and the transmitter power of 26.8 dBW. In addition, constructive and destructive multipath interference is included from coefficients that vary the signal amplitude at specific carrier frequencies. Finally, the effect of thermal noise present in the antenna is simulated using specific numerical functions in Matlab.
Table C1.2 presents the results of applying the developed acquisition process to the generated synthetic signal. The signal generation is carried out with the following configuration: Carrier frequency 30.69 MHz, sampling frequency 122.76 MHz, Doppler frequency 5.0 kHz step, and a signal to noise ratio of -20 dB. In addition, the codes are shifted according to the ID of each satellite for subsequent identification and error calculation. The peak values correspond to the metric used by the receiver in the acquisition process as a result of the correlation process with the local codes. The peak values correspond to reference values concerning negative acquisitions. The results show a correct performance of the GPS software receiver, positively identifying the four simulated satellites. In addition, the carrier frequency detection algorithm presents low-order errors. Once the acquisition process has been tested using synthetic signals, we proceed to capture real signals. The tracking process is not evaluated since the generator does not yet simulate the Doppler frequency and phase changes of the code.

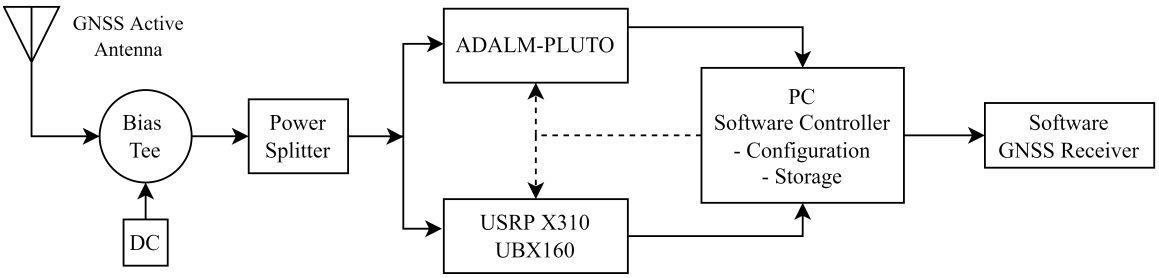

Figure C1.4: Multi-platform experimentation scheme

The main limitation of the synthetic generation of GPS signals using Matlab is that the navigation data bit frequency is very low with respect to the GPS signal carrier's frequency. That is why an intermediate frequency is used in the synthetic signal

generation process. In our case, we use a carrier frequency of 30.69 MHz, which does not interfere with the downconversion process to test the baseband's acquisition and tracking process. We plan to develop a synthetic signal generator that uses the real carrier frequency and introduce a multi-constellation approach in future work.
In our SDR-platform experiment, we propose to use the same RF reception chain formed by an active GPS antenna, a Bias Tee, and a power divider. Subsequently, the SDR platforms receive the configuration and the command of reception through software from a computer that controls the storage of the samples. We capture blocks of 1667 ms at 10 MSps in order to maintain temporal continuity in the acquisition. The results correspond to three captures performed at different time instants in a low-interference indoor environment of the campus of the University of Aquila, Italy. The DAM1575A23.3V antenna was used to capture the 1.57542 GHz L1 GPS signal. This antenna has a voltage standing wave ratio (VSWR) of 1.5:1, bandwidth of ± 5 MHz, and 50 ohm of impedance. The antenna include a LNA/Filter with 28 dB of gain and 7dB attenuation of over 20 MHZ around the center frequency (GPS L1). As noted in the block diagram of Figure C1.4, the antenna is fed by the Bias Tee TW154 Regulated 0.5 to 3GHz, 3.3V. The resistive Power Divider SMA 50 Ohm (Huber&Suhner) was used for the signal split: one component goes to the ADALM-PLUTO platform and the other to the USRP X310 with UXB160 daughterboard. For the SDR platforms' configuration process, we use a personal computer connected to both devices. The start of the acquisition is performed simultaneously, and the samples are stored in the PC. The samples from each device are converted to complex, defining the real part as the in-phase components and the complex part as the quadrature components. The sample blocks are passed to the GPS receiver software for processing and analysis.
The Ettus Research USRP X310 is a high-performance, scalable software defined radio (SDR) platform for designing and deploying next generation wireless communications systems. The hardware architecture combines two extended-bandwidth daughterboard slots covering DC – 6 GHz with up to 120 MHz of baseband bandwidth [ettus]. For this experiment, Ettus UBX-160 RF front-end daughterboards have been chosen for their characteristics in the receiver path. For frequencies in the 0.5 - 6 GHz range, Ettus UBX-160 performs the direct downconversion process. This feature allows a parity evaluation of the acquisition process, as the ADALM-PLUTO platform also features a zero-IF architecture. Certainly, other elements distinguish them in the GNSS field. A crucial aspect is the stability and accuracy of the local oscillator. The level of accuracy (without calibration) presented by the Ettus platform (2.5 ppm)

exceeds ten times the value of the ADALM-PLUTO (25 ppm).

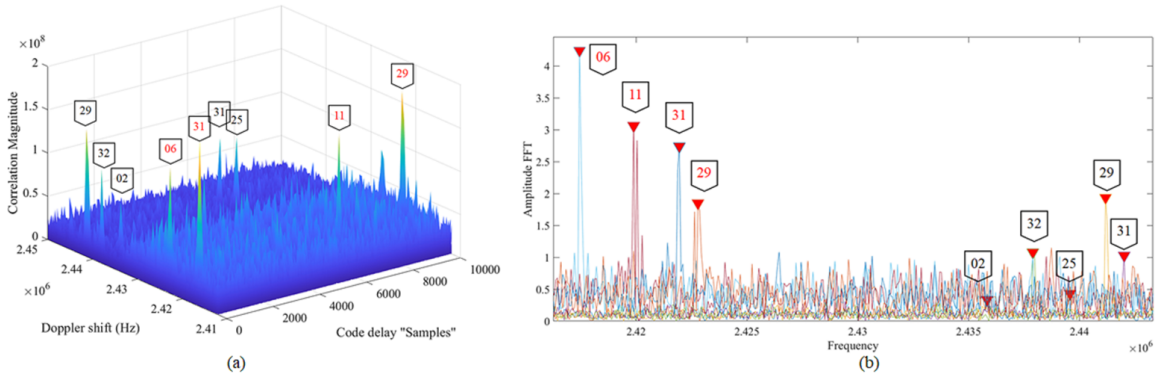

Figure C1.5: Result of the acquisition using ADALM-PLUTO Platform (black font) and Ettus USRP X310 - UBX160 (red font)

In a first test, with a range of 10 KHz in the frequency sweep, we check the failure of the acquisition process with the ADALM-PLUTO device. This failure means that the frequency shift exceeds the frequency search range. Extending the frequency range detects the presence of GPS signals in the range of 20 KHz above the center frequency. This problem requires a considerably longer processing time concerning samples captured with the USRP platform. The USRP platform tested positive acquisitions by keeping a 10 KHz sweep around the frequency 1.57542 GHz. The precision values of the local oscillator of the USRP platform indicate a maximum frequency shift near 2 KHz, enough for the case of the receiver at a stationary position on the ground. These results highlight the limitations of using the ADALM-PLUTO platform in the GNSS signal acquisition process. The frequency shift indicates the need for a calibration process before the acquisition. Figure C1.5 shows the acquisition process results for a block of 1,667 seconds. The USRP platform concentrates the correlation peaks around the center frequency L1, while the Pluto platform experiences a significant shift. In addition, in the carrier detection process through the maxima criterion using the FFT, a low level in the peaks detected with the ADALM-PLUTO platform is verified. For example, the values corresponding to PRN 02 and PRN 25 do not correspond to absolute maximums. These low values of FFT render the correct detection of the carrier frequency impossible. In the case of the USRP platform, all detected satellites are global maximums, indicating a higher value in the Carrier-to-Noise ratio.

The signals are captured in time blocks of 1.667 seconds and processed in off-line mode. The experiment was repeated three times at a uniform time distance of 10

minutes. In each block the acquisition is performed with the two platforms. In order to verify the positive acquisitions of our receiver, it is compared with the GnssLogger v3.0.3.1 application.

Table C1.3: Result of GPS signal acquisition using GnssLogger application, ADALM-PLUTO and Ettus USRP X310 - UBX160

Acquisition Instrument	PRN 02	PRN 06	PRN 11	PRN 12	PRN 16	PRN 22	PRN 25	PRN 29	PRN 31	PRN32
GnssLogger App	x x x	x x o	o o x	x x x	o o o	x o o	o x o	x x x	x x x	x x o
Ettus USRP X310 - UBX160	o x x	x o o	x x x	o x x	o o x	o x x	o o o	x x x	x o o	o x x
ADALM-PLUTO	x o o	o o o	o x o	o o o	o o o	o x o	x o o	x x x	x o o	x o o

The measurements made are described in C1.3. All the satellites detected, with each tool, during the three-time captures are recorded. The symbol x indicates a positive acquisition and o a negative one. In addition, in the case of red color indication, the value of the carrier frequency was not correctly determined due to the low level of the Carrier-to-Noise Ratio. Thus, the ADALM-PUTO platform acquisitions for the PRN 02 and 25 satellites can be considered incomplete. The comparative analysis is focused on the two SDR devices, the GnssLogger application is used to contrast the satellite presence. The mobile device used was the Xiaomi Redmi note 9 Pro smartphone model m2003j6b2g. The GnssLogger application uses Augmented-GNSS, i.e., it combines location information provided by the network serving the mobile device on which it is installed. However, the performance of the pure GNSS receiver, using the USRP device as front-end, presents a number of hits similar to that experienced by the application. When the ADALM-Pluto device is used, the number of hits decreases by almost half to the Ettus device.

C1.5 Conclusion

This article provides a summary of updated GNSS signal characteristics. Identifying new open-civil satellite services reinforces the thesis of using the multi-constellation approach in the localization process. The new operational services indicate a uniformity in the medium access schemes (Glonass L3OC-CDMA).
The proposal of a GNSS SDR receiver, although not new, includes interaction with other localization techniques and presents a flexible modular structure. This scheme generalized the proposal [26], which contemplates specific functions for Front-End and Receiver Software.
The acquisition of real GPS signals demonstrated the limitations of the SDR platforms in the GNSS domain. A fundamental aspect is identified in the accuracy of

the local oscillator. The case of the Ettus USRP X310 -UBX160 platform presents a low-frequency shift that does not affect the GPS identification process. However, the ADALM-PLUTO SDR platform has an accuracy incompatible with the frequency search range implemented by GNSS receivers. Moreover, from the Carrier-to-Noise ratio point of view, some limitations directly impact the correct identification of the carrier. This work is a step forward in the evaluation process of SDR platforms in the GNSS domain. Future work evaluates other platforms with different GNSS constellations and a proposal for full multi-constellation reception.

Conference C2

Opportunistic RSS-based localisation using SDR and ADS-B system

Angel Luis Zuriarrain Sosa, Roberto Alesii, and Fortunato Santucci

In proceedings of
4th URSI AT-RASC, Gran Canaria, 2024

Abstract

Nowadays, a growing increase in air traffic and regulatory directions are leading to using transponders compatible with ADS-B Mode S technology. This scenario means an increase in signal availability that provides helpful information for localisation. Similar to satellite positioning systems, messages sent by aircraft using the ADS-B protocol can be used to estimate the receiver's position. This work is based on a flexible and configurable Software Defined Radio (SDR) receiver that allows for evaluating the effect of the sampling frequency parameter on localisation. The RSS estimates the distance between the aircraft and the receiver using channel models from experimental measurement campaigns. The experimental results demonstrate an improvement in error distance estimation using techniques based on statistical analysis, outlier filtering, classification of received messages, and changing the SDR platform performance.

C2.1 Introduction

According to EUROCONTROL forecasts for 2023-2028 [56], after the drastic drop in air routes in 2020-2022 due to COVID-19, a full recovery (2023) and a further increase is foreseen for the next few years. New programs and regulations have been proposed to deal with this increase in air traffic. In Europe specifically, the SESAR (Single European Sky ATM Research) program and its evolutions (i.e., SESAR 3 Joint Undertaking) [57] stand out, comprising the development of the most advanced technological solutions to manage conventional aircraft, drones, air taxis and vehicles flying at higher altitudes. From a regulatory point of view, European directives set the performance and interoperability requirements for Single European Sky Surveillance [58], which in this case focus on the use of operational secondary surveillance radar transponders compatible with ADS-B (Automatic Dependent Surveillance Broadcast) extended spontaneous signals. Annex II of Implementing Regulation (EU) No. 1207/2011 sets Mode S level 2 as the minimum operating capability of a surveillance secondary radar (SSR) transponder.

The future scenario is clear: more congested airspace, efficient localisation/surveillance emerging technologies and increased onboard communication capabilities. In particular, ADS-B technology shall be used to localise and track devices that can receive the aircraft's transmitted signal. This localisation approach is opportunistic because it uses RF signals and information available in the ether but is not initially designed for the receiver's localisation purpose. The algorithms proposed in this scenario are mainly based on multilateration (MLAT) and use time difference measurements (TDOA) [34], time of arrival (TOA), or distance measurements from RSS Vs. Distance models [35, 36].

The constant evolution of Software Defined Radio (SDR) makes it the preferred radio communications system for implementing RF receivers cost-effectively, flexibly, and rapidly deployable. Platforms such as USRP from Ettus Research and ADALM-PLUTO from Analog Devices Inc. have been used to implement ADS-B receivers. Although the most widely used HW for ADS-B is the RTL-SDR [34] (used in *Flightradar24*, *OpenSky Network*, and *FlightAware*), it has limitations in terms of bandwidths that impact the functionality of the receiver.

This work addresses the challenge of determining receiver position from the opportune decoding of ADS-B signals using SDR. The MLAT algorithm uses the distance determined from the RSS-based measurements of the received signal. An RSS-distance model is generated experimentally from extended periods of measurements

in the field. Furthermore, an evaluation of the impact of the receiver bandwidth on the distance estimation, localisation, and performance of the ADS-B receiver is performed. For this reason, Section C2.2 describes the methodology followed in implementing the receiver and the localisation algorithm. Also, Section C2.3 presents the experimental activities carried out to validate the reception of ADS-B signals and the channel model obtained. Section C2.4 describes the results obtained, and Section C2.5 concludes this article.

C2.2 Methodology and problem formulation

The idea of determining the spatial coordinates of an object using information from the received ADS-B signal is addressed in three fundamental steps. The first step is to use power measurements of the received signal to generate a model to determine the distance between the transmitter (aircraft) and the receiver. The next phase consists of solving a system of equations, one equation for each aircraft available in a given time window, using MLAT algorithms. The last step is the development of the SDR-based receiver to detect, decode and extract radio signal information of ADS-B messages.

RSS Channel Model

The most widespread model used to describe radio signal propagation is the Log-Normal Shadowing model, a generalisation of the Friis free space equation. Equation (C2.1) shows the simplicity of the model, where RSS_0 is a constant term which takes one of the three possible values of the transmission power of airborne transponder, d is the distance between the transmitter and receiver, β is the path loss exponent, η is a zero-mean Gaussian random variable, and N indicates the number of messages considered for average calculation.

$$RSS(dBm) = RSS_O(dBm) - 10\beta log(d) + \eta \tag{C2.1}$$

$$\hat{d} = 10^{\frac{RSS_O(dBm) - RSS(dBm)}{10\beta}} \tag{C2.2}$$

$$RSS(dBm) = \frac{1}{N}\sum_{n=1}^{N} RSS_{n,t} \tag{C2.3}$$

The channel model assumes that the noise has a Gaussian distribution. When a variable has a Gaussian distribution, its mean value equals its averaged value. However, in practical conditions, where some outliers may exist, it is better to use the mean

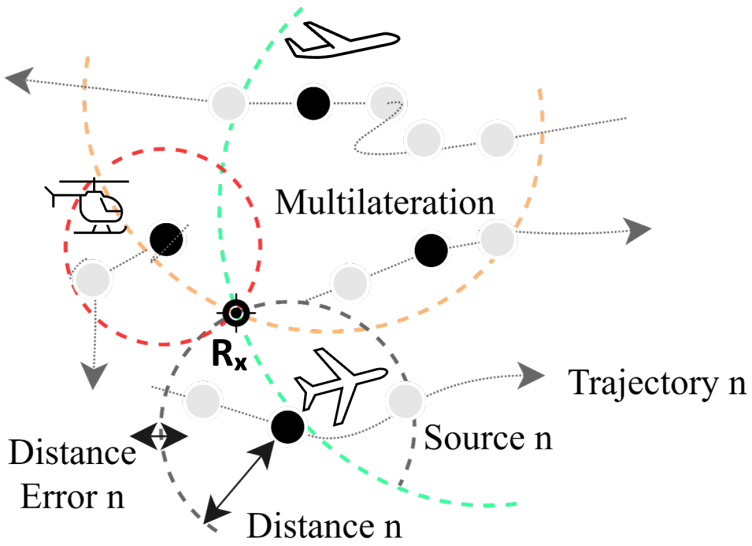

Figure C2.1: MLAT in the ADS-B scenario

value because it is more robust to outliers. In Equation (C2.2), the value of $\hat{d}$ is estimated from the mean RSS measured on a set of messages from the same airborne in a given time window.

Multilateration algorithm

The MLAT is based on the solution of a system of equations where the solution corresponds to the receiver coordinate. In our case, the estimated distance between the transmitter and receiver is determined based on a model relating to the RSS measured at the receiver. The application of a statistical model leads to errors in distance estimation. These errors in distance estimation lead to the use of optimisation methods in selecting the correct solution.

Given the n reference points $P1(X_1, Y_1, Z_1)$, $P2(X_2, Y_2, Z_2)$, $...Pn(X_n, Y_n, Z_n)$ and the range measurements $d_1, d_2, ..., d_n$. The solution of the coordinates (x, y, z) using n points is equivalent to solving a quadratic system of equations with $n-th$ expression:

$$(x - X_n)^2 + (y - Y_n)^2 + (y - Y_n)^2 = d_n^2 \tag{C2.4}$$

Manipulating (C2.4) we can write:

$$(x^2 + y^2 + z^2) - (2xX_n + 2yY_n + 2zZ_n) = d_n^2 - (x_n^2 + y_n^2 + z_n^2) \tag{C2.5}$$

Equation (C2.5) allows to compose a linear system equation with the form $\mathbf{A}x = \mathbf{b}$. Applying Least Squares (LS) methods with the constraint $x \in \{(X_0, X_1, X_2, X_3)^T \in$

$\mathbb{R}^4/X_0 = X_1^2 + X_2^2 + X_3^2\}$, the solution can be written as follows: $\hat{x} = (A^T A)^{-1} A^T b$. However, solutions offered by this method depend only on the matrix **A** and the vector **b**, which depends only on the coordinate values received in the ADS-B messages that are opportunely decoded by the receiver. Finally, the selected solution is the one that minimises the sum of squared distance error.

C2.3 SDR-based ADS-B receiver

There are two types of ADS-B downlink signals (centred at 1090 MHz) in mode S, the short response (56 bits) and the extended response (112 bits), which correspond to the short and long interrogations of the secondary surveillance radar (SSR). The bit duration is 1 microsecond and uses Pulse-Position Modulation (PPM). All Mode S responses start with a fixed preamble of 8 symbols (duration 8 microseconds) and continue with a long or short bit sequence (payload). Civil Aeronautics ADS-B messages after the 8-microsecond preamble continue with the binary sequence 10001, corresponding to the Downlink Format (DF) field. The next three bits correspond to the Transponder Capability field, followed by the ICAO code, which identifies the aircraft and is 24 bits long. The short messages continue sequentially with the 24-bit Parity/Interrogator (PI) field, completing the 56 bits. However, before concluding with the PI, the long messages contain a block message (ME) that contains information concerning the aircraft's position, altitude, speed, heading and status. In particular, the extended messages are helpful for localisation because a stream of information regarding the aircraft's position/speed/altitude is available. Since the distance estimation method uses power-averaged values, a more accurate value will depend on analysing a significant number of messages from the same aircraft. It is essential to detect as many messages as possible to reduce the analysis time window. Theoretically, the frequency of extended position and velocity messages is 2 Hz, and the total number of messages (on average) exceeds six messages per second.
The received data is split into In-phase and Quadrature digital components (I&Q) when using SDR platforms with Zero-IF architectures as Front-End. Although the frequency is centred at 1090 MHz, frequency shifts may be experienced due to errors in the SDR device's local oscillator (LO) [11], the airborne transponder or the Doppler effect introduced by the speed difference between transmitter and receiver. Therefore, the first blocks of our receiver are dedicated to the correct centring of the band, the elimination of the DC component and the filtering of the signal to increase the Signal to Noise Ratio (SNR). The receiver processes the baseband samples to detect

preambles that indicate the start of an ADS-B message. The correlation process uses pre-coded NRZ sequences to obtain a higher sensitivity in preamble detection. Detecting a maximum exceeding a specific threshold means, with a high probability, the presence of an ADS-B message. The threshold is calculated based on living metrics, where the (time) segment or signal frame to be analysed is statistically evaluated.

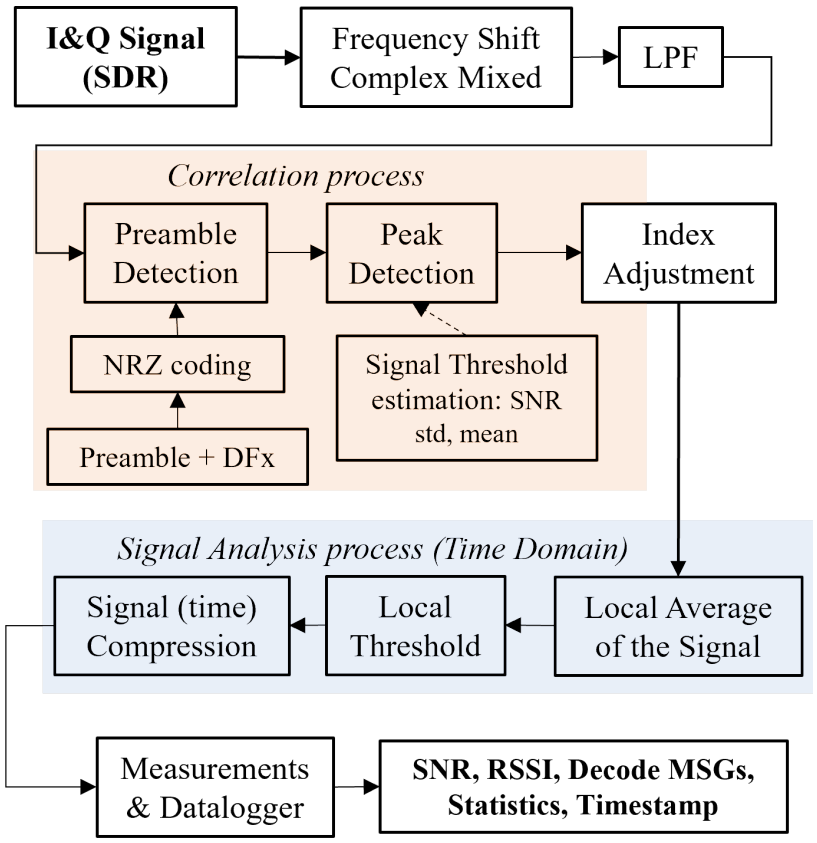

Figure C2.2: SDR-based ADS-B receiver

Figure C2.2 highlights the blocks working in the frequency and time domains. Once the preambles are identified, their position in the time domain is determined, and false positives are discharged. After identifying message positions of the messages in the time domain, a signal analysis is performed to validate the results obtained in the frequency domain. The validation also confirms that the preamble's "high" pulses are within 3dB of the reference power. The parser block (*MSGs Decode*) is responsible to decoder the information. At the end, the data is stored in a structure that includes the following fields: RSS (before filtering and after filtering), decoded message information (Speed, Position, Altitude), internal buffer index, timestamp, SNR and the identification of the aircraft through the ICAO code.

C2.4 Results

The experimental tests were conducted outdoors at the University of L'Aquila (Italy). The scenario is characterised by low air traffic and mountainous geography, which leads to a lower reception of ADS-B messages.
The SDR platform was the USRP X310 (UXB160 daughterboard) configured with 36 dB of RF gain centred to 1090 MHz. The first experimental activities aimed to analyse the performance of the ADS-B receiver, and for this reason, captures were performed at different sampling rates. Table C2.1 demonstrates the impact of using

Table C2.1: ADS-B-based SDR receiver performance

Parameters	2 MSps	4 MSps	10 MSps	20 MSps	50 MSps	100 MSps
No.Airbornes detected [U]	6	7	7	7	9	9
No.Messages passed CRC [U]	957	1192	1315	1862	2934	3382
% position messages	36.57	35.82	35.51	35.66	36.37	36.56
Max. distance [Km]	182.57	200.43	202.61	208.33	210.41	210.57

a higher-performing receiver. An increase in sampling frequency impacts the number of correctly detected messages (passing the CRC), the sensitivity of the receiver and hence, the ability to detect aircraft at a greater distance. Receiver performance is also influenced by geography and atmospheric conditions. The data in Table C2.1 corresponds to an acquisition window of 120 seconds (2 min). The messages are lower than expected (6 x number of aircraft x 120s) since they are not visible to the receiver during the whole interval.
The study of the typology of the messages made it possible to verify both the correct functioning of the receiver and the theoretical nominal frequency described in previous sections. In particular, the messages received according to their typology were distributed as follows: approximately 5% corresponded to identification messages (TC: 1-4); 36% corresponded to messages with position information [1] (TC: 9-18); approximately 37% of the messages indicated the speed (TC: 19) and 22% of the messages indicated airborne status information. Figure C2.3 shows one of the strategies followed to minimise the errors in calculating the distance from formula (C2.2). In this work, we use three models that respond to the three possible values of authorised power in commercial transponders: 500, 250, and 125 W. In addition, Figure C2.3 identifies with different colours the measurements taken from the top or bottom interface of the antennas of each detected aircraft. Specifically, the filtering

[1] All measurement campaigns conducted confirm that the altitude value is obtained from barometric sensors.

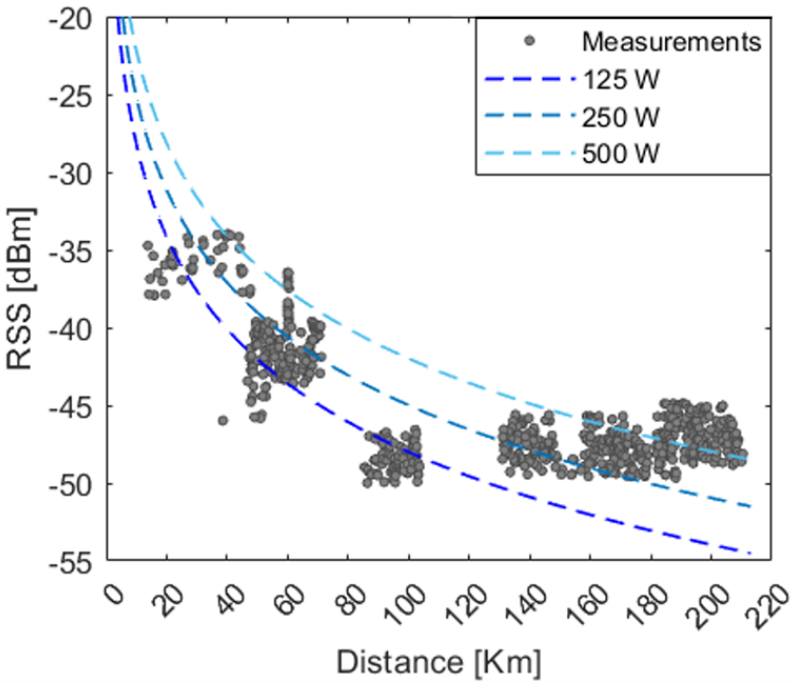

Figure C2.3: RSS Vs. Distance. Three different curves corresponding to aircraft transponder power: 125, 250 and 500 W

strategies to increase the localisation accuracy focus on excluding spurious measurements from the same reflected signals. In many cases, messages from the opposite antenna interface of the same aircraft are detected and decoded. This undesired effect is solved by classifying and sorting based on the timing difference between messages. In addition, highly attenuated measurements that are affected by the geometry of the dipole radiation pattern are excluded. Figure C2.4 shows the final results and the impact of the techniques applied to mitigate errors in distance calculation. In all cases, we start with a time window of two minutes where the statistical methods for eliminating spurious values are applied. The blue curve represents the results obtained from analysing/applying a dedicated model depending on the aircraft classification and, therefore, the type of transponder used. In the second case (yellow curve), the transmission interface identification process is added, thus eliminating values coming from the second antenna, further increasing the accuracy of the measurements. The last case includes all the techniques described above and tries to identify (offline) the outliers caused by the toroidal radiation pattern of the dipole antenna.

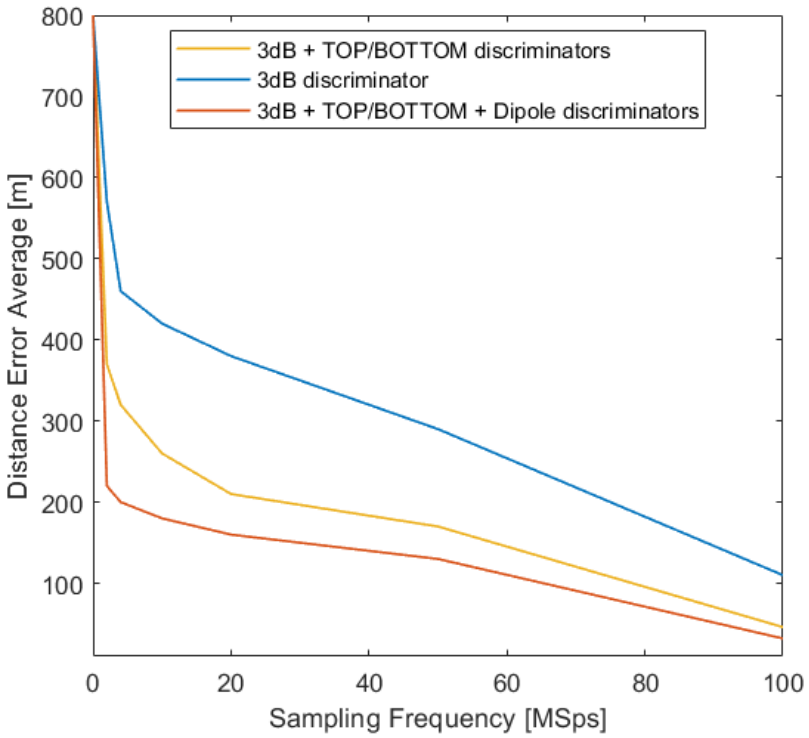

Figure C2.4: Impact of applied techniques and sampling frequency on distance error

C2.5 Conclusions and future directions

This paper presents an SDR-based solution for opportunistic localisation using ADS-B Mode S signals. The methodological approach based on experimentation to describe the channel can be extended to other application fields, exploiting the SDR paradigm's modularity and flexibility. From the point of view of the results in terms of errors in the localisation domain, it constitutes a starting point for future solutions closer to a real-time approach.

Regarding the power analysis, some outliers are registered and must be filtered out before the distance calculation. In most cases, messages with a very distant RSS are caused by reflection occurring at the non-visible antenna interface of the airborne antenna. The solution proposed in this work consists of a time-based filtering analysis. Both aircraft interfaces (Top/Bottom) transmit messages in alternate mode every 0.5 seconds, with a maximum error of 0.1 seconds. This feature allows us to correctly group the information from each interface and promptly discard the values that introduce a significant error. Another element that has been detected and is proposed to be addressed in future work is the error introduced by the antenna radiation pattern itself, which can be attenuated by using gain masks depending on the aircraft's

orientation.
The proposed techniques to improve localisation by timely identifying the phenomena affecting the ADS-B signals are excellent, showing a considerable improvement in the results (8.6 Km) found in the literature. Maintaining distance errors in the model, on average, less than 800 metres, is a remarkable result in the context addressed.

Conference C3

SDR-Based Distributed System for Mobile Communication Network Monitoring and Support

Alex Piccioni, Angel Luis Zuriarrain Sosa, Roberto Alesii, and Fabio Graziosi

Accepted to

Next-Generation Multimedia Services at the Edge: Leveraging 5G and Beyond - co-located with ISCC June 26, 2024 // Paris, France

Abstract

Today mobile communications have evolved from merely connecting devices to enabling a wide range of heterogeneous services, particularly with 5G networks. The rise of multimedia services and digital content consumption is driving the proliferation of wireless devices, which in turn is increasing the complexity of resource management. In this scenario, a distributed system could be the most effective way to manage this complex environment for monitoring and support purposes, especially through Software Defined Radio (SDR) technology. This paper introduces a distributed SDR-based system designed to perform various services to monitor and support any mobile communication systems. After presenting the overall design of the proposed system, this work will describe and test a prototype that can execute a monitoring service for spectral analysis, interference detection, and localization.

C3.1 Introduction

Over the past decades, wireless communications have seen a significant evolution and several services embraced mobile networks, particularly with 5G and the forthcoming 6G networks. With respect to the previous generations, 5G and beyond offer dynamic resource allocation to execute a wide range of services. In line with the recommendations of the International Telecommunication Union (ITU) [59], three main groups have been defined to categorize all the services enabled by 5G networks:

- **Enhanced Mobile Broadband (eMBB):** This category includes services with enhanced performance and high data rate.
- **Ultra-Reliable and Low-Latency Communications (uRLLC):** These services cater to mission-critical applications requiring low latency and high reliability.
- **Massive Machine Type Communications (mMTC):** This category deals with high-density scenarios, facilitating Industry 4.0 and massive Internet of Things (IoT) applications.

The flexibility and diversity of services offered by 5G and future systems like 6G, make it a versatile service-enabler. Various applications have been explored in literature [60], such as applications based on audio spatialization for cultural heritage services in 5G networks [61]. Additionally, Network Function Virtualization (NFV) concept includes the software-oriented approach, where specific network equipment are replaced with Virtual Machines (VMs).

In this sense, the convergence of multimedia and communication technologies has further facilitated the integration of multimedia services within mobile networks, leveraging the enhanced performance and capabilities offered by this emerging technology. This rise of multimedia services is driven by the exponential growth of digital content consumption and the increasing demand for immersive experiences [62]. High-definition video streaming and emerging technologies such as Virtual Reality (VR), Augmented Reality (AR), and Mixed Reality (MR) have reshaped consumer behavior [63], making mobile devices indispensable for accessing multimedia content.

Nevertheless, this represents a challenge for mobile network operators who must provide an infrastructure capable to meet the increasing bandwidth, low latency, and reliability demand. The proliferation of wireless devices in everyday scenarios has led to an increase in connected devices per user. It is estimated that in massive IoT

scenarios there could be up to one million devices per square kilometer [64], a number expected to rise further. This growth of connected devices is closely tied to the use of devices for multimedia purposes [65] and preventive monitoring in various domains. For instance, sensor nodes connected to 5G networks are deployed in constructions to monitor their structural integrity and avert disasters, such as earthquakes [66]. Another relevant scenario is the use of 5G for disaster management and early earthquake warning systems [67].

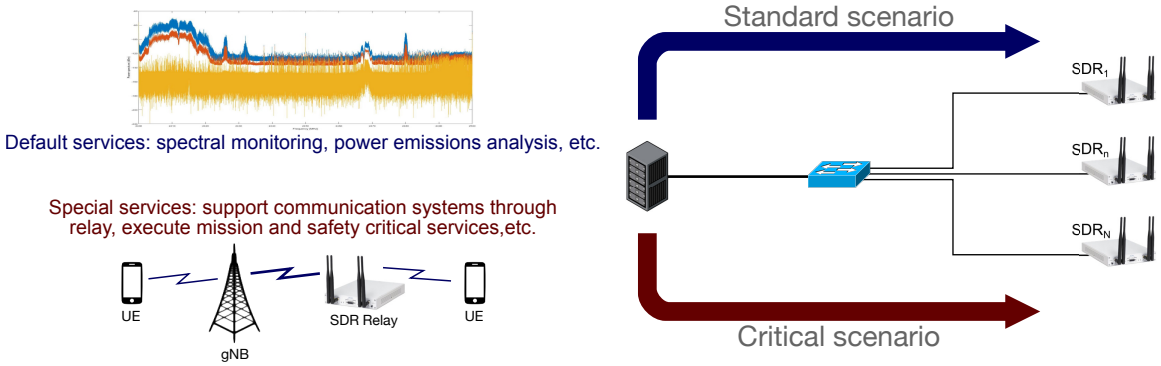

Figure C3.1: Simplified representation

Different unwanted effects can be considered, such as the presence of interference effects and the resource demand increase, just to name a few. Spectral resource usage is becoming a critical issue, especially due to spectrum saturation as a consequence of the increasing radio resource and bandwidth demand. [68] has extensively studied these aspects, and the authors described the challenges due to spectral usage while addressing the need for applications based on spectral monitoring services. Moreover, the requirement for spectral monitoring systems is becoming necessary, purposing to reduce and mitigate interference while evaluating the radio resource usage inside a certain area, eventually supported by techniques to detect and localize the interfering sources. The research community has well investigated this task. [69] proposes signal detectors taking advantage of Software-Defined Radio (SDR) technology [70–72]. In this case is clear the requirement of services that allow to extract information alongside detection techniques to understand the interfering source, as in [73] where a signal recognition technique focused on modulation classification has been described.

In this background, this work aims to present a distributed system for the implementation of real-time heterogeneous services to support modern mobile networks for multimedia applications exploiting SDR technology, whose interoperability and flexibility can be an added value. A prototype is described and tested in a simplified scenario focusing on spectral monitoring for interference detection and localization.

The rest of the paper is organized as follows: Section C3.2 provides a general description of the proposed system and practical scenarios, Section C3.3 presents the prototype and its assessment, while some conclusions are reported in Section C3.4 with the future development.

C3.2 SDR-based distributed system

The proposed system falls within the broader landscape of multimedia services. In this context, the system aims to develop a distributed network of SDR devices capable of executing heterogeneous services across various scenarios, thereby contributing to the advancement of communication technologies in diversified environments.

The core idea revolves around creating a distributed system based on SDR devices that can dynamically adapt to the requirements of various multimedia applications. SDR technology offers several advantages, including enhanced performance, flexibility, and scalability. Moreover, SDR platforms feature reconfigurable hardware, allowing them to be programmed in either hardware or software domains based on performance needs. A remote controller facilitates seamless switching between different sets of services, as depicted in Fig C3.1.

The distributed nature of the system, coupled with an ad hoc network connecting SDR devices and controllers, enables diversified coverage, from extensive coverage across large areas such as entire cities to small areas like buildings. This infrastructure can monitor and support mobile communication systems, applying different policies tailored to specific sub-areas as needed.

Services offered by the proposed system can be categorized in two scenarios, standard and critical (Fig. C3.1). Standard scenarios encompass routine operations of mobile communication systems, where the focus lies on resource optimization and efficiency to ensure the execution of multimedia services. Monitoring services, such as power emission and spectral analysis, play a crucial role in ensuring optimal resource utilization. Additionally, sensing techniques can be integrated to gather environmental data for activities like activity detection and movement analysis, particularly beneficial in energy-saving initiatives within 5G massive MIMO environments.

In contrast, critical scenarios represent exceptional circumstances where network operability may be compromised, such as during disasters or emergencies. In such cases, the proposed system serves to support mobile communication networks. Services like repeater functionality, employing relay methods or Radio Access Network (RAN) element replacement (i.e. gNBs), ensure uninterrupted connectivity for users.

Furthermore, specialized PHY or MAC functionalities can be integrated to support ad hoc tasks, such as disseminating warnings during disasters, i.e. earthquakes. The possibility to remotely control the SDR devices enables multitasking across different sub-areas, facilitating hybrid scenarios where critical services coexist with standard operations.

C3.3 Prototype and testing

While writing this paper, work is ongoing to develop a prototype of the proposed system. The main objective is to create a default service that the system can adopt in standard situations. Consequently, the initial focus has been on implementing services for spectral monitoring, interference detection, and localization. To ensure these functions work correctly, testing has been performed with a static interfering source in a controlled setting.

C3.3.1 Prototype analysis

In today's mobile communication landscape, which is increasingly defined by NFV and software-oriented approaches, the proposed system aims to leverage virtualization. Consequently, the system controller has been developed using a VM that can be run on various devices, including computers and servers. The system, programmed with LabVIEW software, includes three identical high-performance SDR Universal Software Radio Peripheral (USRP) devices by National Instruments (NI), specifically the NI USRP 2954R. These devices feature a user-programmable FPGA, the Xilinx Kintex-7, and offer capabilities such as 2x2 MIMO configuration with four I/O interfaces (two Tx/Rx and two Rx-only), a frequency range from 10 MHz to 6 GHz, a sample rate up to 200 MSps (maximum 100 MSps for a single channel), a maximum instantaneous bandwidth of 160 MHz, a 16-bit Digital-to-Analog Converter (DAC), and a 14-bit Analog-to-Digital Converter (ADC).

At this initial stage of development, system management and service processing are handled in the software domain. The system controller sets all necessary parameters for the SDR devices, such as sample rates and carrier frequencies, and then initiates the service. The SDRs then acquire signals and transfer the sampled data to the controller, which processes them for spectral monitoring, interference detection, and localization services. Working on software domain guarantees low delay for switching from one service to another one since it results from the performance of the controller. Future development plans include offloading some functionalities to the FPGA for

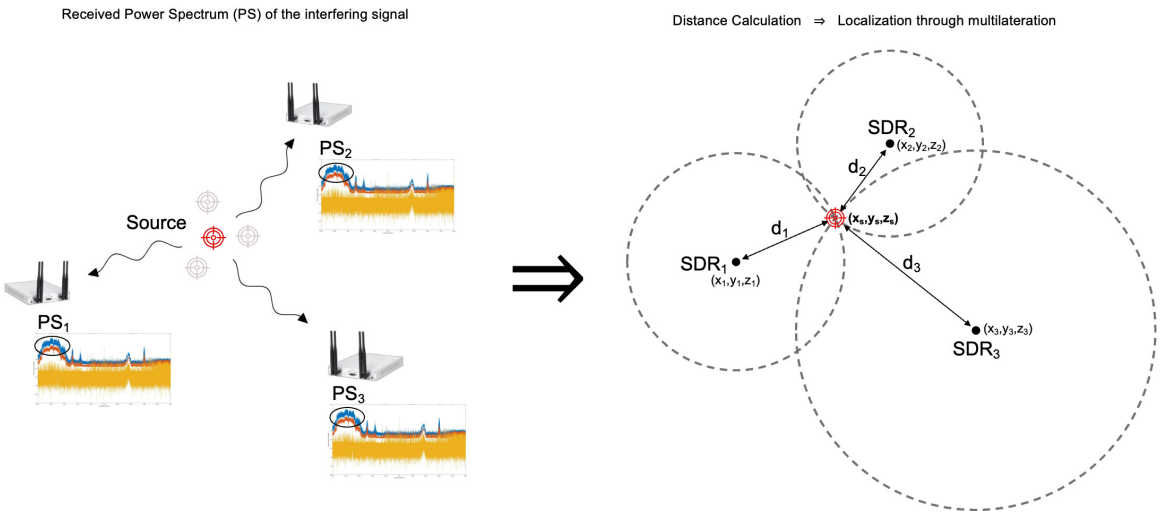

Figure C3.2: Default service representation: from spectral analysis to interference detection and localization

hardware-oriented processing, especially for low-latency services. This will include an increased switching delay since it depends also on the time required to transfer the FPGA bitstream toward the SDR device, which however can be reduced by a local storage of FPGA bitstreams, in addition to the time required for the FPGA initialization. However, this delay strongly depends on the SDR performance.

Once the system is activated, the three NI USRP 2954R devices capture signals from the spectrum under analysis, encapsulate the samples, and transfer them to the system controller. In this initial phase, the controller, running as a VM on a PC, establishes direct connections with each SDR device via a 10Gbps fiber link, allowing for a maximum sample rate of 200MSps per device. However, as the system moves to a dedicated network setup in future iterations, a balance between the controller's link capacity and each SDR device's maximum sample rate must be addressed.

The controller operates under the assumption of RF synchronization among the three devices. While the level of synchronization required depends on the specific service, this assumption remains valid for the prototype. The first analysis performed by the controller is spectral monitoring, where the Power Spectrum (PS) is processed for each sample frame acquired by the SDR devices. The controller computes the maximum, average, and minimum PS for each SDR device over multiple acquisitions. This preliminary analysis aids in assessing the spectrum status, identifying consistently present signals (minimum PS), sporadic signals (maximum PS), and providing insights into the frequency and amplitude variability of signal occurrences (average PS). Specifically, the average PS offers a view of the signal's presence in the acquired

frames: signals closer to the minimum PS may be sporadic, while those nearer to the maximum PS are likely more frequent, provided their amplitude shows low variability.

Successively, the detection algorithm through z-score is performed to detect the interfering signals in the spectrum under analysis. Considering a generic data set s, its z-score is:

$$z = \frac{s - \mu_s}{\sigma_s} \tag{C3.1}$$

where μ_s and σ_s are the average value and standard deviation of s, respectively. Thus, the z-score standardizes the data set distribution, such that it has zero mean and unitary standard deviation. Exploiting Eq. (C3.1), the detection algorithm compares the actual PS and the average PS, such that a new signal is detected through the comparison of its z-score

$$z^* = \frac{|PS - PS_{avg}| - \mu}{\sigma} \tag{C3.2}$$

with the threshold λ [74], which is then used as a comparison to enable two possible situations resumed in Eq. (C3.3):

$$\begin{cases} |z^*| < \lambda & (H_0) \\ |z^*| \geq \lambda & (H_1) \end{cases} \tag{C3.3}$$

If the absolute value of the z-score is equal or greater than λ (i.e., case H_1), it corresponds to a point distant at least λ standard deviations from the mean value, thus a signal has been detected, otherwise there is no detection (H_0). A trade-off is required to reduce the false alarm rate; it can be done by choosing a higher λ, with the drawback of a higher miss rate in the detection. In this step, the focus is on narrowband signals, since the detection algorithm based on the z-score cannot guarantee the detection of wideband signals.

The signals detected by each SDR device are compared to guarantee that the signals under consideration are just the ones simultaneously detected by all devices. This allows the integration of a technique to localize a static signal source. A range-based localization algorithm has been considered to evaluate the source-SDR receivers' distances. In general, localization employs two steps, which are the distance calculation and the position solution. The first relies on a model derived from the Friis formula experimentally adjusted. The following is a simplified formula for the distance and PS relationship:

$$PS(d) = PS_0 - \beta \log_{10}(d) \tag{C3.4}$$

with $PS(d)$ as the PS detected at distance d (in dBm), PS_0 as the one-meter distance PS, and β as path loss parameter. PS_0 and β can be determined empirically and strongly depend on the environment and the channel.

Multilateration technique has been adopted for the localization and is based on a system of equations whose solution, (x_s, y_s, z_s), is the source coordinate [75], as shown in Fig. C3.2. Then, localization is performed by solving the following non-linear equations system:

$$(x - x_n)^2 + (y - y_n)^2 + (z - z_n)^2 = d_n^2 \tag{C3.5}$$

with $n = 1, 2, 3$, and (x_n, y_n, z_n) are the $n - th$ SDR coordinates, and d_n its distance from the source evaluated through Eq. (C3.4). Manipulating Eq. (C3.5), the result is

$$\begin{aligned} &(x^2 + y^2 + z^2) - (2xx_n + 2yy_n + 2zz_n) \\ &= d_n^2 - (x_n^2 + y_n^2 + z_n^2) \end{aligned} \tag{C3.6}$$

Eq. (C3.6) denotes the n-th term of an equation system represented in matrix form as $A\mathbf{s} = \mathbf{b}$, where A is the coefficient matrix, $\mathbf{b}$ is a constant vector, and $\mathbf{s}$ represents the solution vector with coordinates (x_s, y_s, z_s) [76]. It's crucial to note that the values of A and $\mathbf{b}$ are provided by the SDR devices, along with the estimated distance d_n and their respective coordinates (x_n, y_n, z_n).

C3.3.2 Testing operation

This paper aims to provide readers with a fundamental understanding of the proposed SDR system and the developed prototype. Instead of investigating the detection and localization algorithm performance or pursuing algorithm optimization, the emphasis is on the validation of default service. However, preliminary tests for relay operation such as amplify-and-forward relay have been described and tested in [77], evaluating the minimum delay introduced by different SDR platforms, ranging from 0.5μs up to 1.5μs. The $2.4 - 2.5$GHz ISM radio band has been selected, acquiring the spectrum using 100MSps sample rate and 100MHz bandwidth. A static interfering source has been considered to validate the prototype in addition to test the effectiveness of the spectral monitoring application based on the maximum, average, and minimum PS. The interfering source consists of a frequency-modulated signal, i.e. chirp, with 100μs duration and 10MHz bandwidth transmitted with a carrier frequency of 2.45GHz, which has been selected due to the reduced power spectrum fluctuations with respect to other narrowband signals.

The SDR devices have been positioned in the scenario illustrated in Fig. C3.3 (red), while five different positions (blue) as been selected for the interfering source, assuming a static environment neglecting the fluctuations due to the channel. The

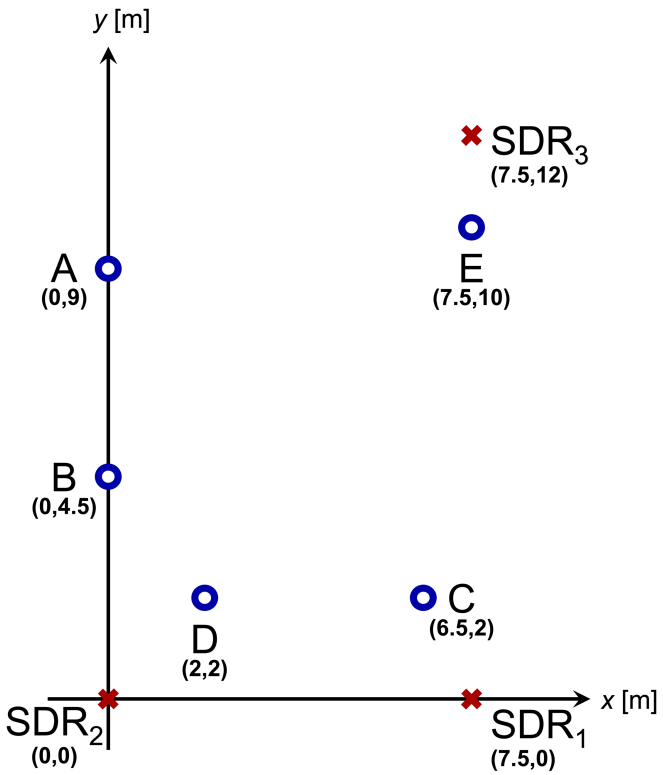

Figure C3.3: Adopted for the prototype test with three SDR devices (red cross) and five interfering source positions (blue circle)

localization has been considered on the xy plane with $z = 0$ due to the limited number of SDR devices. After the system is on, the interfering source can be detected in the PS of each device with different power levels depending on the distance, as in Fig. C3.4-a. Additionally, other signals are visible in the spectrum under consideration, e.g. Wi-Fi. Subsequently, the source has been detected through the z-score as per Eq. (C3.2), both with and without the source, and then compared with a threshold $\lambda = 3$, whose value has been considered acceptable for this test. Fig. C3.4-b shows the static interfering source during the test, which has been detected through the comparison of the detection of each SDR device. This detection strongly depends on the environment in which the SDR devices are placed. In large environments, a different policy is needed since not all the SDR devices will be able to detect the same interfering source, and this will be part of future enhancements along with the implementation of wideband interference detection.

The position of the source is obtained using the algorithm mentioned above. Prior to real-time testing, a measurement campaign is necessary for the considered localization model to evaluate the general behavior of the channel, which will influence

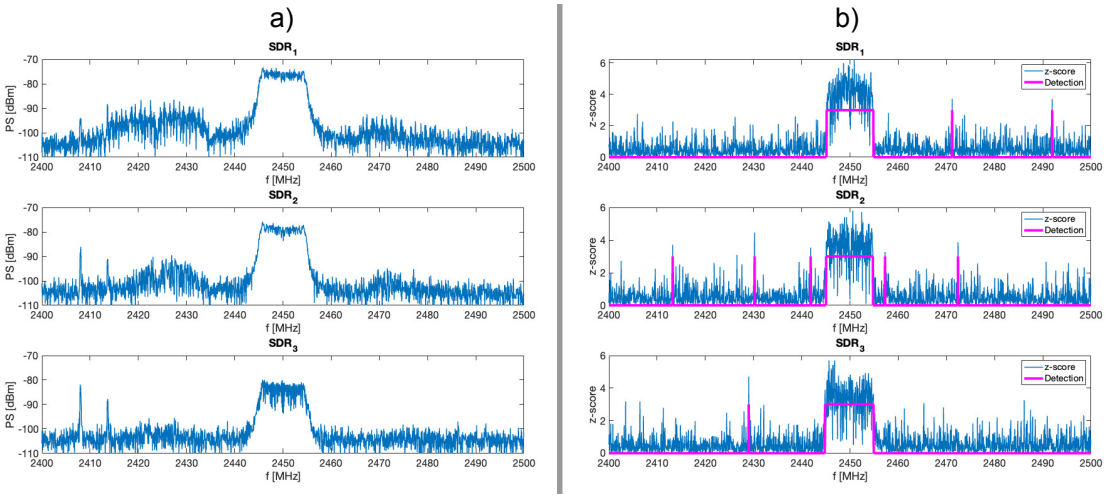

Figure C3.4: The Power Spectrum frame acquired for the $2.4-2.5$GHz range by each SDR device with the chirp interfering source positioned at location C is illustrated in **a**. Additionally, **b** displays the corresponding z-score absolute value of the difference between the PS with and without the interfering source, with the detection overlapped for each SDR device

the localization error. Following the measurement campaign, localization has been tested by placing the interfering source in five different positions (Fig. C3.3) using empirical parameters $PS_0 = -64.96dBm$ and $\beta = 18.96dB$. Table C3.1 reports the estimated positions of the interfering source. Due to the complexity of the environment, the empirical parameters adopted for the test could introduce errors into the model, thereby increasing the localization error. This discrepancy is evident in the final column of Table C3.1, where the average absolute error is 2.54 m for the x variable and 3.25 m for the y variable. It is important to highlight that these errors are common challenges associated with range-based localization algorithms, which are not the primary focus of this work but just a part of it. Thus, a more precise measurement campaign is required to mitigate these errors. Hence, addressing these issues will be part of future testing operations, possibly through the adoption of more accurate localization algorithms that consider parameters beyond PS, such as time of arrival and angle of arrival.

C3.4 Conclusions

With the advent of modern communication technologies such as 5G and 6G along with the increasing interest in the integration of multimedia services, this study has introduced a novel distributed system leveraging SDR to support mobile networks.

Table C3.1: Source localization results

Positions	**Reference [m]**	**Localized [m]**	**Error [m]**
A	$(0, 9)$	$(3.16, 6.28)$	$(+3.16, -2.72)$
B	$(0, 4.5)$	$(2.94, 6.42)$	$(+2.94, +1, 92)$
C	$(6.5, 2)$	$(4.19, 5.95)$	$(-2.31, +3.95)$
D	$(2, 2)$	$(3.66, 5.92)$	$(+1.66, +3.92)$
E	$(7.5, 12)$	$(4.87, 6.28)$	$(-2.63, -3.72)$

The system offers flexibility, allowing for the execution of various services, ranging from fundamental tasks like power emission analysis and spectral monitoring to critical services required in unconventional scenarios where standard operations may not suffice. In this context, an initial prototype has been described, capable of performing a default service centered on spectral monitoring, interference detection, and localization. The prototype, based on three NI USRP 2954R SDR devices, has been evaluated using an interfering source in a limited scenario to validate the system's effectiveness.

Further steps will look in several directions. The primary focus will involve the implementation of new services, starting from 5G relay in critical scenarios, and progressing towards offloading certain functionalities onto the FPGA. Besides, efforts will be directed towards enhancing the spectral monitoring service by integrating signal recognition techniques such as automatic modulation classification, as well as refining the interference detection and localization algorithms currently employed.

Journal J1

OBU for accurate navigation through sensor fusion in the framework of the EMERGE project

Angel Luis Zuriarrain Sosa, Valeria Ioannucci, Marco Pratesi, Roberto Alesii, Carlo Albanese, Francesco Valentini, Elena Cinque, Alessio Martinelli and Michele Brizzi

In Journal
Applied Sciences 14, no. 11: 4401, 2024

Abstract

With the development of advanced driver assistance systems (ADAS) and autonomous vehicles (AV), recent years have seen an increasing evolution of onboard sensors and communication interfaces capable of interacting with available infrastructures, including satellite constellations, road structures, modern and heterogeneous network systems (e.g., 5G and beyond) and even adjacent vehicles. Consequently, it is essential to develop architectures that cover data fusion (multi-sensor approach), communication, power management, and system monitoring to ensure accurate and reliable perception in several navigation scenarios. Motivated by the EMERGE project, this paper describes the definition and implementation of an On Board Unit (OBU) dedicated to the navigation process. The OBU is equipped with the Xsens MTi-630 AHRS inertial sensor, a multi-constellation/multi-frequency Global Navigation Satellite System (GNSS) receiver with the u-blox ZED-F9P module and communication interfaces that afford access to the PointPerfect augmentation service. Experimental results show that GNSS, with corrections provided by augmentation, affords centimetre accuracy, with a Time To First Fix (TTFF) of about 30 s. During the on-road tests, we also collect: the output of fusion with inertial sensor data, monitoring information that assess correct operation of the module, and the OBU power consumption, that remains under 5 W even in high-power operating mode.

J1.1 Introduction

Urban projections until 2050 show an increased concentration of population in urban areas. This scenario presents challenges in terms of mobility and traffic that aim to efficiently and sustainably manage resources to increase the quality of life. There are many approaches behind the concept of smart mobility [78–80]:

- the use of technologies that provide helpful information (IoT, sensors, network);
- the creation of structures that include smart roads and monitoring platforms;
- the reduction of the ecological footprint (energy saving, bike lanes, etc.).

However, most solutions rely on efficient and accurate navigation processes.

The development of advanced driver assistance systems (ADAS) for connected and automated mobility makes the accuracy [81] of the navigation process a key issue. Sensors involved in the process of perception and comprehension of the environment, as well as fusion algorithms and decision-makers, have played a significant and evolving role in recent years. Nowadays, many resources provide helpful information to determine position, orientation, time or velocity. Nevertheless, some scenarios and phenomena directly affect the performance of sensor technologies. Thus, it is essential to develop architectures that use a multi-sensor and redundant approach to mitigate the adverse effects caused by the temporary under-performance of one or several sensors.

Recently, proposed architectural solutions have concentrated on sensor fusion [82, 83] rather than implementing the entire On Board Unit (OBU) system [84–88]. This work addresses the fusion process as the core element of the OBU, dedicating a complete architectural layer. The elements used as sensors in our implementation are the inertial measurement unit (IMU) and the global navigation satellite system (GNSS) receiver, complemented by a GNSS augmentation system.

The primary obstacle in enabling AV navigation lies in establishing a consistently accurate and dependable navigation solution across all landscapes. GNSS stands out as the most prevalent source of navigation solutions due to its enduring stability in offering long-term solutions [39].

However, GNSS encounters limitations in providing accurate or any navigation solution in certain circumstances, notably within urban settings that include tunnels, underground parking areas, and high-rise buildings [40,89,90]. Satellite navigation in the urban environment is subject to some objectively critical issues, mainly related to the following items.

- Satellite visibility: due to the presence of urban canyons, the number of visible satellites, crucial for obtaining an accurate position, is limited.
- Multipath and non-line-of-sight reception: the typicality and singularity of the urban context, in which the vehicles carry out their service, produces a non-trivial multipath error that seriously compromises the performance of the GNSS signal.
- Other kinds of interference that can have a significant impact in an urban context.

These criticalities result in considerable variability in the reception of the GNSS signal, ranging from optimal conditions to complete absence. On the other hand, inertial navigation systems (INS) possess several advantages. They operate continuously, exhibit low short-term noise, and resist jamming and interference. However, INS accuracy degrades over time due to integration errors, and maintaining effective sole-means navigation is costly. While GNSS ensures high long-term accuracy, its output rate is lower, and signal obstruction leads to discontinuous navigation. By integrating INS and GNSS, their advantages complement each other. GNSS measurements prevent inertial solution drift, while INS smoothes GNSS output and bridges signal outages. This integration results in a continuous, high-bandwidth navigation solution with long and short-term accuracy [41].

The navigation system emerges as a pivotal enabling factor within the framework of the EMERGE project [21], co-funded by the European Union and spearheaded by RadioLabs in collaboration with esteemed partners such as the University of L'Aquila, Telespazio, Leonardo, and Elital. The project aims to design and develop a novel OBU to provide dual use capabilities to commercial vehicles (i.e., last mile delivery and emergency operations) enabled by dedicated cloud-edge services. The OBU is equipped with vehicle-to-everything (V2X) connectivity [91], procedures for cyber-secure operations [92] and an accurate geo-localization platform. The role of this latter is to guarantee real-time, precise positioning of all nodes and their relative positions, accomplished through innovative fusion techniques that leverage data from multi-constellation GNSS sensors and inertial sensors.

When elements with different characteristics regarding data rate, coding, data type, and time references are integrated, an infrastructure capable of handling all the information flows must be designed. For this reason, this work continues in Section J1.2, describing an architectural proposal applied in the OBU realisation process.

Also, a multi-sensor approach is employed, consequently using data fusion techniques to achieve accurate and robust navigation in different scenarios. Section J1.3 details the integration algorithm based on an Extended Kalman Filter. The experimental setup is divided into two parts: one dedicated to evaluating the individual sensors' performance and the other to evaluating the whole system's performance based on the integration algorithm. Section J1.4 describes the experimentation procedure, and the main results obtained are presented in Section J1.5. Finally, Section J1.6 concludes this article.

J1.2 EMERGE Onboard System Architecture

This section describes the proposed architecture for achieving the EMERGE On board Unit (OBU). The objective is to present and use a generic architecture: easy to understand and implement, with a scalable structure compatible with different hardware platforms and operating systems. In this sense, the challenge is to define a harmonious level of abstraction that guarantees the following functionalities.

- Compatibility: taking advantage of existing components, both: HW and SW (COTS).
- Efficiency: exploiting the advantage of the HW architecture used.
- Performance: establishment of specific metrics in each context.
- Portability: to guarantee compatibility with Operating Systems and HW.
- Operation: real-time or post-processing execution capability.
- Scalability: add/delete components: Sensors, SW, algorithms.

We highlight the proposal of five specified layers based on their functionality and interaction with the navigation system. Including a monitoring layer allows the evaluation of prototypes and different scenarios, the detection of anomalies and problems in the singular components of the system and the evaluation of performances.

Figure J1.1 shows the diagram of the proposed architecture. Each of the layers to which each element belongs is identified by a different colour. The connections, paths and formats/encoding used by each element of the OBU are specified. Below, some details of the functional blocks are presented.

- Sensor (layer 1): includes all sensors capable of providing helpful information in the navigation process, i.e., GNSS receivers, inertial sensors, RTK stations, augmentation services, onboard sensors, and external clocks.

- Parser; Conditioner (layer 2): handles decoding/parsing and filtering information from the sensor level. When communication with sensors (configuration signals, request) is needed, this block also implements the appropriate coding/decoding process.

- Controller (layer 3): controls the start-up, interruptions, restart, and the change of parameters process (in the case of a dynamic scenario). In addition, it manages the possible changes in information flows that may occur between the Core layer and the Sensor layer.

- Core (layer 4): The system's centre contains all the navigation and information management algorithms. In this application, the Sensor Fusion algorithm and the Integrity SW components are located in this layer.

- Monitoring (layer 5): This layer monitors the system's performance in the navigation process and HW resources. It also introduces remote accessibility functionalities and possibly warning signals and notifications.

The proposed blocks or components in each layer are intended to provide the required functionalities of the OBU. The first element to guarantee is connectivity (*"always connected"*). The OBU COMM and CRYPTO ENGINE blocks are external subsystems ensuring remote communication and IP-based services access. The system uses the connection to receive information from the Augmentation GNSS (AGNSS) systems and exchange information with traffic control elements: control centres, other vehicles, or early alert systems. The system proposes solutions based on loosely coupled integration between the inertial sensor measurements and the previously corrected GNSS navigation solutions. The modules involved in the navigation process are dedicated to decoding the information coming from the sensors, managing the information flows and synchronisation. The main objective is to integrate or fuse all useful navigation information to achieve higher levels of accuracy and integrity.

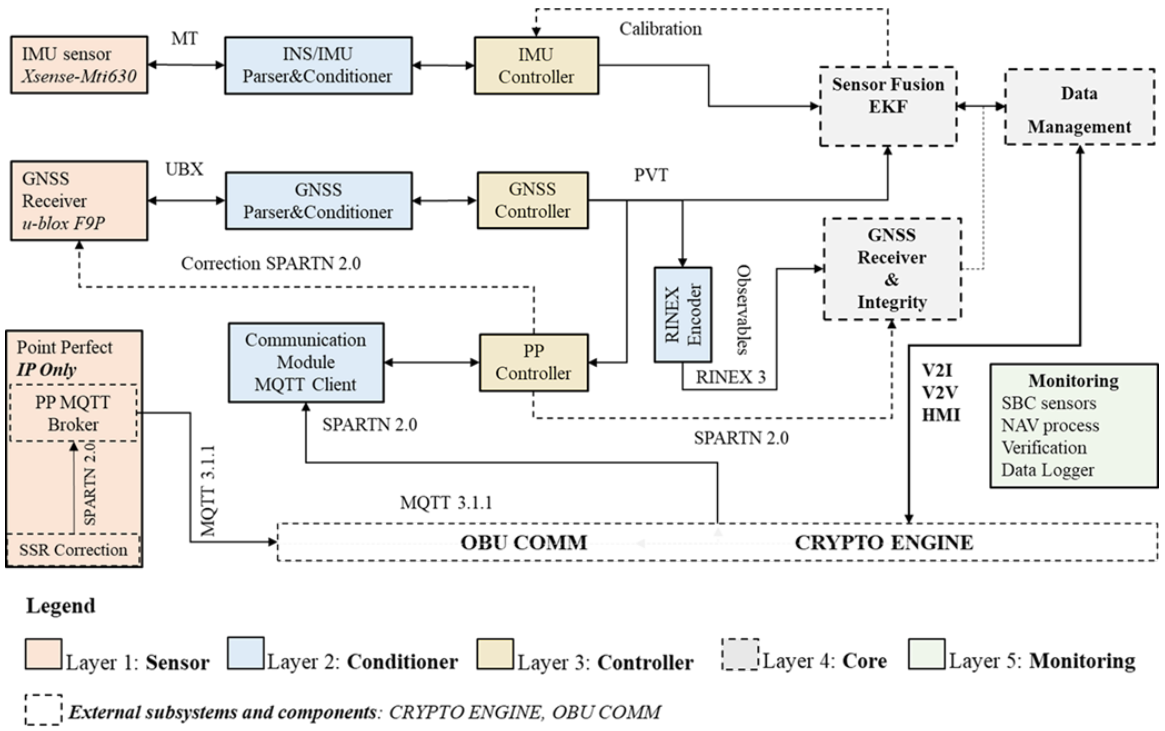

Figure J1.1: EMERGE Onboard System Architecture

J1.2.1 EMERGE Navigation System Implementation

The HW, SW or service components that drive up the onboard navigation system are selected according to the functional requirements of the EMERGE project. For each use case, requirements are mapped to performance measures that sensors must satisfy in the first layer. Specifically, two main elements are considered: the GNSS receiver and the inertial sensor. A GNSS augmentation service is used to improve the satellite positioning performance through corrections received via the OBU connection to the Internet.

J1.2.1.1 Sensors and Services

The GNSS receiver uses the u-blox ZED-F9P positioning module [93] that provides multi-band GNSS for high-volume industrial applications. The ZED-F9P integrates multi-band RTK and PPP-RTK (Precise Point Positioning—Real Time Kinematic) for centimetre accuracy. The module is ideal for automotive and UAV applications due to its low power consumption, accuracy and integration capability. Another strength of this receiver is the ability to handle multiple channels dedicated to different constellations and frequency bands. In addition, the ZED-F9P module features native compatibility with the PointPerfect (PP) augmentation service [20].

PointPerfect is a GNSS augmentation service that supports PPP-RTK for position estimation, with 99.9% availability using the Internet or satellite (L-band) connection. The PP service has reported high availability and high levels of accuracy, making this product optimal for critical applications such as automotive. Specifically, in our configuration, we receive corrections over an IP connection. The GNSS corrections are received in SPARTN 2.0 format transparently and cost-effectively via the MQTT broker offered by the u-blox Thingstream platform [94]. This method of receiving GNSS corrections ensures efficient use of bandwidth, improves GNSS receiver initialisation times and reduces the power consumption of the navigation system. It is important to note that the specific PointPerfect IP service (Unlimited access to IP data stream) costs 19.00 USD per month at the moment of this research.

The inertial sensor component used is the Xsens-MTi-630 AHRS (Attitude and Heading Reference System) [95], which contains a 3-axis gyroscope, a 3-axis accelerometer, a 3-axis barometer magnetometer, a high-precision oscillator and a low-power microcontroller unit (MCU). The MCU applies calibration models (unique to each sensor and including orientation, gain and polarisation offsets, plus more advanced relationships such as non-linear temperature effects and other higher order terms) and runs Xsens' optimised strap-down algorithm, which performs high-speed dead-reckoning calculations up to 2 kHz, enabling accurate capture of high-frequency motions and compensation for coning and sculling. The MTi-600's output data is easily configured and customised according to the application's requirements and can be set to use one of several inertial navigation filter profiles [96]. In particular, the MTi-630 AHRS device enhances the process of determining referenced magnetic north and 3D acceleration, velocity and heading calibration data. The project requirements motivating the use of this sensor are low in-run bias stability (accelerometer: 10 (x,y) and 15 (z) [micro-g]; gyroscope: 8 [deg/h]), low noise density (accelerometers: 60 [micro-g/sqrt(Hz)]) and the ability to output data at a sufficiently high frequency for the sensor fusion process. Successive models of the MTi-600 family (i.e., MTi-670/80) support the INS/GNSS integration process internally; in our case, they do not; therefore, the INS/GNSS synchronisation and integration process is performed by SW.

J1.2.1.2 Sensor Fusion and Integrity

The navigation system's core comprises the components in charge of PNT (Positioning, Navigation and Timing) and Integrity estimation. The multi-sensor approach uses an algorithm that integrates all the elements that provide helpful information

in the navigation process. Section J1.3 describes the loosely coupled algorithm that integrates inertial and GNSS data.

J1.2.1.3 Signal and Information Exchange

The *Always Connected* paradigm opens up many possibilities and functionalities for the satellite navigation system. Therefore, a communication system between internal and external elements is essential. We use MQTT as a communication protocol to guarantee agile and lightweight internal communication between the OBU components (sensors, SW modules, monitoring services). MQTT is a standards-based communication protocol [97], or set of rules, used to communicate from one device/sensor/component to another. Intelligent sensors, wearable devices, and other IoT elements typically transmit and receive data that require modest bandwidth. Therefore, MQTT is used for data exchange as it is easy to implement and can communicate data efficiently.

Figure J1.2 presents a detailed view of exchanging messages via the MQTT broker (Identified as "SBC BROKER"). The arrowheads on the individual connections show whether it is only a subscriber, publishes and subscribes, or only publishes messages through the Broker. For instance, given its observation functionality, the Monitoring block only intends to receive messages and not publish them. Sensors and services publish correctly decoded measured/acquired data in a dedicated Topic, from where they are accessible to the processes or modules that process them. The *Controller4Controllers* module is responsible for configuring, calibrating and synchronising the sensors and coding/decoding SW components. The Broker is also involved in the external communication process via the *OBU COMM* module. After the appropriate encryption process, the information to be transmitted to the control centre is made available in a dedicated Topic managed by the Data Management element.

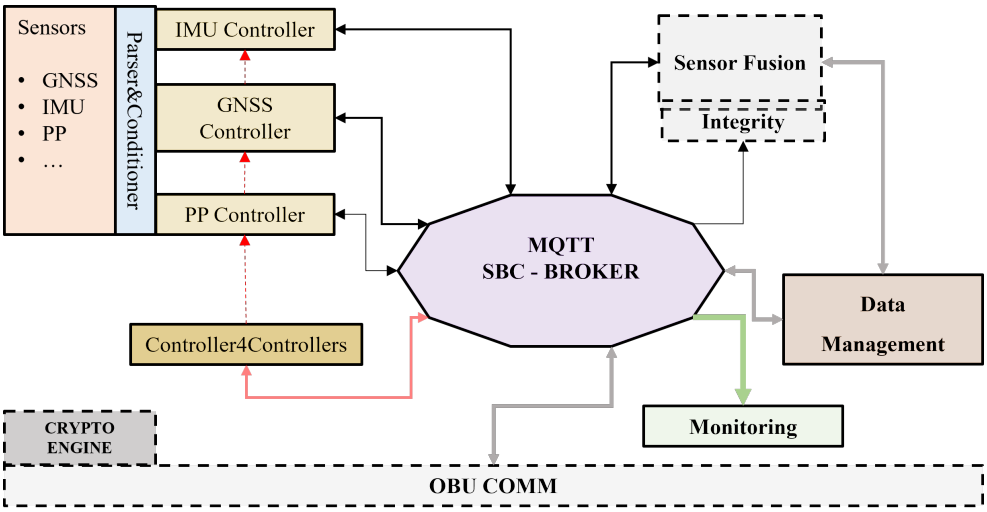

Figure J1.2: MQTT Broker Backend

J1.2.1.4 Power Management System

The Power Management System is the element that ensures that the system stays on and continues to operate regardless of the state of the vehicle. It consists of two elements: a HW and a SW element. The HW element is the Sixfab Power Management; UPS HAT [98], while the SW element is implemented using a dedicated API: Sixfab Power Python API [99]. In short, it is an uninterruptible power supply with a built-in 18650-Li-Ion battery, allowing it to automatically switch from one power supply to another without causing a reboot or system failure. The device can enter deep sleep mode in power-sensitive applications and save battery energy.

J1.2.1.5 Monitoring

The Monitoring component consists of a Frontend, a graphical interface that displays the data of interest, and a Backend responsible for manipulating and storing internal and external information flows. This project's Monitoring layer is based on the Node-RED [100] tool or framework. Node-RED is a flow-based development tool developed to connect devices, APIs and online services as part of the IoT. In addition, it provides a web browser-based flow editor, which can be used to create JavaScript functions. Application elements can be saved or shared for reuse. The runtime uses Node.js, and the streams created are stored using JSON. The decision to use this technology is based on simplicity and support for the MQTT protocol and sensor information. In addition, the built-in WEB service allows remote monitoring of the system.

J1.2.1.6 Hardware Description

The HW used as the On–Board Computer (OBC), shown in Figure J1.3, is the Raspberry Pi 4 Model B [101] equipped with BCM2711, quad–core Cortex–A72 (ARM v8) 64–bit SoC @ 1.8 GHz, 8 GB SDRAM memory and 32 GB Micro–SD card. The sensors (Inertial and GNSS) are connected via USB interfaces, while the Power Management and UPS system is connected to the standard 40-pin GPIO. The SIRIUS RTK GNSS ROVER multi–constellation receiver [102] (based on u–blox ZED-F9P) is connected to the Septentrio Polant–x MF antenna via the SMA interface provided. Both the antenna and the GNSS receiver verify multi–frequency and multi–constellation compatibility. Finally, the components are assembled inside a Sixfab enclosure specifically dedicated to Raspberry projects in outdoor environments with IP65 compliance. An IP65–rated enclosure gives protection against moisture, rain, snow, wind, dust and low–pressure water jets from any direction, as well as condensation and water spray. It is suitable for most outdoor enclosures that will not encounter extreme weather. Figure J1.4 shows the final Navigation Box, the interfaces and the GNSS antenna.

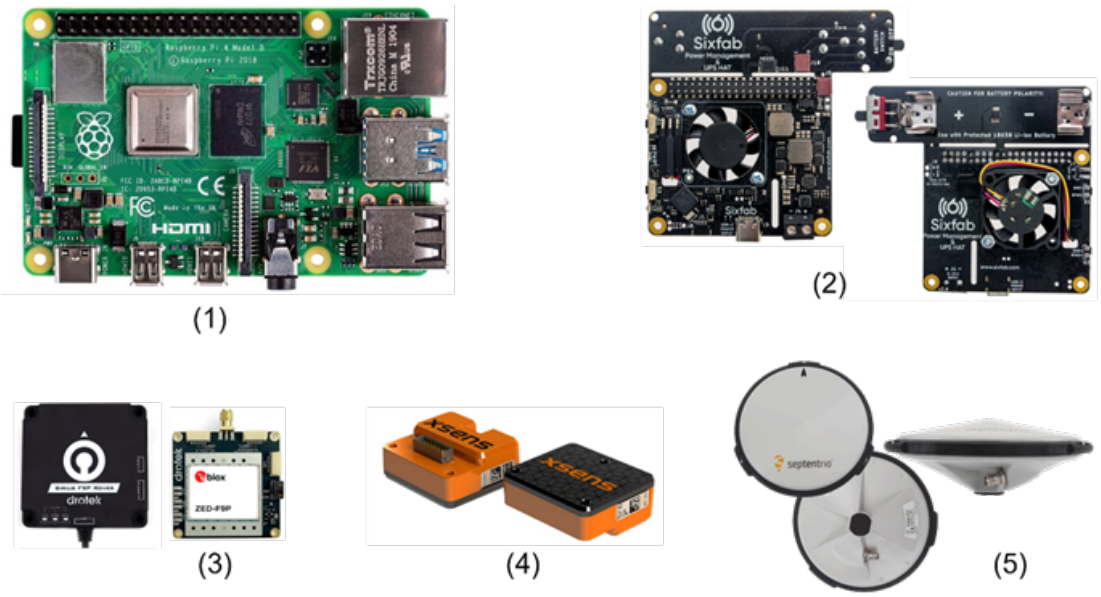

Figure J1.3: OBU HW components: (1) Onboard Computer Raspberry Pi 4 Model B; (2) Sixfab Power Management; UPS HAT power backup system; (3) Drotek (SIRIUS RTK GNSS ROVER: F9P) with u–blox ZED-F9P chip; (4) Xsens MTi630 AHRS Inertial Sensor; (5) Septentrio Polant–x MF antenna

Figure J1.4: Final assembled system: Navigation EMERGE OBU

J1.3 Sensor Fusion: Loosely Coupled Algorithm Implementation

The primary advantage of a loosely coupled integration architecture lies in its simplicity. This architecture is versatile and compatible with any INS and GNSS user equipment, making it ideal for retrofit applications. In a loosely coupled INS/GNSS system, the integration algorithm utilizes GNSS position and velocity solutions as measurement inputs, regardless of the specific type of INS correction or GNSS aiding employed.

In the cascaded operation of this architecture, the GNSS user equipment, which incorporates a navigation filter [41], utilizes the Extended Kalman Filter (EKF) as the integration solution. The EKF is a recursive filter specifically designed to estimate the state of a dynamic system from multiple noisy measurements. It has evolved from the Standard Kalman Filter to address the complexities associated with nonlinear dynamic systems. The resulting integrated navigation solution consists of the INS navigation solution refined by the Kalman filter's error estimates. Specifically, within the loosely coupled integration architecture, the sequential processes can be delineated as follows:

1. calculation of position and velocity with GNSS;

2. calculation of the difference between the estimated position and velocity from the GNSS and INS solutions to assess IMU errors by integrating the estimated differences in a Kalman filter;

3. correction of the INS solution using variational equations.

The tests were carried out in a confined area. To simplify the spatial representation in this local context, the North East Down (NED) coordinate frame was chosen. Consequently, the position of the OBU is represented using geodetic coordinates: latitude L_b, longitude λ_b, and ellipsoidal altitude h_b. The Earth-referenced velocity is resolved in the local navigation axis to give $\mathbf{v}_{eb}^n$, and the attitude is expressed as the body-to-local navigation frame coordinate transformation matrix $\mathbf{C}_b^n$. Here, e denotes the Earth-Centered Earth-Fixed (ECEF) reference frame, n is the NED frame, and b represents the inertial platform "body" coordinate frame.

Figure J1.5 illustrates the functional diagram of the GNSS-INS loose coupling process. It is a closed-loop correction architecture; consequently, the estimated errors in the NED reference frame, $\delta\hat{L}_b$ for the latitude, $\delta\hat{\lambda}_b$ for the longitude, $\delta\hat{h}_b$ for the height, $\delta\hat{\mathbf{v}}_{eb}^n$ for the velocity and $\delta\hat{\mathbf{C}}_b^n$ for the attitude are fed back to the inertial navigation processor, where they are used to correct the inertial navigation solution. In this way, the integrated navigation solution of the navigation system is the inertial navigation solution itself, and is obtained, respectively, for the orientation, $\hat{\mathbf{C}}_b^n$, the velocity, $\hat{\mathbf{v}}_{eb}^n$, and the position, as follows [41]:

$$\hat{\mathbf{C}}_b^{n+} = \left(\mathbf{I} - [\delta\hat{\boldsymbol{\psi}}_{nb}^n]_{\mathrm{X}}\right)\hat{\mathbf{C}}_b^{n-} \tag{J1.1}$$

$$\hat{\mathbf{v}}_{eb}^{n+} = \hat{\mathbf{v}}_{eb}^{n-} - \delta\hat{\mathbf{v}}_{eb}^n \tag{J1.2}$$

$$\hat{L}_b^+ = \hat{L}_b^- - \delta\hat{L}_b \tag{J1.3}$$

$$\hat{\lambda}_b^+ = \hat{\lambda}_b^- - \delta\hat{\lambda}_b \tag{J1.4}$$

$$\hat{h}_b^+ = \hat{h}_b^- - \delta\hat{h}_b \tag{J1.5}$$

where superscripts $-$ and $+$ are used to indicate the solution before and after correction, $\delta\hat{\boldsymbol{\psi}}_{nb}^n$ is the vector of Euler angles of the correction (i.e., yaw, pitch, and roll), and $[\delta\hat{\boldsymbol{\psi}}_{nb}^n]_{\mathrm{X}}$ represents the antisymmetric matrix

$$[\delta\hat{\boldsymbol{\psi}}_{nb}^n]_{\mathrm{X}} = \begin{bmatrix} 0 & -\delta\hat{\psi}_{nb,z}^n & \delta\hat{\psi}_{nb,y}^n \\ \delta\hat{\psi}_{nb,z}^n & 0 & -\delta\hat{\psi}_{nb,x}^n \\ -\delta\hat{\psi}_{nb,y}^n & \delta\hat{\psi}_{nb,x}^n & 0 \end{bmatrix}.$$

The pseudocode outlined in Algorithm 3 presents the procedural steps of the implemented INS-GNSS loosely coupled algorithm. Initially, it sets crucial parameters to account for IMU errors. The algorithm then begins with an initialisation loop

which is used to estimate the initial pose of the vehicle and to configure Kalman filter parameters. After initialisation, it uses IMU data to update the vehicle's pose, incorporating corrections from the Kalman filter when new GNSS measurements are available and providing the integrated navigation solution as output. A Zero Velocity Update (ZUPT) is also employed to mitigate the position drift introduced when the vehicle is stationary. All these steps are further described in the following.

Algorithm 3 Loosely coupled algorithm

Set initialisation parameters for IMU errors;
$frequency_{IMU} = 100$;
init_time = 15;
init_imu_samples = init_time * $frequency_{IMU}$;
level_time = 10;
init_imu_samples_level = level_time * $frequency_{IMU}$;
for $i = 1$ to init_imu_samples **do**
    Initialisation loop
end for
for $i =$ init_imu_samples $+ 1$ to end **do**
    Specific force $f_{ib}^{b} \leftarrow fb_{IMU}(i)$;
    Angular velocity $\omega_{ib}^{b} \leftarrow wb_{IMU}(i)$;
    Correct f and w using estimated biases;
    Update navigation solution with mechanisation equations;
    Apply ZUPT detection algorithm;
    if There are new r, v solution to measure **then**
        Apply Kalman Filter;
        Update the navigation solution with Kalman filter estimates;
    end if
end for

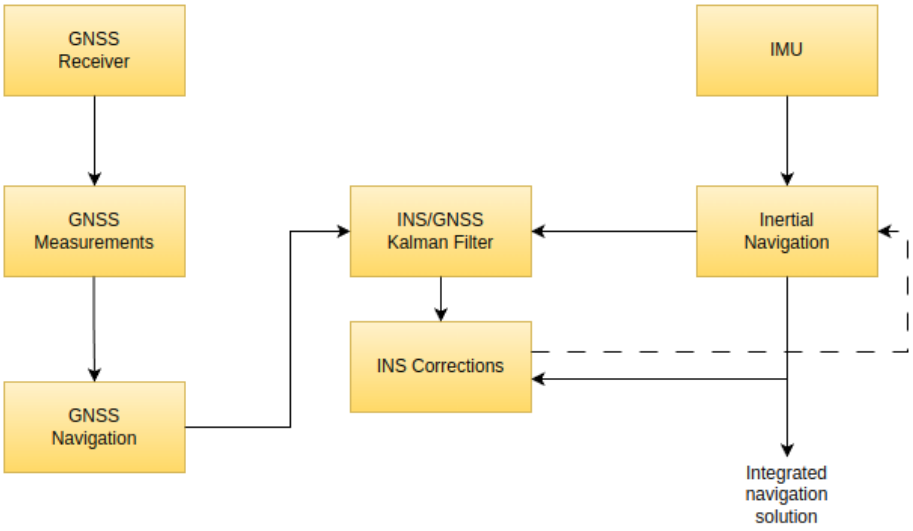

Figure J1.5: Loosely coupled INS/GNSS coupling scheme

J1.3.1 Setting Initialisation Parameters for IMU Errors

Inertial navigation systems are renowned for delivering highly accurate position, velocity, and attitude information, particularly over short time spans. However, this precision degrades significantly over extended periods due to inherent error sources within the sensors. To address this challenge, the algorithm's first step focuses on precisely defining the initialisation parameters related to IMU errors. This meticulous calibration lays the groundwork for mitigating sensor inaccuracies and ensures the subsequent integration of INS-GNSS data yields precise and reliable navigation estimates.

The Allan variance method is used to characterise various error types present in inertial sensor data. This method allows for representing root mean square (RMS) random drift error as a function of averaging time [42]. Inertial measurements were analysed using the Allan variance method; precisely, specific force and angular velocity measurements collected from the IMU during a stationary five-hour test were utilized. These measurements were crucial in deriving parameters such as velocity random walk, angle random walk, angle rate random walk, bias instability, correlation times, and dynamic bias root–PSD for the accelerometer and gyroscope.

J1.3.2 The Initialisation Loop

As a systematic iteration over IMU samples, this Loop (4) calculates essential parameters and initializes the navigation solution when the system is stationary. Its

execution sets the stage for optimal accuracy and performance of the Kalman filter during subsequent integration, establishing a robust foundation for a precise and reliable navigation solution. Within each iteration over the IMU measurement taken in the first seconds, the initialisation loop calculates the IMU sampling interval (tor_i) considering the IMU frequency, facilitating precise temporal alignment. Specific force ($\mathbf{f}_{ib}^{b}$) and angular velocity ($\boldsymbol{\omega}_{ib}^{b}$) measurements are extracted from the IMU, providing essential motion-related data. The Algorithm 4 checks for the availability of new GNSS measurements within the current Inertial Navigation System (INS) time. If available, it updates the GNSS position. To account for synchronisation issues between the IMU and GNSS data, this check is performed within a time window of duration $2\varepsilon = 5$ ns. IMU measurements are accumulated for levelling purposes and also GNSS data, when available, is accumulated to initialize position parameters. Roll and pitch are computed based on the averaged specific force measurements through the levelling process. The estimated position is initialized using the averaged GNSS positions.

Algorithm 4 Initialisation Loop

gnss_eps= ε; %duration in seconds
for $i = 1$ to init_imu_samples **do**
  if $i == 1$ **then**
    $tor_i \leftarrow \frac{1}{frequency_{IMU}}$;
  else
    $tor_i \leftarrow t_{IMU}(i) - t_{IMU}(i-1)$;
  end if
  $f^b_{ib} \leftarrow fb_{IMU}(i)$;
  $\omega^b_{ib} \leftarrow wb_{IMU}(i)$;
  $is\_gnss\_available \leftarrow$ false;
  $gdx \leftarrow$ find($t_{GNSS} \geq (t_{IMU}(i) -$ gnss_eps) and $t_{GNSS} < (t_{IMU}(i) +$ gnss_eps));
  %Check if there is a new GNSS measurement to process at current INS time
  if (!isempty(gdx) and $gdx > 1$) **then**
    $is\_gnss\_available \leftarrow$ true;
    $last\_gdx \leftarrow gdx$;
    gnss position $GNSS\_r \leftarrow$ [gnss_lat(gdx); gnss_lon(gdx); gnss_h(gdx)];
  end if
  $f\_leveling^b_{ib} \leftarrow [f\_leveling^b_{ib}, f^b_{ib}]$;
  $\omega\_leveling^b_{ib} \leftarrow [\omega\_leveling^b_{ib}, \omega^b_{ib}]$;
  % accumulate GNSS solutions to initialize position
  if $is\_gnss\_available$ **then**
    $GNSS\_r\_init = [GNSS\_r\_init, GNSS\_r]$;
  end if
  if ($i >$ init_imu_samples_level and !isempty($GNSS\_r\_init$)) **then**
    $ave\_f^b_{ib} \leftarrow$ mean($f\_leveling^b_{ib}$, 2);
    $roll \leftarrow$ atan2($-ave\_f^b_{ib}(2), -ave\_f^b_{ib}(3)$);
    $pitch \leftarrow$ atan($ave\_f^b_{ib}(1)/\sqrt{ave\_f^b_{ib}(2)^2 + ave\_f^b_{ib}(3)^2}$);
    $est\_r^n_{eb} \leftarrow$ mean($GNSS\_r\_init$, 2); %Initializes the estimated position
  end if
end for

J1.3.3 Specific Force and Angular Velocity Error Model

The primary sources of error are modelled as follows. Given the true specific force and angular velocity measurements, their noisy counterparts $\tilde{\mathbf{f}}^b_{ib}$ and $\tilde{\boldsymbol{\omega}}^b_{ib}$ are given by:

$$\tilde{\mathbf{f}}^b_{ib} = \mathbf{f}^b_{ib} + \mathbf{b}_a + \mathbf{n}_a \tag{J1.6}$$

$$\tilde{\boldsymbol{\omega}}^b_{ib} = \boldsymbol{\omega}^b_{ib} + \mathbf{b}_g + \mathbf{n}_g \tag{J1.7}$$

where $\mathbf{b}_a$ and $\mathbf{b}_g$ represent the biases on accelerometers and gyroscopes measurements, respectively, and $\mathbf{n}_a$ and $\mathbf{n}_g$ are the corresponding random noises, assumed to follow a zero-mean normal distribution.

J1.3.4 Navigation Solution Update

Mechanisation equations (in discrete form) [43] are used to update the solution of inertial navigation at the next time instant, using measurements of angular velocity $\boldsymbol{\omega}_{ib}^{b}$ and specific force $\mathbf{f}_{ib}^{b}$ from the IMU sensors. Since mechanisation equations are the result of approximations and may introduce errors, an optimised version was used.

Orientation update: if the angular velocity remains constant during the integration interval, i.e., if τ is sufficiently small, the updated rotation matrix representing the attitude of the vehicle is given by

$$\mathbf{C}_{b}^{n}(t+\tau) \approx \mathbf{C}_{b}^{n}(t)\left(\mathbf{I}+[\boldsymbol{\omega}_{ib}^{b}]_{\mathrm{X}}\tau\right)-\left(\boldsymbol{\Omega}_{ie}^{n}+\boldsymbol{\Omega}_{en}^{n}\right)\mathbf{C}_{b}^{n}(t)\tau, \tag{J1.8}$$

where

$$\boldsymbol{\Omega}_{ie}^{n}=[\boldsymbol{\omega}_{ie}^{n}]_{\mathrm{X}}=\boldsymbol{\omega}_{ie}\begin{pmatrix}0 & \sin(L_b) & 0\\ -\sin(L_b) & 0 & -\cos(L_b)\\ 0 & \cos(L_b) & 0\end{pmatrix} \tag{J1.9}$$

is the skew-symmetric matrix of the rotation vector of the Earth, $\boldsymbol{\omega}_{ie}$, resolved in the navigation frame,

$$\boldsymbol{\Omega}_{en}^{n}=[\boldsymbol{\omega}_{en}^{n}]_{\mathrm{X}} \Leftarrow \boldsymbol{\omega}_{en}^{n}\begin{pmatrix}v_{eb,E}^{n}/R_E(L_b)+h_b\\ -v_{eb,E}^{n}/R_N(L_b)+h_b\\ v_{eb,E}^{n}\tan(L_b)/R_E(L_b)+h_b\end{pmatrix} \tag{J1.10}$$

is the skew-symmetric matrix of the *transport rate* vector due to the rotation of the navigation frame with respect to the Earth, R_E and R_N are respectively the radius of transverse curvature and the radius of curvature of the meridian at that point. We further define the attitude axis update matrix as the coordinate transformation matrix from the body reference frame at the end of the update ($b+$) to the body reference system at the beginning ($b-$)

$$\mathbf{C}_{b+}^{b-}=\exp[\boldsymbol{\alpha}_{ib}^{b}]_{\mathrm{X}}=\sum_{r=0}^{\infty}\frac{[\boldsymbol{\alpha}_{ib}^{b}]_{\mathrm{X}}^{r}}{r!}, \tag{J1.11}$$

where $\boldsymbol{\alpha}_{ib}^{b}=\boldsymbol{\omega}_{ib}^{b}\tau$ is the attitude increment. Truncating the formula to the fourth order gives the Rodrigues formula, used to calculate:

$$\mathbf{C}_{b}^{n}(t+\tau) \approx \left[\mathbf{I}-\left(\boldsymbol{\Omega}_{ie}^{n}+\frac{1}{2}\boldsymbol{\Omega}_{en}^{n}(t)+\frac{1}{2}\boldsymbol{\Omega}_{en}^{n}(t+\tau)\right)\tau\right]\mathbf{C}_{b}^{n}(t)\mathbf{C}_{b+}^{b-} \tag{J1.12}$$

where $\boldsymbol{\Omega}_{en}^{n}(t+\tau)$ is calculated from $L_b(t+\tau), \lambda(t+\tau), h_b(t+\tau)$.

Velocity update:

$$\mathbf{v}_{eb}^{n}(t+\tau)=\mathbf{v}_{eb}^{n}(t)+[\mathbf{f}_{ib}^{n}+\mathbf{g}_{b}^{n}(L_b,h_b)-(\boldsymbol{\Omega}_{en}^{n}+2\boldsymbol{\Omega}_{ie}^{n})\mathbf{v}_{eb}^{n}(t)]\tau \tag{J1.13}$$

where the acceleration due to gravity, $\mathbf{g}_b^n$, is modeled as a function of latitude and height, and $\mathbf{f}_{ib}^n$ denotes the measured specific force resolved in the local navigation frame using the estimated $\mathbf{C}_b^n$.

Position update:

$$\mathbf{h}_b(t+\tau) \approx \mathbf{h}_b(t) - \frac{1}{2}[\mathbf{v}_{eb,D}^n(t) + \mathbf{v}_{eb,D}^n(t+\tau)] \tag{J1.14}$$

$$\mathbf{L}_b(t+\tau) \approx \mathbf{L}_b(t) + \frac{1}{2}\left[\frac{\mathbf{v}_{eb,N}^n(t)}{R_N(\mathbf{L}_b(t)) + \mathbf{h}_b(t)} + \frac{\mathbf{v}_{eb,N}^n(t+\tau)}{R_N(\mathbf{L}_b(t)) + \mathbf{h}_b(t+\tau)}\right]\tau \tag{J1.15}$$

$$\boldsymbol{\lambda}_b(t+\tau) \approx \boldsymbol{\lambda}_b(t) + \tfrac{1}{2}\left[\frac{\mathbf{v}_{eb,E}^n(t)}{(R_N(\mathbf{L}_b(t))+\mathbf{h}_b(t))\cos\mathbf{L}_b(t)} + \frac{\mathbf{v}_{eb,E}^n(t+\tau)}{(R_N(\mathbf{L}_b(t+\tau))+\mathbf{h}_b(t+\tau))\cos\mathbf{L}_b(t+\tau)}\right]\tau \tag{J1.16}$$

In order to have the velocity and position accuracy update, we considered the use of the mean transformation matrix $\bar{\mathbf{C}}_b^n$ in the transformation of the specific force in the NED coordinate system:

$$\bar{\mathbf{C}}_b^n = \mathbf{C}_b^{n-}\mathbf{C}_{\bar{b}}^{b-} \tag{J1.17}$$

where

$$\mathbf{C}_{\bar{b}}^{b-} = \mathbf{I} + \frac{1-\cos\left|\boldsymbol{\alpha}_{ib}^b\right|}{\left|\boldsymbol{\alpha}_{ib}^b\right|^2}[\boldsymbol{\alpha}_{ib}^b]_{\mathrm{X}} + \frac{1}{\left|\boldsymbol{\alpha}_{ib}^b\right|^2}\frac{1-\sin\left|\boldsymbol{\alpha}_{ib}^b\right|}{\left|\boldsymbol{\alpha}_{ib}^b\right|}[\boldsymbol{\alpha}_{ib}^b]_{\mathrm{X}}^2 \tag{J1.18}$$

obtaining

$$\mathbf{f}_{ib}^n = \bar{\mathbf{C}}_b^n\mathbf{f}_{ib}^b, \qquad \bar{\mathbf{C}}_b^n = \mathbf{C}_b^{n-}\mathbf{C}_{\bar{b}}^{b-} - \frac{1}{2}(\boldsymbol{\Omega}_{ie}^n + \boldsymbol{\Omega}_{en}^n)\mathbf{C}_b^{n-}\tau \tag{J1.19}$$

J1.3.5 ZUPT Detection Algorithm

To make the Algorithm 3 more efficient, the ZUPT algorithm is used to correct errors accumulated in the inertial navigation data when the system is stationary [44]. Thus, when the average velocity is below a fixed threshold of 0.5 m/s for a fixed time period of 4 s, the algorithm assumes the system is stationary and updates the current attitude and position based on the averages of the previous values.

J1.3.6 Kalman Filter

The state vector of the Kalman filter includes orientation, velocity and position errors used in Equations (J1.1)-(J1.5), as well as the biases of the accelerometer and gyroscope, and is given by

$$\hat{\mathbf{x}}_k = [\delta\boldsymbol{\psi}_{eb}^n \quad \delta\mathbf{v}_{eb}^n \quad \delta\mathbf{p}_b \quad \mathbf{b}_a \quad \mathbf{b}_g]^T, \tag{J1.20}$$

where the position error is

$$\delta \mathbf{p}_b = [\delta \hat{L}_b \quad \delta \hat{\lambda}_b \quad \delta \hat{h}_b]^T. \tag{J1.21}$$

Whenever corrections from the integration filter are applied, the values of the corresponding states are reset to zero. The algorithm implemented for the Kalman filter is as follows.

1. Calculation of the covariance matrix $\mathbf{P}_k^-$ of the prediction error at instant k:

$$\hat{\mathbf{x}}_k^- = 0 \tag{J1.22}$$

$$\mathbf{P}_k^- = \boldsymbol{\Phi}_{k-1} \mathbf{P}_{k-1}^+ \boldsymbol{\Phi}_{k-1}^T + \mathbf{Q}_{k-1} \tag{J1.23}$$

   where $\mathbf{Q}$ is the system noise covariance matrix, and the transition matrix $\boldsymbol{\Phi}$ is obtained by computing the expected value of the time derivative for each state.

2. Calculation of the Kalman gain matrix:

$$\mathbf{K}_k = \mathbf{P}_k^- \mathbf{H}_k^T \left(\mathbf{H}_k \mathbf{P}_k^- \mathbf{H}_k^T + \mathbf{R}_k \right)^{-1}, \tag{J1.24}$$

   where $\mathbf{H}_k$ is the measurement matrix and $\mathbf{R}_k$ is the measurement noise covariance matrix at epoch k.

3. Filtering of the system state at instant k based on GNSS measurements, and calculation of the covariance matrix of the filtered estimation error:

$$\hat{\mathbf{x}}_k^+ = \mathbf{K}_k \delta \mathbf{z}_k^- \tag{J1.25}$$

$$\mathbf{P}_k^+ = \left(\mathbf{I} - \mathbf{K}_k \mathbf{H}_k \right) \mathbf{P}_k^- \tag{J1.26}$$

where $\delta \mathbf{z}_k^-$ is the vector of observations (i.e., the difference between position and velocity measured by the GNSS and the corresponding values estimated by the inertial navigation system). It's important to note that all estimated quantities are derived based on preceding correction. The matrices $\boldsymbol{\Phi}_{k-1}$, $\mathbf{Q}_{k-1}$, $\mathbf{R}_k$ used in this study are resolved in a local navigation (NED) frame and can be found in Appendix J1.7 of this paper.

J1.4 Experimental Setup

This section describes the materials and methods used to test the navigation system. The testing process covered all the elements: sensors, services, communication elements, power consumption, synchronisation and especially the data integration/fusion for the PNT process. In the first part, this section describes the experimental activities to evaluate the performance of sensors/services used in the navigation process: GNSS receiver, GNSS Augmentation (PP) service and inertial sensor. Afterwards, it is shown the experimental methodology carried out during the final test, evaluating the impact of the sensor fusion algorithm on the navigation process in an on-road environment.

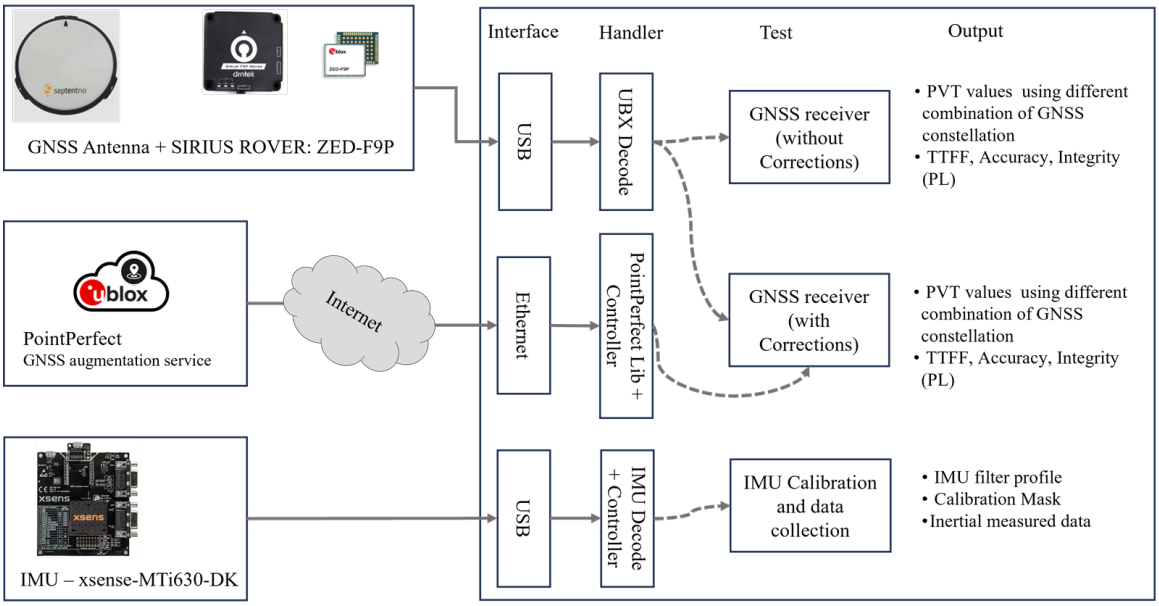

Figure J1.6: Experimental setup for sensor verification tests

Figure J1.6 presents a detailed schematic of the experimental verification process conducted for each key system sensor. The purpose of this experimental setup is to evaluate each subsystem and check the individual performances of each sensor. The test also covers the process of interconnection, configuration, decoding, and calibration of the sensors involved. The verification process of the GNSS system (GNSS Antenna + SIRIUS ROVER: ZED-F9P) is a comprehensive series of experimental tests where all the components of the GNSS subsystem are thoroughly evaluated: the antenna, the communication interfaces, the SW block dedicated to decoding, as well as the performances in terms of position calculation (PVT). Furthermore, the impact of the PointPerfect (Augmentation service) is evaluated. Table J1.1 shows

some performance results that cover the cases of using the GPS and Galileo constellations without PointPerfect service and the case of using the full capability of the GNSS receiver (GNSS + PointPerfect augmentation service). The reference system (known positions) used in the performance calculations was derived from previous long-term PPP-RTK measurements. The inertial measurement system (Xsens—MTi 630 AHRS) is also verified in a stationary scenario. In addition to validating the interfaces, configuration, and SW components dedicated to coding/decoding, this process allows the creation of a data set that was used to characterise the inertial error in the GNSS + IMU integration process.

Table J1.1: Experimental results of the satellite navigation module with two different configurations: using two GNSS constellations (GPS + GALILEO) and the GNSS (all constellations) integrated with the PointPerfect Augmentation service (GNSS + PointPerfect)

KPI[1]	**Parameters**	**GPS + GALILEO** [μ, σ]	**GNSS**[2] **+ PP** [μ, σ]
Accuracy	3D Position [m]	2.26, 0.52	0.10, 0.25
	H-Position [m]	1.70, 0.44	0.06, 0.18
	V-Position [m]	1.46, 0.41	0.08, 0.18
Availability	SVs Used [U]	11, 2	26, 2
	SVs Tracked [U]	24, 3	36, 3
	SVs Received [U]	28, 3	37, 3
	C/N0 [dB-Hz]	36.47, 1.5	34.31, 1.3
	GDOP	2.3, 0.6	1.7, 0.2
	HDOP	1.3, 0.5	0.8, 0.13
	VDOP	1.4, 0.4	1.2, 0.2
Integrity	North PL [m]	8.86, 4.69	2.09, 5.59
	East PL [m]	10.93, 6.61	2.44, 6.03
	Down PL[m]	11.75, 6.41	4.08, 7.48
TTFF	TTFF[3] [s]	28.75, 1.71	26.34, 1.33
Other	3D Vel. Err. [m/s]	0.26, 0.09	0.13 0.03
	Time Acc. [ns]	0.0036, 0.0009	0.0011, 0.0009
	TDOP	1.1, 0.3	0.8, 0.1

[1] This table groups together the information collected that could provide answers to the KPIs organised in the first column. In the last years, some constellations have defined specific KPIs and parameters, so note that the parameters shown are only a portion of these. [2] TTFF is time elapsed between a *coldstart* and a valid *Fix* detected in the GNSS receiver.

J1.4.1 Sensors Verification

Before a complete and definitive test of the OBU in a real environment, we present a verification of the performance of some fundamental system elements. Particular interest falls on the satellite navigation system, which includes two fundamental components: the GNSS multi-constellation/multi-frequency receiver and the Augmentation Service, which provides the relevant atmospheric and clock corrections. The sensor verification process allows the performance to be characterised locally through appropriate configuration. For example, it can evaluate satellite availability and coverage (DOP), the average Time To First Fix (TTFF), and the most effective combinations of constellations and frequencies in the case of GNSS.

J1.4.2 Xsens MTi630 AHRS

The tests conducted on the inertial navigation component focused on the calibration process and the correct functionality of the sensor. For this, we used the Xsens Development Kit MTi630 connected to a PC and a SW dedicated to the configuration and storage of data measured by the sensor. The experiment configuration just involves connecting the PC and the MTi-630-DK using the corresponding USB interface. The acquisition SW is implemented in Python and uses the manufacturer open-source API (Xsens Device API) as a library. To summarise the flow followed by the acquisition SW: a first "startup" phase dedicated to the creation of the XDA library objects, scanning of the ports/interfaces and configuration of the device; a second "processing and storage" step dedicated to the manipulation and storage of the inertial data; and finally the step of closing and disconnection of the objects, ports and files used. In addition, a summary of the IMU configuration profile, including motion filters, calibration information and inertial system data, is stored.

J1.4.3 GNSS and Augmentation Service

The whole satellite navigation system testing process is conducted in two steps: one dedicated only to verify the correct functionality of the GNSS receiver, and the other one to evaluate the impact of using the PointPerfect service. For the first step, the HW involved includes the GNSS antenna, the GNSS receiver, and a PC that connects to the receiver via the USB port. In the second step, an Internet connection is needed to access the PointPerfect service. This test is performed in a controlled outdoor environment with the antenna placed on a vehicle at rest.

The tests described in this section focus on the evaluation of the performance of the satellite navigation system, considering the most important parameters in automotive positioning: TTFF, Accuracy, Integrity and satellite availability metrics. The GNSS data acquisition process (with and without corrections) is carried out using an acquisition SW that has the following flow: *startup* (configuration of interfaces, objects, devices and services); *processing* (main cycle in charge of decoding the UBX data coming from the GNSS sensor, the corrections coming from the PointPerfect service, and the file storage process); and finally the *close* and *disconnection* phase of the elements used. In the case of the TTFF tests, repetitions are performed from a "*coldstart*" and, afterwards, the mean is applied to the measured values.

J1.4.4 Sensor Integration: On-Road Navigation Test

The integration or fusion process is the core of the multi-sensor approach to navigation. This section describes the verification process applied to the algorithm described in Section J1.3. Having characterised the sensors/services in terms of noise, accuracy and response times, the improvements introduced by the integration process can be quantified. In this case we use a Raspberry Pi 4 Model B (On Board Controller), the satellite navigation system described in the previous section (Antenna, Receiver and GNSS Augmentation Service), the MTi630-DK, everything installed within a compact city car. Figure J1.7 specifies the coordinate system and reference frame used. In this case, we match the inertial reference frame (i-frame) with the body/vehicle reference frame (b-frame). The conversion process to a unique navigation system (n-frame) uses the corresponding transformations (i.e., Rotation Matrix). Likewise, the reference system used by the ground-referenced GNSS (e-frame) shall be in a compatible reference system to facilitate the fusion process.

The navigation information is captured during the experiment using a data-logger script. The main elements to be stored for future analysis are GNSS information, inertial measurements, timing information, sensor fusion process results, and Kalman filter status information. This experiment has been conducted in the following sequence.

- *Startup phase*

    1. Power-up of the whole system: vehicle and OBU
    2. Initialisation of the acquisition SW

(a) Initialise libraries, dependencies, SW modules and objects: MQTT client/ server, parser/deparser UBX, XDA, PP credentials and files

(b) Sensor (interface) detection

(c) Calibrate and/or configure sensors

(d) Prepare for data handling, KF and synchronisation structures

(e) Change operation mode (in sensors) to measurement mode

- *Processing and Measurement phase*

3. Handle incoming data: queuing, synchronisation, data availability and status of the fusion algorithm
4. Write the information (synchronously) to the logger files
5. Verify the conditions for the end of the experiment

- *Exit and Close phase*

6. Save the final status of the system elements: GNSS, Inertial Unit and SF data
7. Close open devices and files
8. Clear and release all the structures used
9. Print the successful experiment message

The On-road experiment has been conducted in the parking area of the Tecnopolo of Abruzzo, city of L'Aquila, Italy, and lasted 240 s (4 min). During the test, a trajectory along the marked tracks has been followed, with acceleration intervals and stops, to simulate a typical urban scenario. For this experiment, the inertial sensor samples the data with a frequency of 100 Hz, and the GNSS receives one updated PVT message every second.

J1.5 Results

The most relevant results in evaluating the sensors lie in the accuracy of the satellite navigation system and the improvement obtained using an external GNSS augmentation service. In addition, considering the automotive application, we provide some statistics on the power consumption of the OBU. This section concludes by showing the final results of the sensor fusion application.

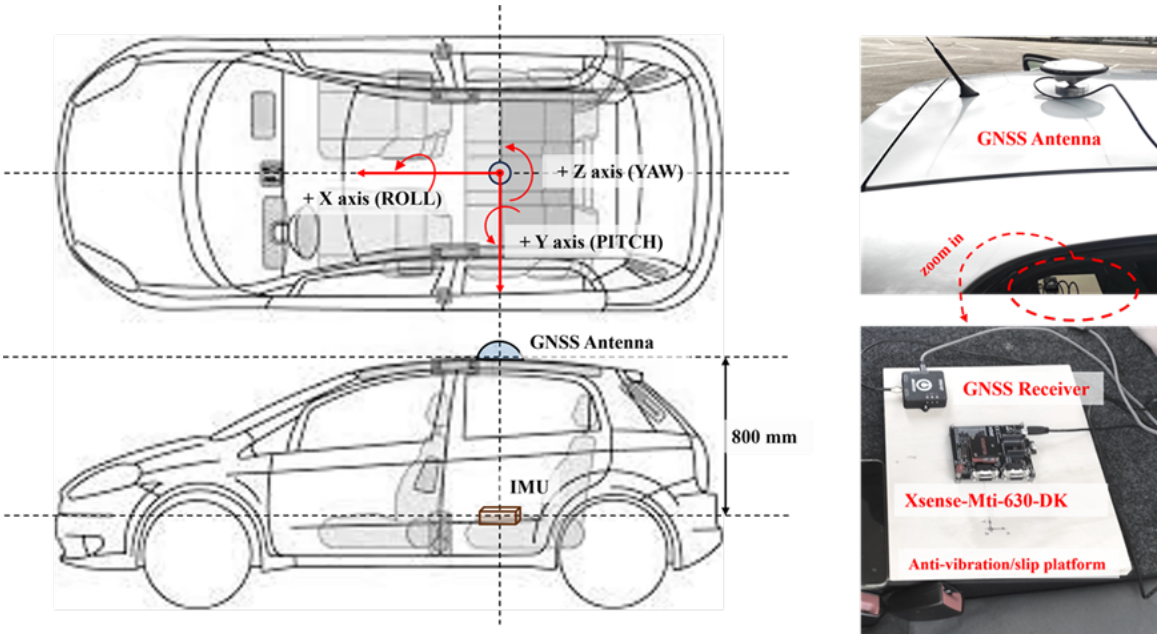

Figure J1.7: Coordinate system, frame and hardware for final integration test

J1.5.1 GNSS Performance

The first GNSS key issue analysed was the TTFF measured starting from a "cold-start" signal to the receiver until it reaches a 3D Fix. Experimental results show a mean value slightly below 30 s. This result has been verified and is consistent with the data provided by the manufacturer. However, even if this result is satisfactory, it is important to note that a lower TTFF may be required in automotive applications. This is why the OBU foresees the storage of the last position/status of the vehicle after it has been powered off, functionalities supported thanks to the internally incorporated Power Management and UPS system. The backup battery connected to the Protected 18650 Li-Ion Separable Battery Holder has a capacity of 3000 mAh (at 3.7 V). Assuming OBU consumption in the high operating mode (average power of 5 W and average current of 1 A), the system autonomy, regarding energy, is about 2 h.

Table J1.1 compiles the results obtained in the evaluation process by accumulating data for more than 5000 s (approximately one and a half hours) at different fixed known positions and times along a given day. The first two columns of the table match the key performance indicators (KPI) with the corresponding measured parameters. Two measurement sets have been collected: one using two GNSS constellations (GPS + GALILEO) without GNSS corrections and the other, including the

PointPerfect Augmentation service and using all four GNSS constellations. Measurements collected at different times during multiple trials, are summarized statistically through their mean and variance. Accuracy measures are improved by using the augmentation service, as expected: performance obtained using the GALILEO and GPS constellations simultaneously, but without atmospheric and clock correction, are significantly lower than the ones obtained when the full potential of the GNSS receiver is used. Exploitation of the PointPerfect augmentation service noticeably improves the mean of the accuracy values and also reduces their variance, leading to an improved system's reliability. Although requirements of the EMERGE reference use cases are satisfied, the recommendation points towards a configuration with a Multi-constellation/multi-frequency approach plus GNSS Augmentation Service. With the perspective to support high and full automated Connected and Automated Driving (CAD) functions, as detailed in 3GPP and ETSI latest specifications [103], a reduction of the order of magnitude (from metres to centimetres) in the accuracy parameters in the satellite navigation process is more than justified.

One of the advantages of using configurations where all available satellite resources in view are utilised is the ability to select (exclude) the SVs (space vehicles) to be used. This has a direct impact on other satellite navigation KPIs. When using all the potentialities of the receiver, it can be observed doubling of the number of used satellites, as well as an improvement in the Dilution of Precision (DOP) parameters that mitigate the (mathematical) error in the calculation of the position due to the effects of the spatial distribution of the GNSS satellites.

Values of the Integrity, which is a crucial element in automotive applications, may be insufficient for some specific automotive use cases, despite the usage of the *GNSS and PointPerfect* configuration. This is why other sensors such as Lidar, Radar and Video cameras must be incorporated in specific scenarios and applications. In fact, the EMERGE project foresees the integration of sensors (as external components) in the navigation OBU.

J1.5.1.1 Sensor Integration

This section presents the experimental results obtained using the sensor fusion (SF) algorithm described in Section J1.3. To assess the algorithm's performance under realistic conditions, we conducted tests in an open-sky environment near our laboratory, ensuring clear visibility. We intentionally opted for this setting to maintain flexibility in test conditions and trajectory definition. We decided to focus our tests

on real-world scenarios characterized by urban signal degradation, therefore augmenting the GNSS data further would not have provided significant insights. Our aim was to underscore the practical utility of INS in complementing GNSS signals, particularly in scenarios where augmentation may not substantially contribute to improving accuracy or robustness. Hence, we opted to concentrate on the primary GNSS signals from GPS and Galileo, augmented by the INS data. To replicate inaccuracies typically encountered in urban settings, a fault injection operation was applied to the GNSS–only signal (GPS + GALILEO). To simulate the effect of random variations and inaccuracies in the GNSS signal, we employed a Brownian motion process, that generated random additive variations for latitude, longitude, and altitude. In this way, we obtained the estimates of position provided by a degraded GNSS. Subsequently, we evaluated the impact of data fusion in terms of accuracy, stability, and response to errors that affected GNSS without augmentation and IMU sensors.

The tests were conducted near the Abruzzo Technopole, located at Strada Statale 17 Loc. Boschetto di Pile, 67100 L'Aquila (AQ), Italy.

Figures J1.8-J1.12 illustrate the five test scenarios under consideration. Each figure depicts the starting point of the vehicle detected by each sensor with a circle. The dashed line represents the ground truth, derived from the GNSS signal (all constellations) with PP corrections applied every 5 s. The solid purple line represents the trajectory obtained with GNSS-only, while the solid blue line represents the trajectory obtained from the sensor fusion algorithm, integrating IMU/GNSS data. A preliminary analysis reveals that in each scenario, the trajectory from GNSS-only, which has undergone degradation, deviates further from the ground truth compared to the trajectory returned by the data fusion algorithm.

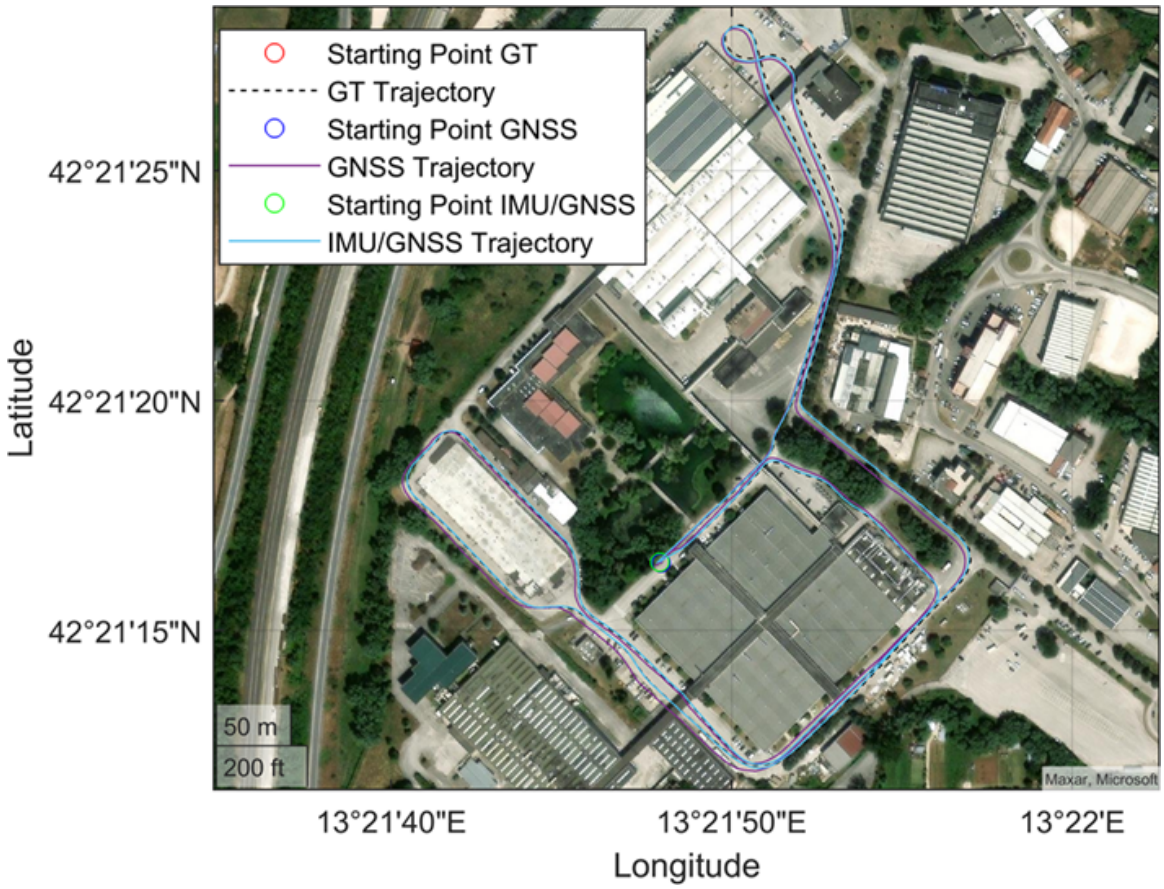

Figure J1.8: Scenario 1. Satellite view

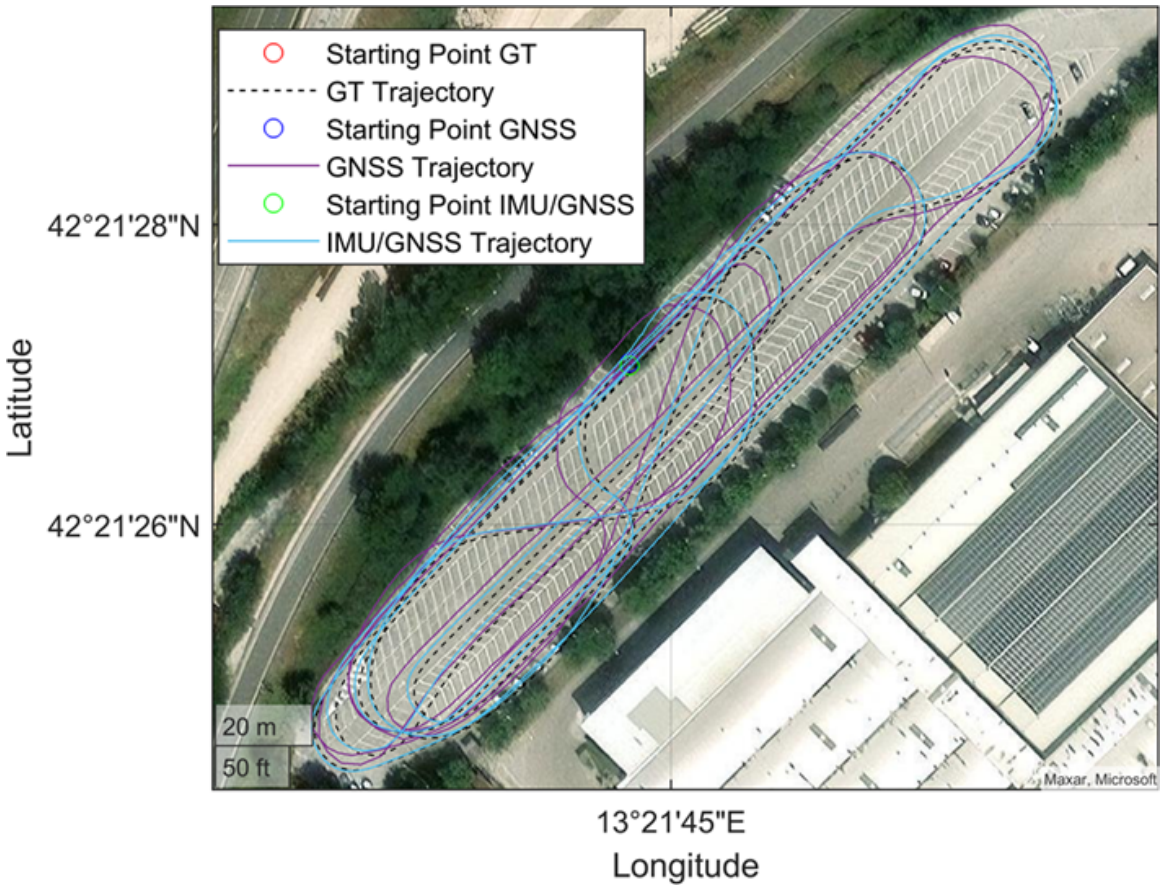

Figure J1.9: Scenario 2. Satellite view

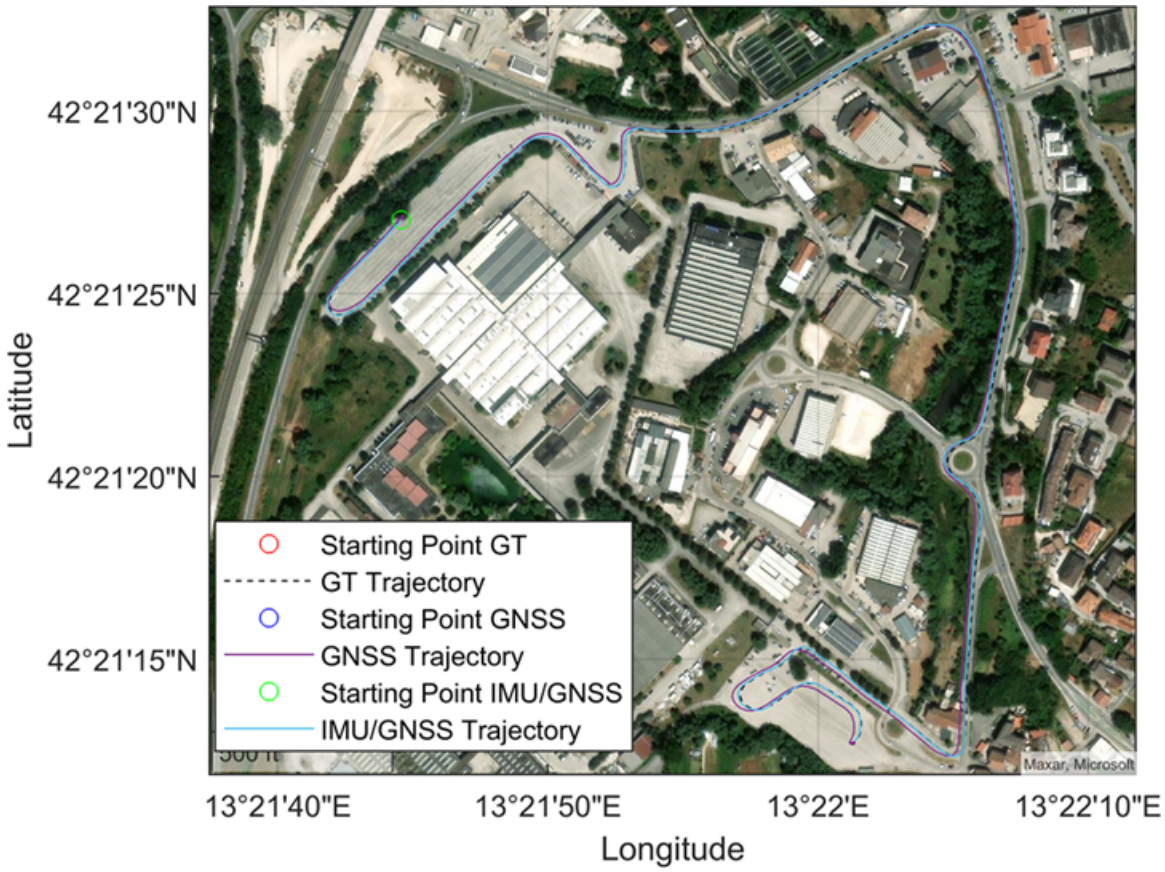

Figure J1.10: Scenario 3. Satellite view

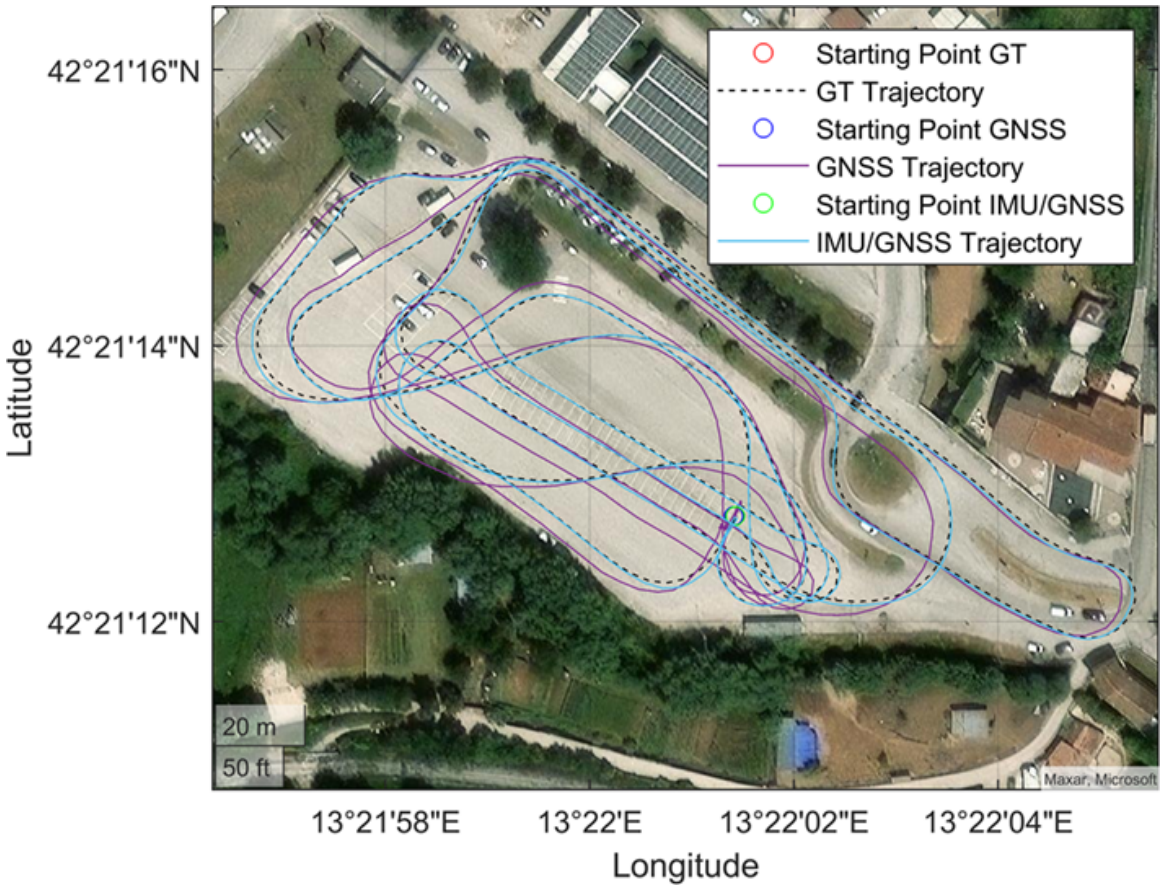

Figure J1.11: Scenario 4. Satellite view

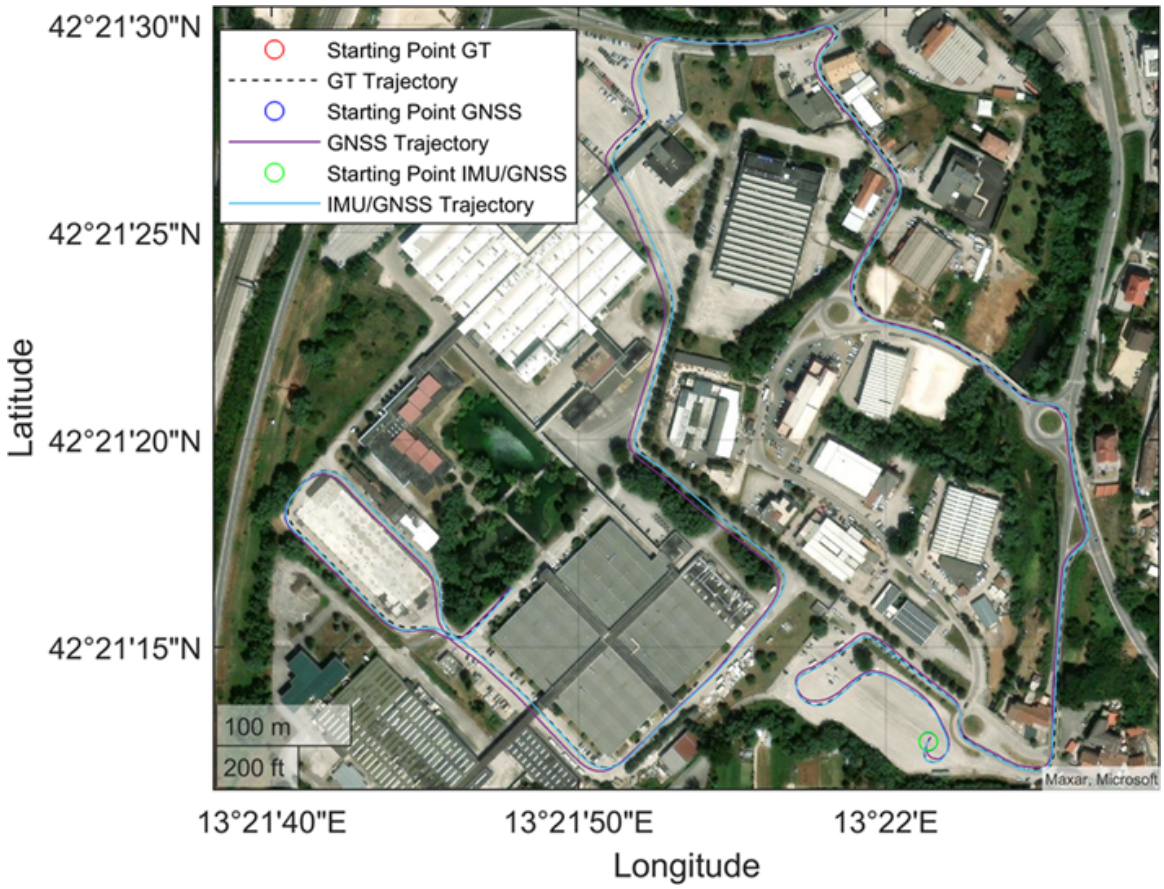

Figure J1.12: Scenario 5. Satellite view

Figures J1.13-J1.17 compare, on one side, the graphs of latitude, longitude, and altitude, and on the other side, the error on the same calculated as the distance from the ground truth. The gray line corresponds to GNSS-only, the blue line represents the result of IMU/GNSS integration, and the dashed line represents the ground truth. Here, it becomes even more evident how the algorithm reduces disturbances introduced by GNSS and minimizes error.

This observation is also numerically confirmed in Table J1.2 which presents the Root Mean Square Error data calculated with respect to the ground truth for the position obtained with GNSS–only and for the position estimated by the SF algorithm.

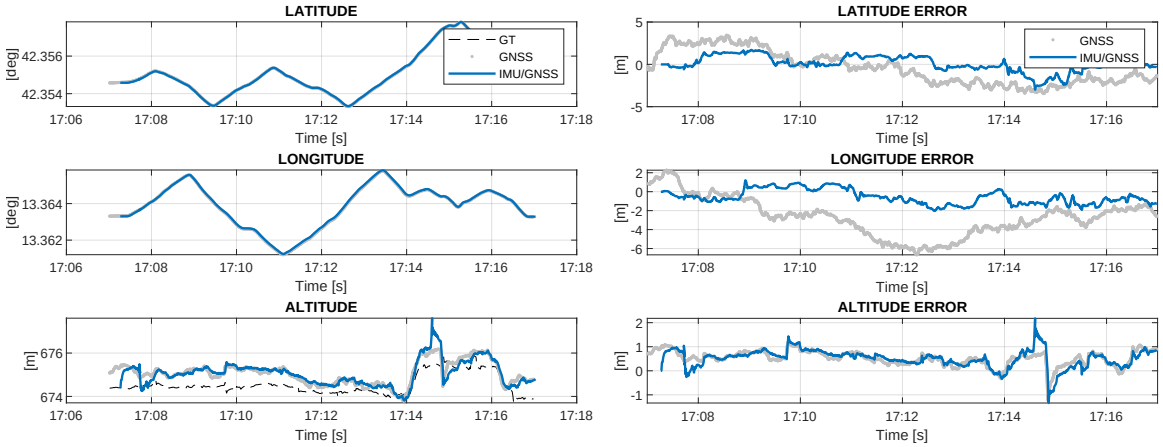

Figure J1.13: Scenario 1. Latitude, longitude and altitude

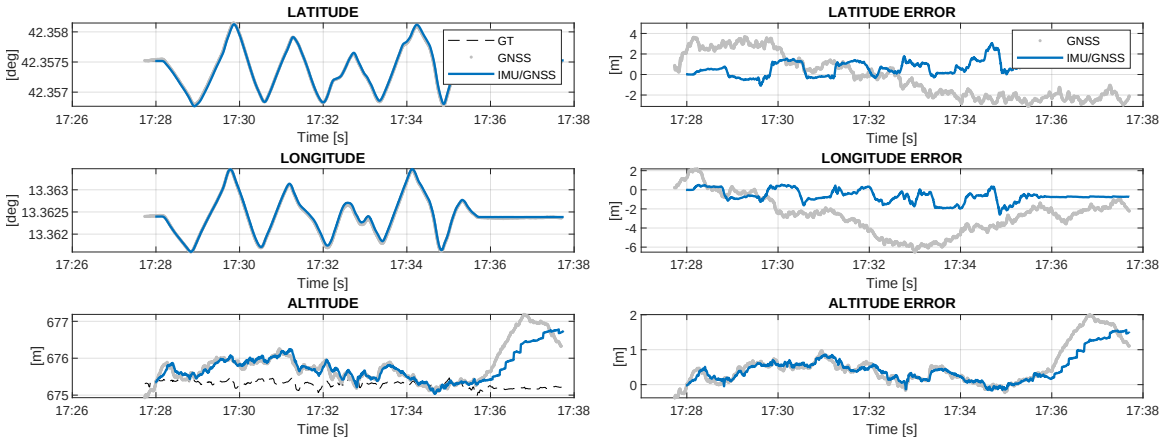

Figure J1.14: Scenario 2. Latitude, longitude and altitude

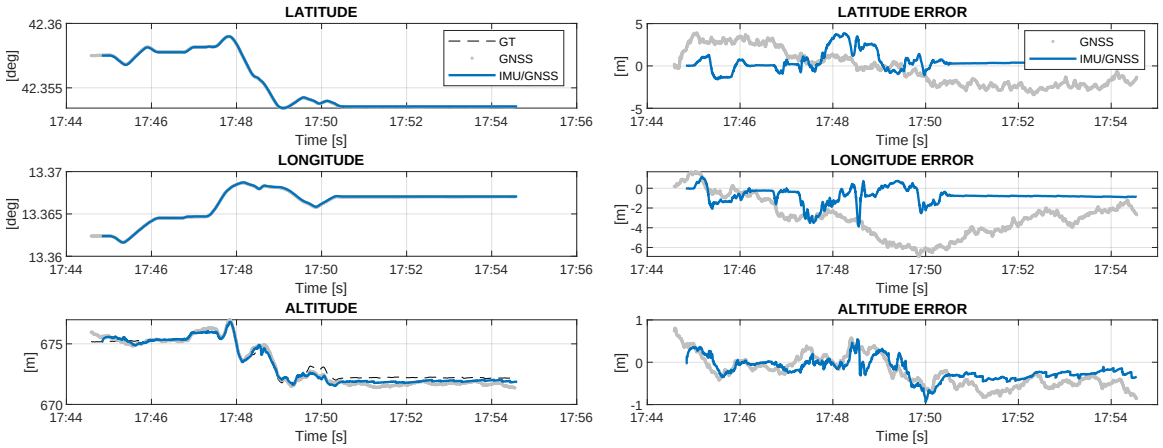

Figure J1.15: Scenario 3. Latitude, longitude and altitude

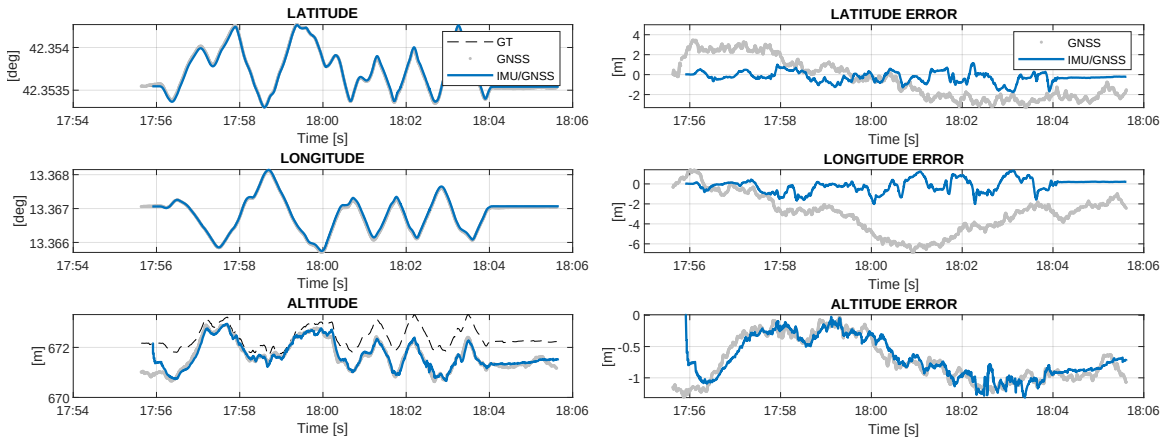

Figure J1.16: Scenario 4. Latitude, longitude and altitude

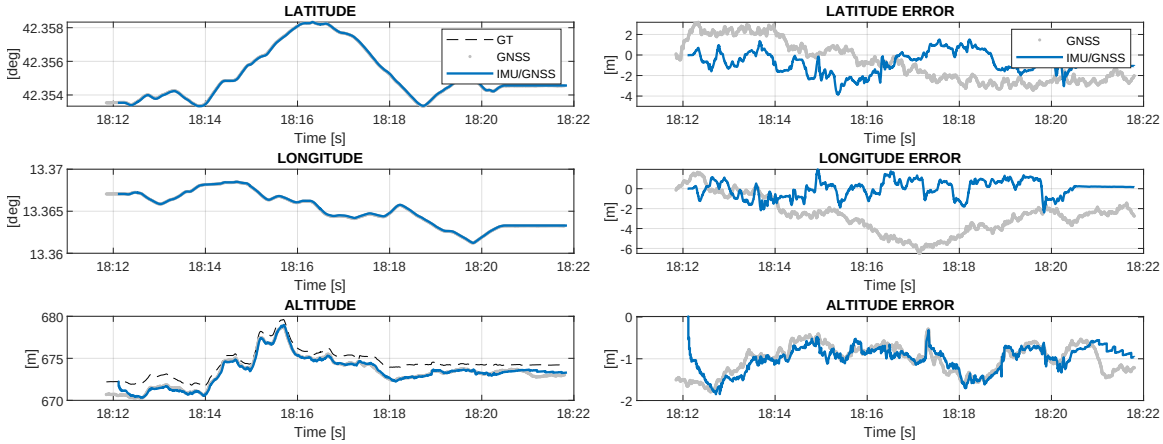

Figure J1.17: Scenario 5. Latitude, longitude and altitude

Table J1.2: Comparison between the estimated trajectory derived from degraded GNSS-only signals, simulating an urban environment, and the trajectory estimated by the SF algorithm using the Root Mean Square Error (RMSE) method, calculated relative to the ground truth

Scenario	**RMSE GNSS [m]**	**RMSE SF [m]**
1	X = 1.9903, Y = 3.0736, Z = 1.5573	X = 0.8272, Y = 0.8704, Z = 0.8553
2	X = 2.0940, Y = 3.0223, Z = 1.5049	X = 0.5824, Y = 0.9131, Z = 1.0307
3	X = 1.7010, Y = 3.3717, Z = 1.7452	X = 0.7041, Y = 1.1537, Z = 0.8255
4	X = 1.6152, Y = 3.3778, Z = 1.8267	X = 0.4870, Y = 0.6889, Z = 0.8041
5	X = 1.6983, Y = 3.0971, Z = 1.8797	X = 0.8312, Y = 0.8393, Z = 1.4352

These results indicate a substantial discrepancy between the positions estimated via GNSS and the actual positions measured in the field, and demonstrate that the

combined use of IMU and GNSS significantly improved the accuracy of position estimation.

The velocity plots in Figures J1.18-J1.22 confirm the previous analyses for the same scenarios, with a perfect overlap of the velocity estimates on all three axes for the ground truth, GNSS-only, and the result of data fusion. On the other side, the error on the three axes fluctuates slightly around zero, confirming the algorithm's good performance.

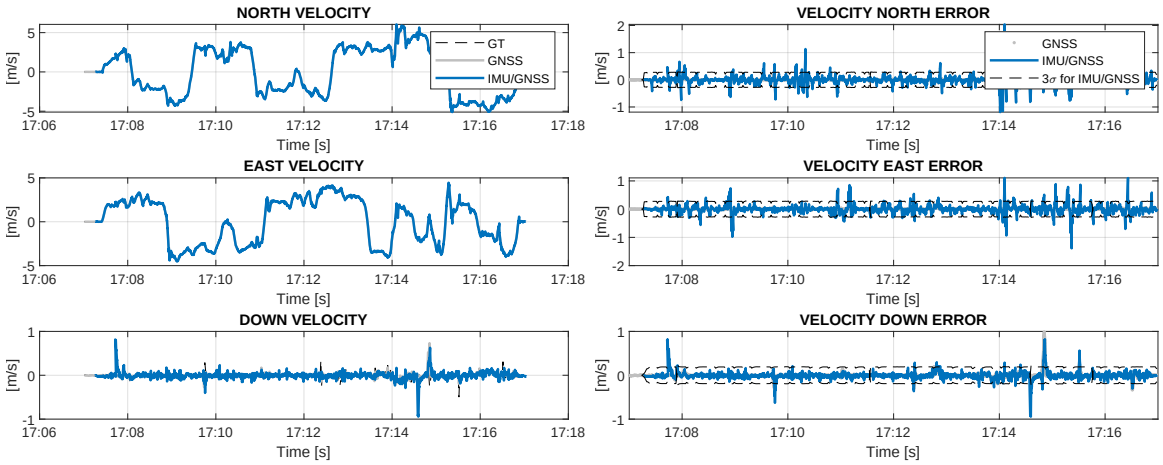

Figure J1.18: Scenario 1. NED velocity

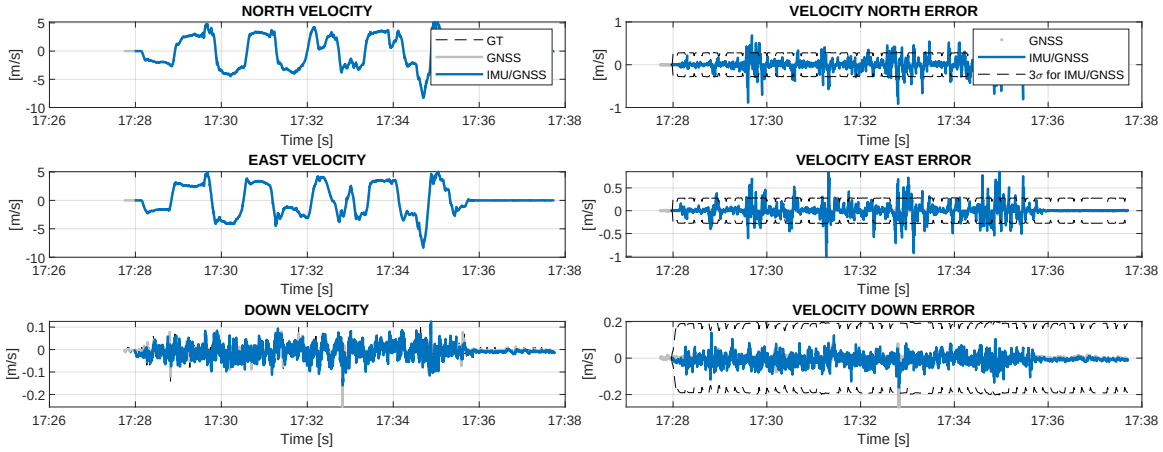

Figure J1.19: Scenario 2. NED velocity

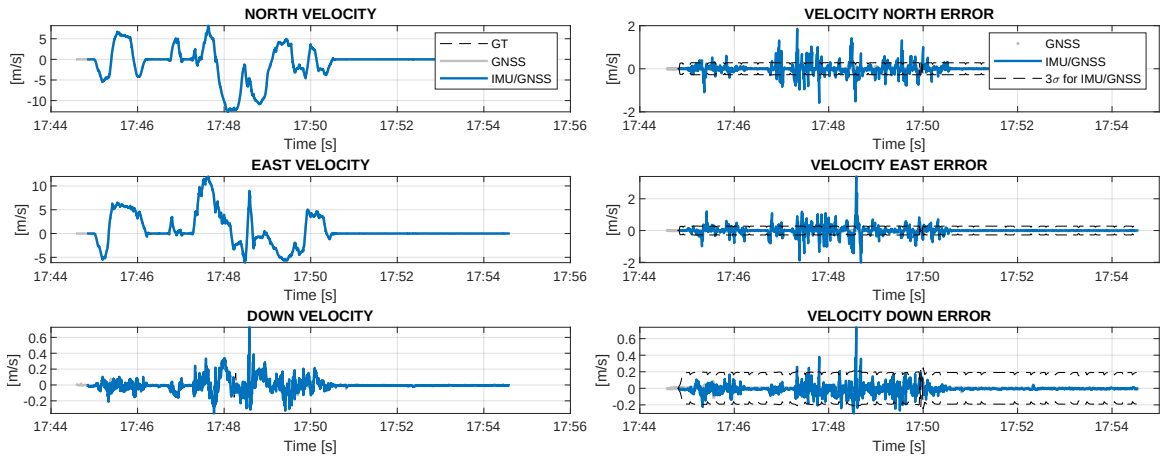

Figure J1.20: Scenario 3. NED velocity

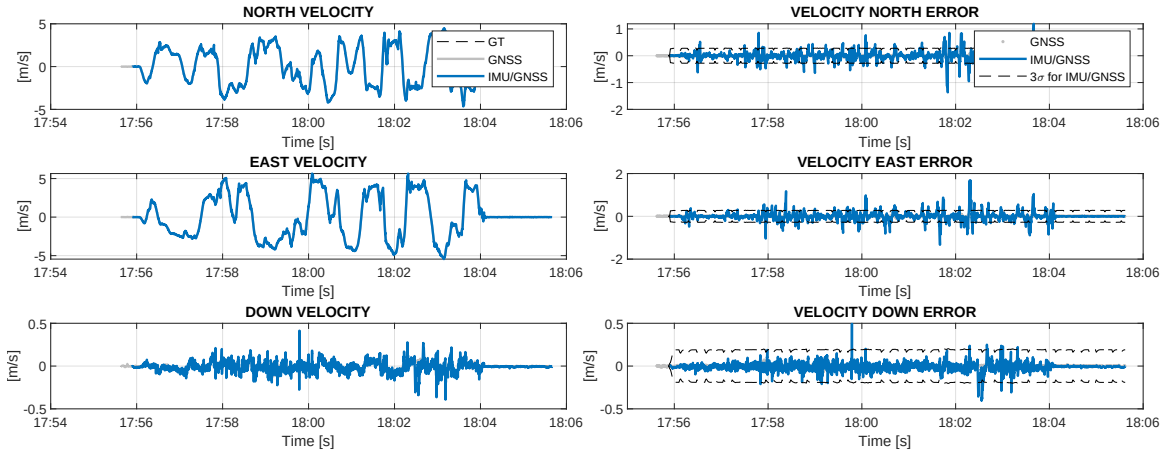

Figure J1.21: Scenario 4. NED velocity

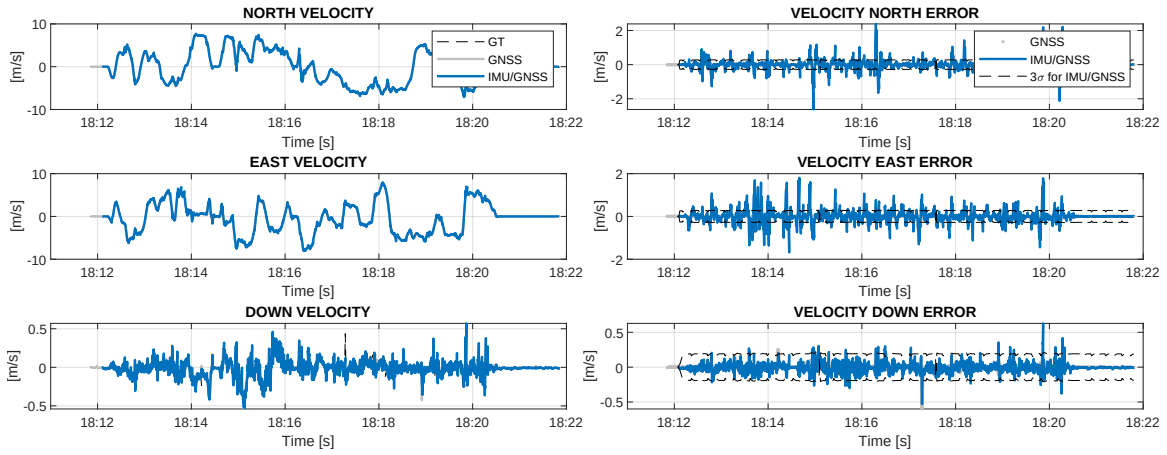

Figure J1.22: Scenario 5. NED velocity

J1.6 Conclusions

This paper presents the OBU architecture for the EMERGE project, and the implementation of the navigation system, based on a multi–sensor approach.

The proposal of a concrete architecture for OBU development provides a starting point in the automotive field for other application areas and services. Although in the test–bed used for this work most of the processing has been performed locally, the presented architecture can be applied to systems that foresee a microservice approach, cloud computing [104], and/or edge computing. This transformation entails moving some elements (mainly from the core layer of the onboard system architecture, i.e., layer 4 in Figure J1.1) out of the OBU and enhancing elements dedicated to communication, according to specific performance requirements. The modular architecture has helped in using the agile methodology for developing the product, facilitating the selection process of HW components and technologies, and error debugging.

Experimental tests conducted on inertial and GNSS sensors have demonstrated their accuracy, that satisfies performance requirements of the EMERGE project and justifies their use in real automotive scenarios. The GNSS receiver provides excellent performance when integrated with the u–blox augmentation system, i.e., PointPerfect, that enhances the accuracy of the satellite navigation system from metres to centimetres. This work uses the u–blox commercial IP plan, which provides corrections through an MQTT broker. The subscription to the root topic provides to the experiment all the set of corrections—i.e., orbits, bias, atmosphere, and clock—whenever they are available. The effectiveness of the multi–sensor approach has also been validated in challenging scenarios like urban contexts. In fact, tests documented in Section J1.5.1.1 show that merging inertial and satellite data provides a more accurate estimate of the trajectory, particularly when the GNSS is degraded.

Implementing an OBU in the automotive context also involves the power analysis of the system. The Power Management and UPS systems guarantee the operating autonomy needed by the OBU to conclude and store the states of the active processes.

J1.7 Appendix: Kalman Filter (NED)

State prediction matrix:

$$\Phi = \Phi_{INS} = \begin{bmatrix} I + F_{11}\tau & F_{12}\tau & F_{13}\tau & 0 & \hat{C}_b^n\tau \\ F_{21}\tau & I + F_{22}\tau & F_{23}\tau & \hat{C}_b^n\tau & 0 \\ 0 & F_{32}\tau & I + F_{33}\tau & 0 & 0 \\ 0 & 0 & 0 & I & 0 \\ 0 & 0 & 0 & 0 & I \end{bmatrix} \tag{J1.27}$$

where

$$F_{11} = -[\omega_{in}^n]_{\mathrm{X}} \tag{J1.28}$$

$$F_{12} = \begin{bmatrix} 0 & \frac{-1}{R_E(L_b)+h_b} & 0 \\ \frac{1}{R_N(L_b)+h_b} & 0 & 0 \\ 0 & \frac{\tan(L_b)}{R_E(L_b)+h_b} & 0 \end{bmatrix} \tag{J1.29}$$

$$F_{13} = \begin{bmatrix} \omega_{ie}\sin(L_b) & 0 & \frac{v_{eb,E}^n}{(R_E(L_b)+h_b)^2} \\ 0 & 0 & \frac{-v_{eb,N}^n}{(R_N(L_b)+h_b)^2} \\ \omega_{ie}\cos(L_b) + \frac{v_{eb,E}^n}{(R_E(L_b)+h_b)\cos^2(L_b)} & 0 & \frac{-v_{eb,E}^n\tan(L_b)}{(R_E(L_b)+h_b)^2} \end{bmatrix} \tag{J1.30}$$

$$F_{21} = -[\hat{C}_b^n \hat{f}_{ib}^b]_{\mathrm{X}} \tag{J1.31}$$

$$F_{22} = \begin{bmatrix} \frac{v_{eb,D}^n}{R_N(L_b)+h_b} & 0 & \frac{v_{eb,N}^n}{R_N(L_b)+h_b} \\ \frac{v_{eb,E}^n\tan(L_b)}{R_E(L_b)+h_b} + 2\omega_{ie}\sin(L_b) & \frac{v_{eb,N}\tan(L_b)+v_{eb,D}^n}{R_E(L_b)+h_b} & \frac{v_{eb,E}^n}{R_E(L_b)+h_b} + 2\omega_{ie}\cos(L_b) \\ -\frac{2v_{eb,N}^n}{R_N(L_b)+h_b} & -\frac{2v_{eb,E}^n}{R_E(L_b)+h_b} - 2\omega_{ie}\cos(L_b) & 0 \end{bmatrix} \tag{J1.32}$$

$$F_{23} = \begin{bmatrix} \frac{(v_{eb,E}^n)^2\sec^2(L_b)}{R_E(L_b)+h_b} - 2v_{eb,E}^n\omega_{ie}\cos(L_b) & 0 & \frac{(v_{eb,E}^n)^2\tan(L_b)}{(R_E(L_b)+h_b)^2} - \frac{v_{eb,N}^n v_{eb,D}^n}{(R_N(L_b)+h_b)^2} \\ \begin{pmatrix} \frac{v_{eb,N}^n v_{eb,E}^n\sec^2(L_b)}{R_E(L_b)+h_b} + 2v_{eb,N}^n\omega_{ie}\cos(L_b) \\ -2v_{eb,D}^n\omega_{ie}\sin(L_b) \end{pmatrix} & 0 & \frac{v_{eb,N}^n v_{eb,E}^n\tan(L_b)+v_{eb,E}^n v_{eb,D}^n}{(R_E(L_b)+h_b)^2} \\ 2v_{eb,E}^n\omega_{ie}\sin(L_b) & 0 & \begin{pmatrix} \frac{(v_{eb,E}^n)^2}{(R_E(L_b)+h_b)^2} + \frac{(v_{eb,N}^n)^2}{(R_N(L_b)+h_b)^2} \\ -\frac{2g_0(L_b)}{r_{eS}^e(L_b)} \end{pmatrix} \end{bmatrix} \tag{J1.33}$$

$$F_{32} = \begin{bmatrix} \frac{1}{R_N(L_b)+h_b} & 0 & 0 \\ 0 & \frac{1}{(R_E(L_b)+h_b)\cos(L_b)} & 0 \\ 0 & 0 & -1 \end{bmatrix} \tag{J1.34}$$

$$F_{33} = \begin{bmatrix} 0 & 0 & -\frac{v_{eb,N}^n}{(R_N(L_b)+h_b)^2} \\ \frac{v_{eb,E}^n\sin(L_b)}{(R_E(L_b)+h_b)^2\cos^2(L_b)} & 0 & \frac{v_{eb,E}^n}{(R_N(L_b)+h_b)^2\cos(L_b)} \\ 0 & 0 & 0 \end{bmatrix} \tag{J1.35}$$

Prediction error covariance matrix:

$$Q = Q_{INS} \approx \begin{bmatrix} S_{rg}I & 0 & 0 & 0 & 0 \\ 0 & S_{ra}I & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & S_{bad}I & 0 \\ 0 & 0 & 0 & 0 & S_{bgd}I \end{bmatrix} \tau \tag{J1.36}$$

Where S_{rg} , S_{ra} , S_{bad} , and S_{bgd} denote, respectively, the power spectral densities of the noise of the gyroscope and accelerometer, and the power spectral densities of the bias variations of the gyroscope and accelerometer.

Vector of observations:

$$\delta z_k = \begin{pmatrix} \hat{r}^n_{GNSS} - \hat{r}^n_{eb} - \hat{C}^n_b l^b_{ba} \\ \hat{v}^n_{GNSS} - \hat{v}^n_{eb} - \hat{C}^n_b(\hat{\omega}^b_{ib} \times l^b_{ba}) + [\omega^n_{ie}]_{\times}\hat{C}^n_b l^b_{ba} \end{pmatrix} \tag{J1.37}$$

is given by the difference between the position and velocity solution of the GNSS and the corrected inertial navigation solution, plus a term to account for the displacement between the inertial platform and the GNSS antenna, l^b_{ba}.

Measurement matrix:

$$H^e = \begin{pmatrix} [\hat{C}^n_b l^b_{ba}]_{\times} & 0 & -I & 0 & 0 \\ [\hat{C}^n_b(\hat{\omega}^b_{ib} \times l^b_{ba}) - [\omega^n_{ie}]_{\times}\hat{C}^n_b l^b_{ba}]_{\times} & -I & 0 & 0 & \hat{C}^n_b[l^b_{ba}]_{\times} \end{pmatrix} \tag{J1.38}$$

Part III

Conclusion

Concluding remarks and future directions

This doctoral research focuses on the use of Software Defined Radio (SDR) technology in localization applications, building on the contributions outlined throughout this research journey. This thesis has introduced innovative solutions and methodologies to improve accuracy and reliability in positioning systems by addressing the challenges and demands of modern navigation systems. The versatile use of SDR platforms has allowed the exploration and analysis of complex scenarios ranging from physical signal analysis to implementing complex digital processing algorithms.

The thesis offers a comprehensive overview of the updated characteristics of GNSS signals, emphasizing the relevance of adopting a multi-constellation and multi-frequency approach in navigation solutions. While not entirely new, the proposed SDR-based GNSS receiver incorporates interactions with other localization techniques and presents a flexible modular structure. This general framework includes specific functions for both the front-end and the GNSS software receiver. The practical evaluation of various hardware platforms allows for studying the main issues affecting GNSS signals and highlights the SDR devices' limitations. In this context, the local oscillator's precision was crucial in the acquisition and tracking processes, directly impacting the receiver's performance. Moreover, the study of SDR-based GNSS simulation solutions offers practical test benches for solution validation.

This research contributes to further applications on multi-sensor integration and the use of GNSS augmentation services, which are essential components for enhancing the integrity of a navigation system. The proposed architecture for developing an On-Board Unit (OBU) within the framework of the EMERGE project provides a starting point for applications in the automotive industry and other fields. While the described architecture assumes that most of the processing is done locally (on-board), the processing can also be carried out remotely according to the microservice approach, cloud computing, or edge computing.

Experimental tests conducted with inertial sensors and GNSS receivers have proved their effectiveness, justifying their use in real automotive scenarios. Regarding GNSS, the receiver shows excellent results when complemented with a correction service capable of implementing advanced positioning solutions such as PPP-RTK. The U-blox PointPerfect GNSS augmentation service performs well, improving satellite navigation system accuracy from meters to centimetres. This research uses Ublox's commercial IP plan, which sends corrections via an MQTT broker. The experiment

uses corrections information from a dedicated root topic (MQTT), aggregating all data: orbits, bias, atmosphere, and clock. This process can take approximately 30 seconds between corrections, affecting the real-time performance of the navigation system. To solve this problem, we ensure clock corrections every 5 seconds using a dedicated subscription to the specific clock correction topic. This approach would allow the satellite navigation system to maintain continuous performance, essential in an automotive scenario. Through experimentation and analysis, the integration of augmentation services emerged as a crucial aspect of the research, offering critical corrections for atmospheric and clock errors in GNSS signals. Implementing these services led to tangible improvements in positioning accuracy and reliability, validating the effectiveness of the proposed methodologies.

Furthermore, exploring opportunistic localization techniques using SDR technology, specifically ADS-B signals, demonstrated the potential of using alternative sources in the localization process. In this area, experimental results showed the possibility of enhancing positioning performance, as reported in the literature, by applying advanced channel description techniques and statistical signal processing. Indeed, with the advent of more advanced SDR platforms in the following years, it will be possible to experiment with more advanced processing techniques and algorithms, potentially including artificial intelligence.

In addition, the thesis addressed the issue of interference detection and localization using advanced signal processing techniques and SDR-based distributed sensor networks. The successful identification of interference sources through spectral analysis techniques demonstrated the validity of dedicated systems to support next-generation mobile networks. The experimental results of a small-scale prototype provide a starting point test-bed for developing and exploring new SDR-based services for advanced functionalities such as sensing and localization.

In conclusion, this thesis has made contributions using SDR for localization solutions in different scenarios and application fields, including autonomous navigation, urban mobility, opportunistic positioning using ADS-B signals, and support for next-generation mobile networks.

References

[1] U. N. D. of Economic and S. A. P. Dynamics, "World urbanisation prospects 2018 – 2050," 2018. (accessed on 12 September 2023).

[2] G. GSA, "Market report issue 3," *European Global Navigation Satellite Systems Agency*, 2013.

[3] S. Tadic, A. Favenza, C. Kavadias, and V. Tsagaris, "Ghost: a novel approach to smart city infrastructures monitoring through gnss precise positioning," in *2016 IEEE International Smart Cities Conference (ISC2)*, pp. 1–6, IEEE, 2016.

[4] E. E. S. Agency, "Infrastructure and smart cities," 2019. (accessed on 12 January 2024).

[5] Y. Zhuang, X. Sun, Y. Li, J. Huai, L. Hua, X. Yang, X. Cao, P. Zhang, Y. Cao, L. Qi, J. Yang, N. El-Bendary, N. El-Sheimy, J. Thompson, and R. Chen, "Multi-sensor integrated navigation/positioning systems using data fusion: From analytics-based to learning-based approaches," *Information Fusion*, vol. 95, pp. 62–90, 2023.

[6] A. M. Wyglinski, R. Getz, T. Collins, and D. Pu, *Software-defined radio for engineers*. Artech House, 2018.

[7] R. Akeela and B. Dezfouli, "Software-defined radios: Architecture, state-of-the-art, and challenges," *Computer Communications*, vol. 128, pp. 106–125, 2018.

[8] J. Qadir, N. Ahmed, and N. Ahad, "Building programmable wireless networks: an architectural survey," *EURASIP Journal on Wireless Communications and Networking*, vol. 2014, p. 172, Oct 2014.

[9] R. G. Machado and A. M. Wyglinski, "Software-defined radio: Bridging the analog–digital divide," *Proceedings of the IEEE*, vol. 103, no. 3, pp. 409–423, 2015.

[10] A. Piccioni, "Software-defined radio for spectral analysis and integrated sensing and communications." `https://hdl.handle.net/11697/213799`, July 2023.

[11] A. L. Z. Sosa, R. Alesii, and F. Santucci, "Cross-platform evaluation for software defined radio gnss receiver," in *2022 3rd URSI Atlantic and Asia Pacific Radio Science Meeting (AT-AP-RASC)*, pp. 1–4, 2022.

[12] S. W. Smith *et al.*, "The scientist and engineer's guide to digital signal processing," 1997.

[13] T. M. Inc., "Software-defined radio," 2024.

[14] G. R. org., "Gnu radio," 2024.

[15] K. Borre, D. M. Akos, N. Bertelsen, P. Rinder, and S. H. Jensen, *A software-defined GPS and Galileo receiver: a single-frequency approach*. Springer Science & Business Media, 2007.

[16] J. Sanz, J. Juan, and M. Hernández-Pajares, "Gnss data processing, vol. i: Fundamentals and algorithms," *ESA Communications*, vol. 14, p. 15, 2013.

[17] M. Fantino, L. L. Presti, and M. Pini, "Digital signal processing in gnss receivers," *Handbook of Position Location: Theory, Practice, and Advances*, pp. 975–1022, 2011.

[18] A. Flores, "Navstar gps space segment/navigation user interfaces," techreport IS-GPS-200, U.S. National Coordination Office for Space-Based Positioning, Navigation, and Timing, 200 N. Pacific Coast Highway, Suite 1800El Segundo, CA 90245, Apr. 2021.

[19] G. C. Designer, "Icd code division multiple accessopen service navigation signalin l3 frequency band," techreport 1.0, Russian Space Systems, JSC, 4A Lenin Street, Korolev, Moscow area, 141070, Russia, 2016.

[20] U-blox, "Pointperfect product summary," techreport, U-blox - PP service, Mar. 2021.

[21] E. Board, "Overview of the emerge initiative: Connected, geo-localized and cybersecure vehicles." online.

[22] A. L. Zuriarrain Sosa, V. Ioannucci, M. Pratesi, R. Alesii, C. Albanese, F. Valentini, E. Cinque, A. Martinelli, and M. Brizzi, "Obu for accurate navigation through sensor fusion in the framework of the emerge project," *Applied Sciences*, vol. 14, no. 11, 2024.

[23] J. Reed, *Software Radio: A Modern Approach to Radio Engineering*. USA: Prentice Hall Press, first ed., 2002.

[24] T. Ulversoy, "Software defined radio: Challenges and opportunities," *IEEE Communications Surveys & Tutorials*, vol. 12, no. 4, pp. 531–550, 2010.

[25] D. Medina, *Robust GNSS Carrier Phase-based Position and Attitude Estimation*. PhD thesis, Universidad Carlos III de Madrid, 2022.

[26] C. Fernandez-Prades, J. Arribas, P. Closas, C. Aviles, and L. Esteve, "Gnss-sdr: An open source tool for researchers and developers," in *Proceedings of the 24th International Technical Meeting of The Satellite Division of the Institute of Navigation (ION GNSS 2011)*, pp. 780–794, 2011.

[27] S. Cui, D. Wang, B. Holtkamp, X. Yao, T. Chi, and J. Fang, "A multi-frequency acquisition algorithm for a gnss software receiver," in *IGARSS 2018-2018 IEEE International Geoscience and Remote Sensing Symposium*, pp. 1082–1085, IEEE, 2018.

[28] M. Mehmood, "Improving the detection and estimation processes of acquisition module in global navigation satellite systems (gnss)," in *2015 Fourth International Conference on Aerospace Science and Engineering (ICASE)*, pp. 1–5, IEEE, 2015.

[29] S. Tennina, R. Alesii, F. Tarquini, and F. Graziosi, "Indoor localization solutions to support independent daily life of impaired people at home," in *2016 IEEE International Conference on Communications Workshops (ICC)*, pp. 45–50, 2016.

[30] G. Destino, "Positioning in wireless networks: non-cooperative and cooperative algorithms," 2012.

[31] M. Rabbat and R. Nowak, "Distributed optimization in sensor networks," in *Proceedings of the 3rd international symposium on Information processing in sensor networks*, pp. 20–27, 2004.

[32] J. Liu, Y. Zhang, and F. Zhao, "Robust distributed node localization with error management," in *Proceedings of the 7th ACM international symposium on Mobile ad hoc networking and computing*, pp. 250–261, 2006.

[33] D. Cassioli, M. Z. Win, and A. F. Molisch, "The ultra-wide bandwidth indoor channel: from statistical model to simulations," *IEEE Journal on selected areas in Communications*, vol. 20, no. 6, pp. 1247–1257, 2002.

[34] R. Calvo-Palomino, F. Ricciato, B. Repas, D. Giustiniano, and V. Lenders, "Nanosecond-precision time-of-arrival estimation for aircraft signals with low-cost sdr receivers," in *2018 17th ACM/IEEE International Conference on Information Processing in Sensor Networks (IPSN)*, pp. 272–277, IEEE, 2018.

[35] J. Naganawa and H. Miyazaki, "Aircraft–receiver distance estimation using ads-b signal strength for position verification application," in *2021 IEEE-APS Topical Conference on Antennas and Propagation in Wireless Communications (APWC)*, pp. 178–183, IEEE, 2021.

[36] E. u. Heirbaut and I. u. v. p. Moerman, "Localisation by airplane ads-b signal strength vectors," 2019.

[37] A. L. Z. Sosa, R. Alesii, and F. Santucci, "Opportunistic rss-based localisation using sdr and ads-b system," in *2024 4th URSI Atlantic Radio Science Meeting (AT-RASC)*, pp. 1–4, 2024.

[38] H. Stanislaw and N. Todorov, "Calculation of signal detection theory measures," *Behavior Research Methods, Instruments, & Computers*, vol. 31, pp. 137–149, Mar 1999.

[39] A. Noureldin, T. B. Karamat, and J. Georgy, *Fundamentals of inertial navigation, satellite-based positioning and their integration*. Springer Science & Business Media, 2012.

[40] R. B. Rustamov and A. M. Hashimov, *Multifunctional Operation and Application of GPS*. BoD–Books on Demand, 2018.

[41] D. P. GrovesP, "Inertial, and multisensorintegrated navigationsystems," 2008.

[42] N. El-Sheimy, H. Hou, and X. Niu, "Analysis and modeling of inertial sensors using allan variance," *IEEE Transactions on Instrumentation and Measurement*, vol. 57, no. 1, pp. 140–149, 2008.

[43] P. D. Groves, "Principles of gnss, inertial, and multisensor integrated navigation systems, [book review - chapter 5]," *IEEE Aerospace and Electronic Systems Magazine*, vol. 30, no. 2, pp. 26–27, 2015.

[44] P. D. Groves, "Principles of gnss, inertial, and multisensor integrated navigation systems, [book review - chapter 15]," *IEEE Aerospace and Electronic Systems Magazine*, vol. 30, no. 2, pp. 26–27, 2015.

[45] U. Nations, "Urban population long-run with 2050 projections (owid)." `https://ourworldindata.org/grapher/urban-and-rural-population-2050?tab=table&time=2020..latest&country=~OWID_WRL`, 2018. [Online; accessed 11-January-2022].

[46] C. G. P. a André Ferreira, A. J. C. Moreira, C. Martins, and H. Silva, "Challenges in characterization of GNSS precise positioning systems for automotive," in *Proceedings of the International Conference on Localization and GNSS (ICL-GNSS 2020), Tampere, Finland, June 2nd to 4th, 2020 - Work in Progress Papers* (A. Ometov, J. Nurmi, E. S. Lohan, J. Torres-Sospedra, and H. Kuusniemi, eds.), vol. 2626 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2020.

[47] N. Franconi, S. Sabogal, A. George, M. Hassouneh, J. Mitchell, and C. Wilson, "A novel rf architecture for simultaneous communication, navigation, and remote sensing with software-defined radio," *34th Annual Conference on Small Satellites, August 3-8*, 2020.

[48] T. Helaly and N. Adnani, "A new category of software-defined instrumentation for wireless test," in *2016 IEEE AUTOTESTCON*, pp. 1–8, IEEE, 2016.

[49] M. L. Nicolás, I. Artamonov, and T. Kürner, "Low-cost usrp sdr receiver for the investigation of multipath influence on gnss systems," in *2013 7th European Conference on Antennas and Propagation (EuCAP)*, pp. 2300–2304, IEEE, 2013.

[50] D. Tcherniakovski, K. Veprev, N. Filippov, and A. Tsikhamirava, "Gnss sdr based on multi-channel multi-band multi-system rffe ic," in *Proceedings of the 29th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS+ 2016)*, pp. 120–127, 2016.

[51] E. A. Thompson, N. Clem, I. Renninger, and T. Loos, "Software-defined gps receiver on usrp-platform," *Journal of Network and Computer Applications*, vol. 35, no. 4, pp. 1352–1360, 2012.

[52] R. Di, S. Peng, S. Taylor, and Y. Morton, "A usrp-based gnss and interference signal generator and playback system," in *Proceedings of the 2012 IEEE/ION Position, Location and Navigation Symposium*, pp. 470–478, IEEE, 2012.

[53] B. Kumar and C. Paidimarry, "Improved real time gps rf data capturing for gnss sdr applications," *Gyroscopy and Navigation*, vol. 11, pp. 59–67, 2020.

[54] S.-S. Lin and Y.-H. Li, "A sdr-based gps receiver with low accuracy of local oscillator," in *2021 International Symposium on Intelligent Signal Processing and Communication Systems (ISPACS)*, pp. 1–2, IEEE, 2021.

[55] E. U. A. for the Space Programme (EUSPA), "Os sis icd," techreport 2.0, European Union 2021, Janovského 438/2 170 00 Prague 7 – Holesovice Czech Republic, 2021.

[56] EUROCONTROL, "Eurocontrol seven-year forecast 2023-2029," techreport, European Aviation, https://www.eurocontrol.int/sites/default/files/2023-03/eurocontrol-seven-year-forecast-2023-2029-spring-2023.pdf, Mar. 2023.

[57] SESAR-3-JU, "Single european sky atm research joint undertaking," 2023. Accessed on October 09, 2023.

[58] B. The European Commission: Brussels, "The european commission. implementing regulation (eu) no. 2020/587," Apr. 2020.

[59] ITU-R, "Imt vision – framework and overall objectives of the future development of imt for 2020 and beyond." Report ITU-R M.2083-0, Sep. 2015.

[60] A. Piccioni, A. Marotta, C. Rinaldi, and F. Graziosi, "Enhancing mobile networks for urban air mobility connectivity," *IEEE Networking Letters*, vol. 6, no. 2, pp. 110–114, 2024.

[61] C. Rinaldi, F. Franchi, A. Marotta, F. Graziosi, and C. Centofanti, "On the exploitation of 5g multi-access edge computing for spatial audio in cultural heritage applications," *IEEE Access*, vol. 9, pp. 155197–155206, 2021.

[62] F. Martusciello, C. Centofanti, C. Rinaldi, and A. Marotta, "Edge-enabled spatial audio service: Implementation and performance analysis on a mec 5g infrastructure," in *2023 4th International Symposium on the Internet of Sounds*, pp. 1–8, 2023.

[63] C. Centofanti, A. Marotta, C. Rinaldi, F. Franchi, D. Cassioli, and F. Graziosi, "Improved dash video streaming performance by mec-enabled optical access," in *2021 Asia Communications and Photonics Conference (ACP)*, pp. 1–3, 2021.

[64] A. Ijaz, L. Zhang, M. Grau, A. Mohamed, S. Vural, A. U. Quddus, M. A. Imran, C. H. Foh, and R. Tafazolli, "Enabling massive iot in 5g and beyond systems: Phy radio frame design considerations," *IEEE Access*, vol. 4, pp. 3322–3339, 2016.

[65] A. Nauman, Y. A. Qadri, M. Amjad, Y. B. Zikria, M. K. Afzal, and S. W. Kim, "Multimedia internet of things: A comprehensive survey," *IEEE Access*, vol. 8, pp. 8202–8250, 2020.

[66] L. D'Errico, F. Franchi, F. Graziosi, A. Marotta, C. Rinaldi, M. Boschi, and A. Colarieti, "Structural health monitoring and earthquake early warning on 5g urllc network," in *2019 IEEE 5th World Forum on Internet of Things (WF-IoT)*, pp. 783–786, 2019.

[67] F. Franchi, A. Marotta, C. Rinaldi, F. Graziosi, and L. D'Errico, "Iot-based disaster management system on 5g urllc network," in *2019 International Conference on Information and Communication Technologies for Disaster Management (ICT-DM)*, pp. 1–4, 2019.

[68] D. Chen, J. Yang, J. Wu, H. Tang, and M. Huang, "Spectrum occupancy analysis based on radio monitoring network," in *2012 1st IEEE International Conference on Communications in China (ICCC)*, pp. 739–744, 2012.

[69] A. Mariani, A. Giorgetti, and M. Chiani, "Robust detection with low-complexity sdrs: A pragmatic approach," in *2018 IEEE 29th Annual International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC)*, pp. 1–6, 2018.

[70] A. Piccioni, R. Alesii, F. Santucci, and F. Graziosi, "Sdr-based ground target for identification and tracking through satellite sar systems," in *2021 IEEE Aerospace Conference (50100)*, pp. 1–10, 2021.

[71] A. Piccioni, R. Alesii, F. Santucci, and F. Graziosi, "Sdr sar target: Corner reflector and communication," in *2022 3rd URSI Atlantic and Asia Pacific Radio Science Meeting (AT-AP-RASC)*, pp. 1–4, 2022.

[72] A. Piccioni, R. Alesii, F. Santucci, and F. Graziosi, "Satellite sar testing framework for integrated sensing and communication," in *2024 4th URSI Atlantic Radio Science Meeting (AT-RASC)*, pp. 1–4, 2024.

[73] C. Weber, M. Peter, and T. Felhauer, "Automatic modulation classification technique for radio monitoring," *Electronics Letters*, vol. 51, no. 10, pp. 794–796, 2015.

[74] H. Stanislaw and N. Todorov, "Calculation of signal detection theory measures," *Behavior Research Methods, Instruments, & Computers*, vol. 31, pp. 137–149, Mar 1999.

[75] H. Kwasme and S. Ekin, "Rssi-based localization using lorawan technology," *IEEE Access*, vol. 7, pp. 99856–99866, 2019.

[76] K. Vasudeva, B. S. Çiftler, A. Altamar, and I. Guvenc, "An experimental study on rss-based wireless localization with software defined radio," in *WAMICON 2014*, pp. 1–6, 2014.

[77] A. Piccioni, R. Alesii, F. Santucci, and F. Graziosi, "Software-defined corner reflector for satellite sar systems," in *2022 IEEE Aerospace Conference (AERO)*, pp. 1–7, 2022.

[78] R. Faria, L. Brito, K. Baras, and J. Silva, "Smart mobility: A survey," in *2017 International Conference on Internet of Things for the Global Community (IoTGC)*, pp. 1–8, 2017.

[79] S. Paiva, M. A. Ahad, G. Tripathi, N. Feroz, and G. Casalino, "Enabling technologies for urban smart mobility: Recent trends, opportunities and challenges," *Sensors*, vol. 21, no. 6, 2021.

[80] P. Ribeiro, G. Dias, and P. Pereira, "Transport systems and mobility for smart cities," *Applied System Innovation*, vol. 4, no. 3, 2021.

[81] C. Xiang, C. Feng, X. Xie, B. Shi, H. Lu, Y. Lv, M. Yang, and Z. Niu, "Multi-sensor fusion and cooperative perception for autonomous driving: A review," *IEEE Intelligent Transportation Systems Magazine*, vol. 15, no. 5, pp. 36–58, 2023.

[82] D. J. Yeong, G. Velasco-Hernandez, J. Barry, and J. Walsh, "Sensor and sensor fusion technology in autonomous vehicles: A review," *Sensors*, vol. 21, no. 6, 2021.

[83] H. MAILKA, M. Abouzahir, and M. Ramzi, "An efficient end-to-end ekf-slam architecture based on lidar, gnss, and imu data sensor fusion for autonomous ground vehicles," *Multimedia Tools and Applications*, vol. 83, no. 18, pp. 56183–56206, 2024.

[84] J. Santa, L. Bernal-Escobedo, and R. Sanchez-Iborra, "On-board unit to connect personal mobility vehicles to the iot," *Procedia Computer Science*, vol. 175, pp. 173–180, 2020.

[85] N. Ganeshkumar and S. Kumar, "Obu (on-board unit) wireless devices in vanet (s) for effective communication—a review," *Computational Methods and Data Engineering: Proceedings of ICMDE 2020, Volume 2*, pp. 191–202, 2020.

[86] L. He, Z. Deng, and J. Huang, "Navigation and communication platform for on board unit of logistics traffic," in *2009 Joint Conferences on Pervasive Computing (JCPC)*, pp. 305–308, 2009.

[87] N. Schindler, "On board unit for the european electronic tolling service," in *Proceedings 9th ITS European Congress, Dublin. ERTICO (ITS Europe)*, 2013.

[88] J. Santa, A. F. G. Skarmeta, and B. Ubeda, "An embedded service platform for the vehicle domain," in *2007 IEEE International Conference on Portable Information Devices*, pp. 1–5, 2007.

[89] N. Zhu, J. Marais, D. Betaille, and M. Berbineau, "Gnss position integrity in urban environments: A review of literature," *IEEE Transactions on Intelligent Transportation Systems*, vol. 19, no. 9, pp. 2762–2778, 2018.

[90] T. G. Reid, S. E. Houts, R. Cammarata, G. Mills, S. Agarwal, A. Vora, and G. Pandey, "Localization requirements for autonomous vehicles," *arXiv preprint arXiv:1906.01061*, 2019.

[91] G. Di Sciullo, L. Zitella, E. Cinque, F. Santucci, M. Pratesi, and F. Valentini, "Experimental validation of c-v2x mode 4 sidelink pc5 interface for vehicular communications," in *2022 61st FITCE International Congress Future Telecommunications: Infrastructure and Sustainability (FITCE)*, pp. 1–6, 2022.

[92] W. Tiberti, R. Civino, N. Gavioli, M. Pugliese, and F. Santucci, "A hybrid-cryptography engine for securing intra-vehicle communications," *Applied Sciences*, vol. 13, no. 24, 2023.

[93] U-blox, "Zed-f9p high precision gnss module integration manual," techreport, U-blox, Aug. 2023.

[94] U-blox, "U-blox services product overview: Iot location-as-a-service," techreport, U-blox, 2023.

[95] Xsens, "Mti-630 ahrs datasheet," techreport, Xsense - Movella, May 2023.

[96] Xsens, "Mti 600-seriesdatasheet: Imu, vru, ahrs and gnss/ins," techreport, Xsense - Movella, June 2023.

[97] MQTT, "Mqtt version 5.0 oasis standard," techreport, MQTT Version 5.0, March 2023.

[98] Sixfab, "Sixfab raspberry pi 3g-4g/lte base hat datasheet: v1.0," techreport, Sixfab, March 2019.

[99] Sixfab, "Sixfab api documentation," techreport, Sixfab, 2023.

[100] Node-RED, "Low-code programming for event-driven. node-red documentation," techreport, Node-RED, March 2023.

[101] RaspberryPi, "Raspberry pi 4 model b datasheet: release 1," techreport, raspberrypi Ltd., June 2019.

[102] Drotek, "Sirius rtk gnss rover (f9p): Compact and high-precision l1/l2 gnss rover based on u-blox zed-f9p (gps / glonass / beidou / galileo)," techreport, Drotek, March 2020.

[103] ETSI, "Service requirements for enhanced v2x scenarios (3gpp ts 22.186 version 17.0.0 release 17)," techreport, ETSI TS 122 186 V17.0.0. 5G, April 2022.

[104] H. J. Desirena Lopez, M. Siller, and I. Huerta, "Internet of vehicles: Cloud and fog computing approaches," in *2017 IEEE International Conference on Service Operations and Logistics, and Informatics (SOLI)*, pp. 211–216, 2017.