POLITECNICO
MILANO 1863

SCUOLA DI INGEGNERIA INDUSTRIALE
E DELL'INFORMAZIONE

O-RAN-Based Real-Time Positioning in 5G Networks: A Preliminary Analysis Using Uplink SRS

Tesi di Laurea Magistrale in
Telecommunications Engineering - Ingegneria delle telecomunicazioni

Author: **Claudia Baz Alvarez**

Student ID: 10916443
Advisor: Prof. Ilario Filippini
Co-advisors: Viola Bernazzoli
Academic Year: 2022-24

Abstract

This thesis introduces an innovative 5th generation (5G) localization approach leveraging Sounding Reference Signal (SRS) within the Open Radio Access Network (O-RAN) framework. The system employs a trilateration xApp on the near-RT RIC to achieve real-time User Equipment (UE) positioning, using real-world SRS signals fed into gNB emulators functioning as 5G base stations. Experimental results demonstrated a final UE localization error of approximately 2 meters under optimal conditions, unlike GNSS, which suffers from limited accuracy in dense urban and indoor settings. The proposed 5G-based solution evidences the potential of SRS-based positioning within 5G architectures. Further, this work explores the potential of programmable networks for real-time localization, opening avenues for future advancements in network-driven positioning systems.

Keywords: 5G localization, SRS, O-RAN, trilateration, near-RT RIC

Contents

Introduction

In the early 90s, Mobile Radio Networks (MRNs) took off as a consumer product with the 2th generation (2G). Nowadays, MRNs are guided by the 5G standard, which is expected to become the dominant access technology, reaching 5.6 billion subscriptions by 2029. North America, North/East Asia, and Western Europe lead the prospects, whereby in 2029, almost 80% of the traffic should come from 5G. India, nonetheless, should have 60% of its mobile traffic running through 5G by 2029 [18].

In this context, 5G's potential extends beyond high-speed data transfer, establishing it as an ideal framework for real-time localization. With inherent Radio frequency (RF) capabilities and dense station architecture, it presents a natural environment to implement localization techniques based on ranging. Unlike Global Navigation Satellite System (GNSS), which lacks effectiveness indoors or in dense urban areas due to its reliance on line-of-sight, the 5G stations can act as localization anchors closer to the final user.

The current 3rd Generation Partnership Project (3GPP) releases already focus on the enhanced version of 5G (Release 18 [2]) and program the 6th generation (6G) for 2026 [17]. Posterior 3GPP releases, 16 and 17, which already focus on 5G, introduce various location technologies to support regulatory and commercial use cases [3]. The increasing interest in MRN as a localization framework provides expectations of real-time positioning with high accuracy, high availability, and low latency, compared to the traditional technologies.

Furthermore, 5G distinguishes itself as a generation recognized for its innovative focus on programmability and advanced virtualization technologies. The rise of new organizations, such as O-RAN and O-RAN Software Community (OSC), evidences this evolution leap. Its new interoperability and open Radio Access Network (RAN) interfaces, the RAN programmability, and added cloud-based technologies introduce more speed and less latency. In addition, the 5G O-RAN RAN Intelligent Controllers (RICs) and xApps provide new possibilities for real-time applications, enhancing network control and performance.

In the context of these new 5G improvements, this thesis aims to contribute to the innovation of one emerging application: a user 5G localization system based on RAN reference signals. Specifically, the SRS, a signal sent by the UE, which was originally designed for

link adaptation and is here leveraged for positioning purposes.

The localization framework presented is built on trilateration techniques using these reference signals. The SRSs sequences are real signals captured in a 5G Standalone (SA) environment by a previous study [7]. These are then injected into three 5G base stations (gNBs) emulators and, utilizing cross-correlation of SRS sequences, the system calculates the time delay, thereby estimating distances to localize user equipment UE, reaching a final error of approximately 2 meters in the best cases. This demonstrates the potential effectiveness of SRS-based positioning within 5G architectures.

The thesis is then organized as follows: chapter 1 introduces the 5G architecture, its fundamental components based on the comparison with 4th generation (4G), the SRS within the physical layer and their original purpose for link adaptation, to describe after later how they are use in localization; chapter 2 provides an overview of localization technologies focusing on range-based methods, and examines trilateration techniques and their limitations; chapter 3 details the implementation of the O-RAN UE-Positioning Micro-Service, which leverages SRS-based ranging for real-time localization, it describes the infrastructure outlining how distance measurements are derived from SRS signals and processed to estimate UE positions through trilateration; finally chapter 4 presents the experimental results of the localization the system comparing the performance of the Kalman filter and Non-linear Least-squares (NLS) algorithms in terms of positioning accuracy.

1 | 5G New Radio background

To fully grasp the nuances of the localization system proposed, it is paramount to understand the basics MRNs architecture, specifically the 5G one.

MRNs became complex systems with many heterogeneous devices and protocols, divided into different networks and utility planes. Normally, they are described as the evolution of the past generation, so for 5G, this means understanding the 4G.

In this section, the 5G architecture is explained based on a comparison with 4G. This helps identify the benefits of the new upgrades and analyze how both generations overlap, their backward compatibility, and the foundations for the localization system.

Firstly, there is an overview of their general architecture, focusing on 5G. In sequence, the SRS is explained in detail, and its original purpose within the 5G. How this signal is later used in the localization is reserved for further chapters.

1.1. 5G architecture

They are built on two main modules: the RAN, composed of distributed nodes and radio cells, and the Core Network (CN), a collection of functions that store and process the signaling information, rule the Quiality of Service (Qos) flows and control the external connections.

It is a natural separation of concerns, as UEs communicate with the RAN through wireless links and native radio protocols. In contrast, the CN works with a packet approach and technologies from the Transmission Control Protocol (TCP)/Internet Protocol (IP) stack.

In addition, UEs are in constant reconnection and motion. The MRN has to track them, manage the wireless environment, and control the routes and authorization while keeping the communication stable. For this, there is another division of concerns in operation planes: the User Plane (UP), where the actual data is routed, and the Control Plane (CP), which manages the signaling for controlling the business logic and communications.

Figure 1.1 shows the basic structure of 4G where there is the division in RAN, or Long

Term Evolution (LTE), CN, or Evolved Packet Core (EPC), and CP-UP.

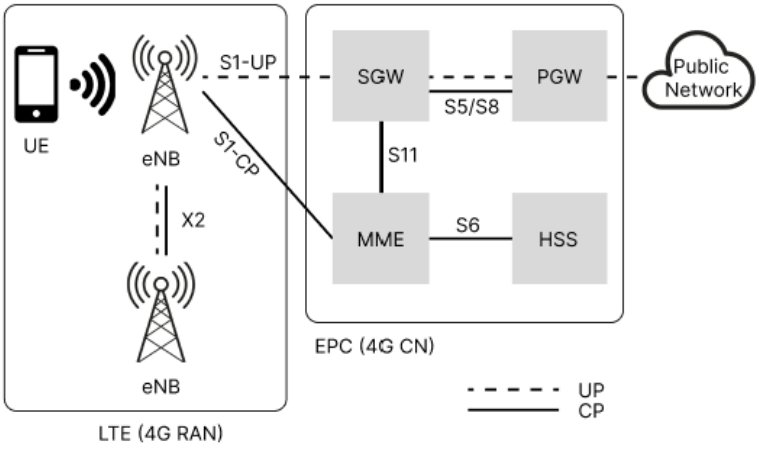


Figure 1.1: 4G architecture

1.1.1. EPC to 5G CN

The EPC and the Evolved Universal Terrestrial Radio Access Network (e-UTRAN) are the CN and the RAN of 4G, respectively [15].

The EPC brought the all-IP paradigm to CNs, dropping the circuit-switched services [13]. The communication is done over an underlying IP-based transport network [15]. Hence, the new 5G CN has evolved hand-to-hand with the Internet stack thanks to this approach. Even more, new virtualization, cloud, and slicing technologies are introduced.

The 5G CN can be understood as an evolution of the 4G one. The EPC is built by several modules, called core functions, which are then leveraged by the 5G CN. For example, the Mobility Management Entity (MME) handles UE mobility, authentication, and security; the Serving Gateway (S-GW) and the Packet Data Network Gateway (P-GW) manage routing; and the Home Subscriber Server (HSS) hold UE data used by the entire network. The 5G CN then incorporates new functions, modifies some previous ones, and standardizes communication between them, all of which makes it more complex compared to the EPC.

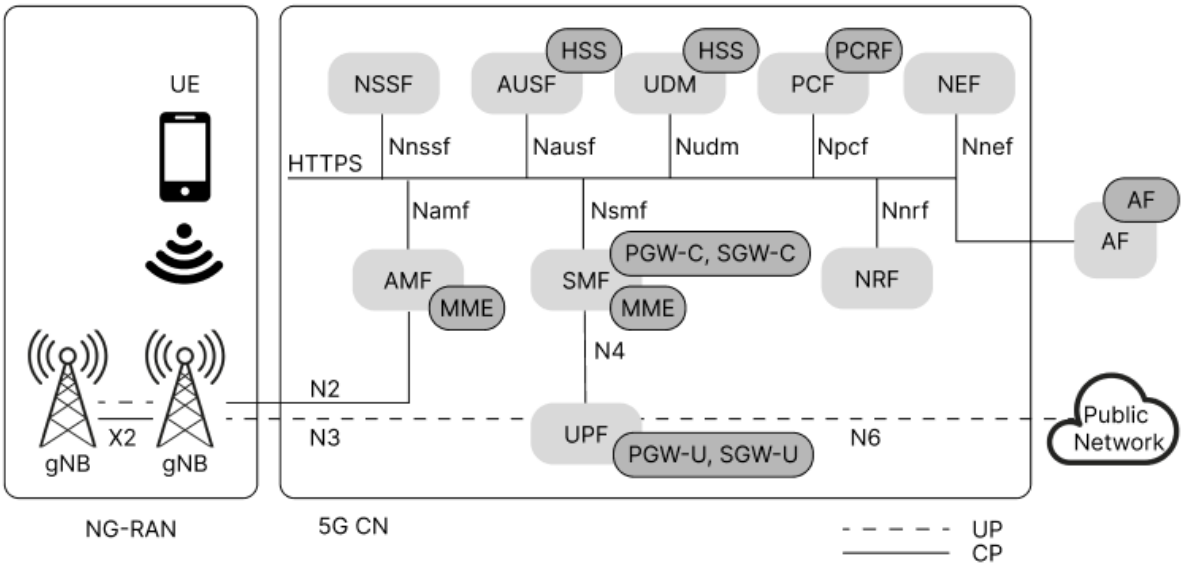


Figure 1.2: 5G CN based on EPC main functions

To route information from the RAN, the CN has to associate virtual tunnels for each UE connection. When a tunnel is ready (CP), the device can start to send and receive actual data (UP). The CN functions work together to build this structure, see Figure 1.2.

In the 5G CN, all the functions are virtualized and hosted in cloud environments. They communicate in a service-based manner, which keeps the number of network procedures manageable [15]. This allows network automation, more flexibility, less complexity, and more cost-effective upgrades [15].

Finally, the 5G CN allows non-3GPP access and supports data networks with other protocols such as Ethernet [15]. This opens the context to machine-type communications, lower latencies, and new technologies for indoor RANs, such as Wi-Fi.

1.1.2. LTE to 5G NG-RAN

The RAN connects the UEs, or devices, with the CN. Its primary element, the Base Station (BS), is the first access point for the UE. The communications work on radio transmissions and digital signaling processing [15].

The 5G Next Generation Radio Access Network (NG-RAN) represents a big disruption, as it brings cloud technologies to the access network. In 4G, the RAN is the e-UTRAN, an evolution of the 3th generation (3G) e-UTRAN, but most commonly called LTE. It is a "flat network" as all its nodes, or Evolved Node B (eNB), are the same. This homogeneity, with the interface standardization, helps the interoperability between different vendors.

The main interfaces are the X2 and the S1, where the eNBs interact between them and the EPC, respectively, see Figure 1.1.

The eNB capabilities include the transmission and reception of radio signals. Its specifications vary with the generation, and they build the physical layer. This layer is later discussed in Section 1.2.1. The rest of MRN native protocols are layered on top of this layer to build the LTE protocol stack:

Layer	Protocol	Responsibilities
6	Non Access Stratum (NAS)	To manage connection between the UE and the CN
5	Radio Resource Control (RRC)	Handover decisions based on measurements; Pages UE over the air; Broadcast; UE measurement reporting; Sets cell-level temporary UE identifiers; Maintenance and set up of radio bearers [29]
4	Packet Data Convergence Protocol (PDCP)	Compression/decompressing headers of IP packets (UP); ciphering; routing of radio bearers [15]
3	Radio Link Control Layer (RLC)	To format traffic between UE and eNB; elimination of duplicates; segmentation depending on radio conditions; Qos reliability with Hybrid Automatic Repeat Request (HARQ) [29]
2	Medium Access Control (MAC)	Scheduling physical data flows based on priorities; Mapping transport to logical channels
1	Physical (PHY)	Section 1.2.1

Table 1.1: Stack of LTE protocols used by eNB

NG-RAN protocols stack is built on top of LTE. There is also a new layer to deal with network slicing and the NAS layer is divided in two.

Layer	Protocol	Responsibilities
7	Service Data Adaptation Protocol (SDAP)	To map the Qos flows of CN to radio bearers; Protocol added to work with 5G network slicing features
6	NAS-SM and NAS-MM	The NAS layer in the SP is divided into Session Management (SM) and Mobility Management (MM)

Table 1.2: The 5G air interface has the capability of playing with Qos priorities thanks to the new layer added to the protocol stack

Finally, there is the separation between UP and CP. Each plane has its stack, as shown in Figure 1.3:

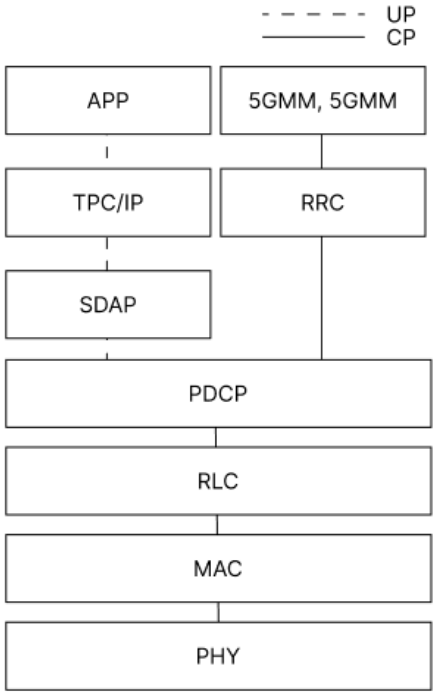


Figure 1.3: NG-RAN stack is similar to LTE with improvements of the network slicing and programmability

Carrier Aggregation and Dual Connectivity

An enhancement of LTE worth mentioning is the Carrier Aggregation (CA), introduced in Release 10. With it, a device can communicate with multiple radio cells of the same eNB.

There is a Primary Cell (PCell) that will control the signaling (CP), while the Secondary Cells (SCells) carries only traffic (Uplink (UL)). The same cell can act as PCell or a SCell for different devices [15]. This approach increases the data throughput.

In the 3GPP 12 Release, the CA was extended for multiple BSs. The PCell is in the master node (eNB or gNB), and the SCells are in the secondary nodes (eNB or gNB). The secondary BSs won't have any signaling with the EPC, only a UP connection [15]. This configuration, where more than one node serves the same device, is called Dual Connectivity (DC). As long the master node a 5G gNB, it is considered a SA deployment, as the right part of the Figure 1.4.

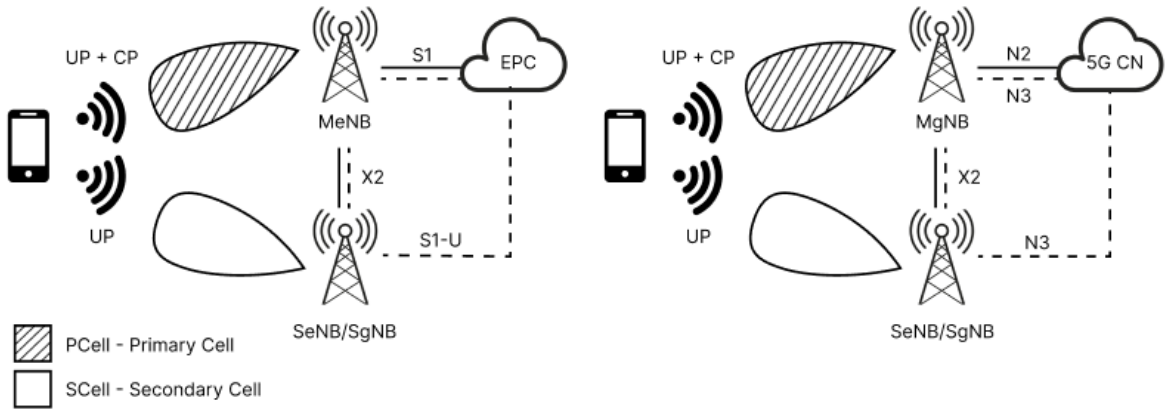


Figure 1.4: Dual Connectivity

This study explores a new face of the gNB associations, as instead of through X2, the gNBs interact through the E2 interface. This is a new interface introduced by 5G. It will be described in Section 1.3.1.

eNB to gNB

In NG-RAN, the modern BS is the gNB. It is distributed in submodules that implement network programmability, virtualization, and cloud technologies. Moreover, these new components connect to new RAN controllers and orchestrators to implement management policies, automation, and slicing.

The gNB is compliant with the EPC, LTE, and 5G CN interfaces. Therefore, it can communicate with both the CN of 4G and the LTE nodes. The backward compatibility is key given that most of the deployments are Non-Standalone (NSA); see section 1.1.3.

In a traditional LTE setting, the BS is all set up at the local cell site. The NG-RAN divides the gNB into distributed components:

- Radio Unit (RU): gNB radio cells circuits that communicate with the UE. Normally implemented in Field-Programmable Gate Arrays (FPGAs). They perform precoding, Fast Fourier Transform (FFT), cyclic prefix insertion/removal, and RF operations. They might include an important component worth mentioning: the Software Defined Radios (SDR). These are programmable radios that provide the flexibility to reconfigure the RU's functionality through software, enabling support for various frequencies, modulation schemes, and protocols without changing the hardware. This adaptability is particularly beneficial in 5G, where different spectrum bands and network configurations may be needed.
- gNB-Distributed Unit (DU): responsible for the higher functions of the PHY layer, scheduling, the MAC, and the RLC layers. A DU can be matched to one or multiples RUs.
- gNB-Central Unit (CU): handles up to one or several DUs. It divides the flows into CP and UP and exchanges messages with the UE, the CN, and other BSs. It implements the highest-level functions, such as decryption and encryption.

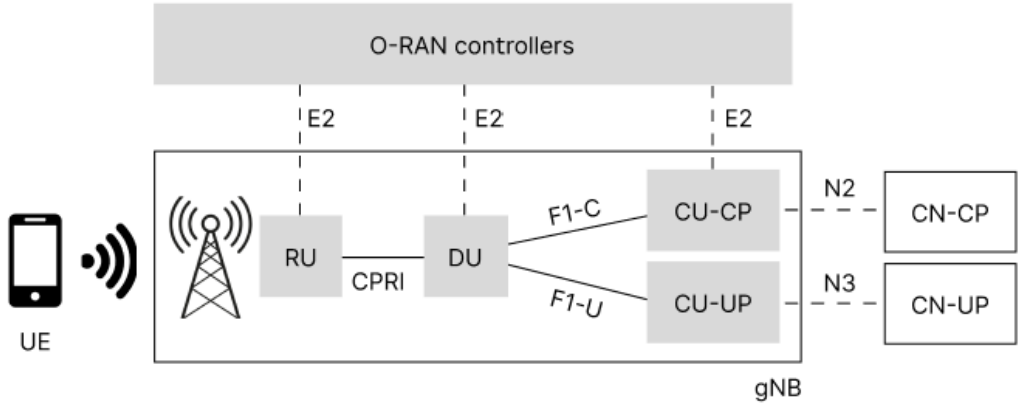


Figure 1.5: The new gNB in NG-RAN

The 3GPP introduced new interfaces with different requirements between the RU, DU, and CU, see Figure 1.6. This split offers possibilities of centralization, capacity aggregation, and versatility when assigning computing resources [45].

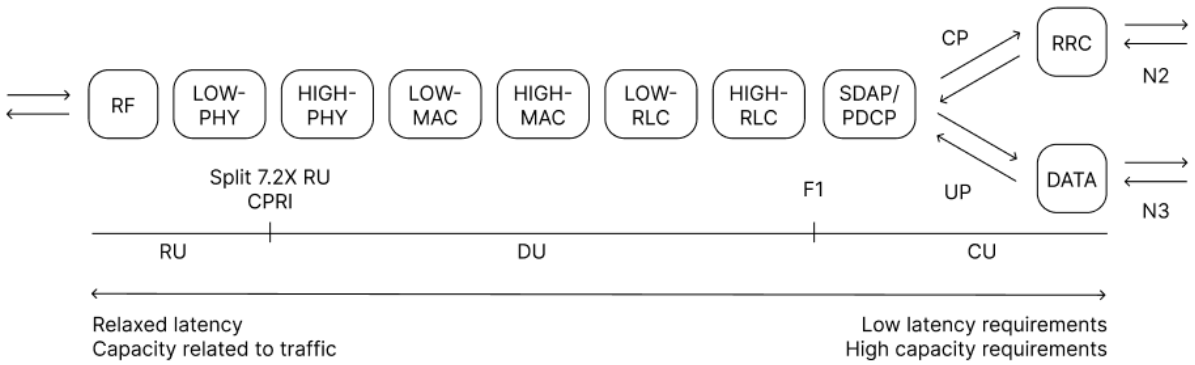


Figure 1.6: The eNB division into submodules to form the new elements of the gNB. Based on [45].

Before NG-RAN, outside the BSs, there were mainly passive elements such as access hubs, routers, and security gateways [15]. Now the network intelligence is distributed. The fronthaul, mid-haul, and backhaul, in Figure 1.7, can be adjusted to support different network requirements. For example, a Mobile Network Operator (MNO) can structure the topology by centralizing the DUs and keep the RU at the cell site to reduce the UP latency and support low-latency applications [15].

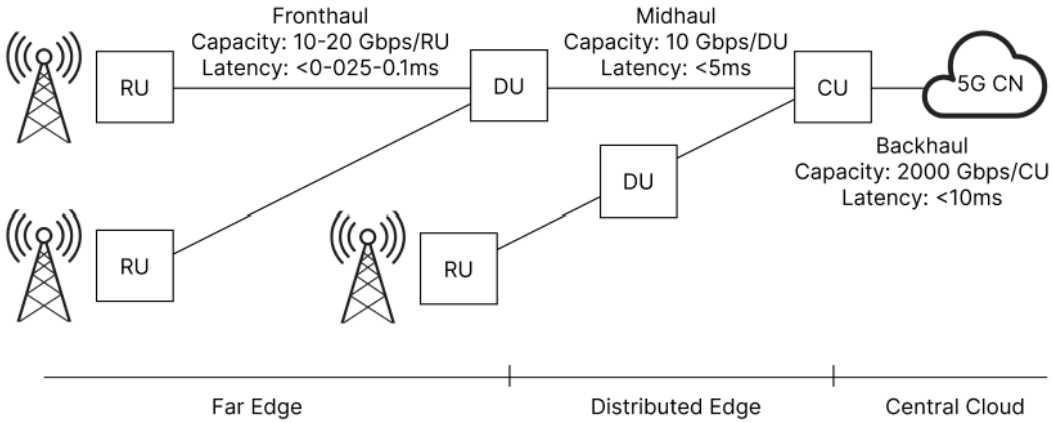


Figure 1.7: The NG-RAN topology can be changed by MNOs to meet different capacity and latency requirements. Based on [14]

Hence, the NG-RAN architecture is now disaggregated and is more flexible and programmable. This opens the doors for different MNO's deployment options. Finally, the intelligence of the network is not just only at its core or at the cell site; now, it is distributed, and it can be managed by controllers (see section 1.3).

1.1.3. SA vs. NSA

For a complete SA deployment, the master BS must be a gNB and the CN must be the 5G CN. In December 2022, already almost 86% of the announced devices by vendors were compatible with SA. Yet, in the same year, only 25% of network operators worldwide invested in public SA deployments [19]. The conclusion is that the network infrastructure is the one slowing the 5G implementation.

The new infrastructure requires very large investments from MNOs due to a higher BS density (smaller coverage per gNB), investments in spectrum licensing, transport networks, and core elements. This not only bounds the total number of SA deployments but also changes even how the business models work. There is an increasing tendency for MNOs to share the same infrastructure to reduce costs [27].

In the meantime, several NSA topologies are being studied. Where the CN is still the EPC and the NG-RAN holds both eNB and gNB. They interact through the X2 interface, even though they are from different generations. Finally, DC applies to the NG-RAN too, as shown in Figure 1.4. As long as the master node is from 4G, it is still considered NSA. It is considered a SA deployment when the master node is a gNB.

1.2. 5G signaling

Compared to free space, the urban environment is extremely complex when working with wireless communications. Common materials, such as brick and concrete, have high penetration losses, and radio waves suffer from reflection, diffraction, and scattering [15].

Furthermore, in MRNs, UEs are in movement, connecting and detaching constantly. In this scenario, the signaling messages in the CP are responsible for evaluating and enhancing the channel, the UP streams, and scheduling. These messages work out the radio cell associations, beamforming, UE reconnection, handover, the transmitted power, and much more to manage in such a challenging context.

The PHY takes care of the reference signals and the lower details of the RF settings. The proposed localization system is based on the SRS, which is one of the PHY reference signals.

1.2.1. 5G Physical layer

The PHY is responsible for the signal's digital processing, modulation, the antenna's transmissions, and the RF configurations.

4G Ultra High Frequencies (UHF) have become more and more congested, and UEs request higher data rates, which demand more spectrum. To deal with these, the 5G PHY introduces two new frequency bands:

- the FR1 (from 410 to 7125MHz): can work within the LTE range
- the FR2 (from 24.25 to 52.6GHz): also called mmWave (millimeter-wave). This band offers larger bandwidths and significant potential for localization services. However, it also offers smaller coverages in the air interface and suffers from significant losses [15].

In 5G, the modulation of these signals adjusts both the signal's amplitude and/or the initial phase ϕ. The modulated sequences have different waveforms from the original carrier and integrate a sum of frequencies known as bandwidth. All of this is processed by the RU.

On top, the MAC layer deals with the scheduling and multiplexing based on the structure provided by the PHY. Multiplexing is paramount, given the BSs must be able to handle several simultaneous streams as numerous devices are connected to the same radio cell.

There are several technologies to implement user multiplexing. In 2G, Global System for Mobile Communications (GSM) used Frequency Division Multiple Access (FDMA) and Time Division Multiple Access (TDMA), where each device had assigned distinct frequencies and time slots. 3G implemented the Code Division Multiple Access (CDMA), where the connections are labeled with different codes. Finally, both LTE and NG-RAN use Orthogonal Frequency Division Multiple Access (OFDMA) [15].

Overall, multiplexing is key as it allows MNOs to provide services to multiple users with the same infrastructure.

Resource Grid

The PHY 5G layer uses the OFDMA approach, where devices communicate through targeted time windows and frequencies. OFDMA is then divided into two domains: time and frequency.

The time units are defined by frames and slots, while each frequency is called a subcarrier. The devices are allowed to transmit RFs on assigned slots and subcarriers. This allows multiple devices to have dedicated communication channels between them simultaneously.

OFDMA is represented by a 2D plane where each axis represents slots and subcarriers. This is called the Resource Grid (RG). Figure 1.8 shows its layout and how each axis is

divided.

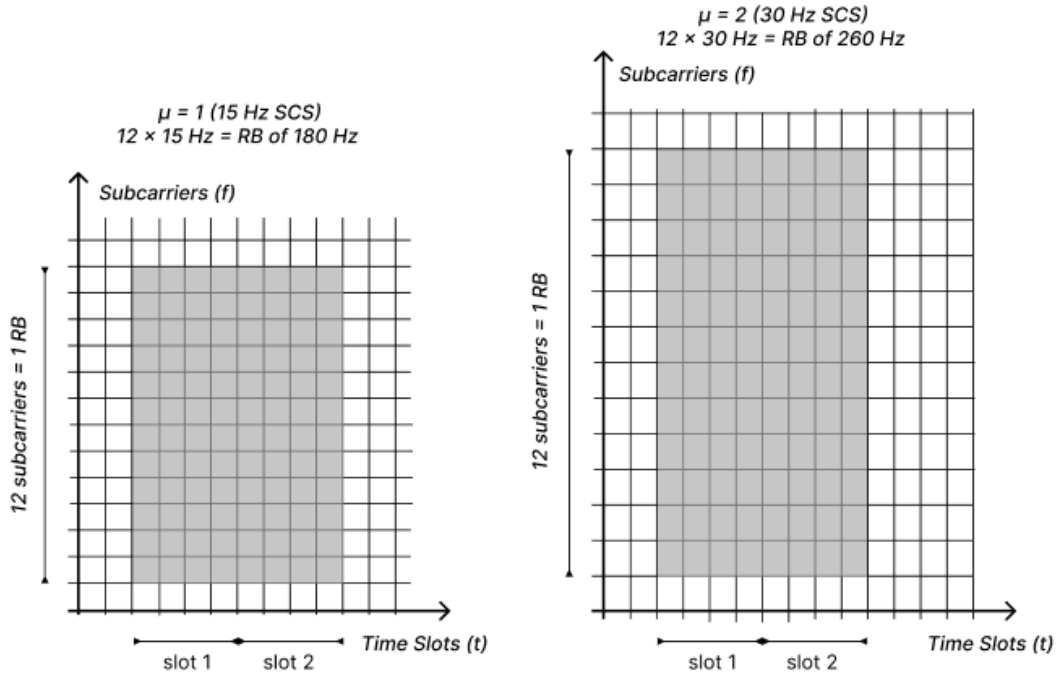


Figure 1.8: 5G Resource Grid for different 5G numerologies

The devices can transmit and receive data in several modes:

- Frequency Division Duplex (FDD): the uplink and downlink are transmitted on different subcarriers at the same time.
- Time Division Duplexing (TDD): the uplink and downlink alternate on the same carrier at different times.

Combining both, each resource on the grid has an associated mode, where a time slot and frequency are defined for uplink or downlink. Normally, the devices use a set of sequential subcarriers and time slots to transmit data. 12 sequential subcarriers in a time window define a Resource Block (RB). Finally, the RG coordinates of all the gNBs and UEs describing the RBs dedicated to each communication.

5G offers flexibility to configure the RG, where frequency settings allow to adapt the bandwidth for each subcarrier. These configurations, called numerologies, are described in the following subsection, which details the one used to capture the SRSs used for the localization system.

Numerologies and Frequency Domain

The signals captured during the experiments were allocated in the FR1 band. From 410 to 7125 MHz, this band offers 275 RBs (3300 subcarriers) [15]. These subcarriers are divided into RB.

Each subcarrier is defined by a central frequency and the Subcarrier Spacing (SCS), which can be configured to define the subcarrier bandwidth. 5G brings the flexibility to choose between different SCS configurations, also called numerologies (denote by letter μ) [15].

In addition, there are guard bands at the upper and lower subcarriers of the contiguous block in frequency. They help to prevent interference and their size varies depending on the PHY characteristics, such as urban or rural environment. What is important to understand is that they occupy part of the available RBs.

For a defined channel bandwidth, the vendor can choose how many data streams can be sent at a time, increasing or decreasing the numerology, hence, the individual band of each subcarrier. The amount of RB in the RG is then defined by the channel bandwidth and the numerology. In the FR1 band, the possible SCSs are 15,30 and 60Hz.

μ	**SCS (kHz)**	**Slots/Frame**	**RB bandwidth (kHz)**
0	15	$2^0 = 1$	180
1	30	$2^1 = 2$	360
2	60	$2^2 = 4$	720

Table 1.3: 5G FR1 numerologies

3GPP Releases state which variants of numerology and channel bandwidth are allowed together. In each cell, a device has up to two numerologies: one for uplink and the other for downlink.

For this study, the UE gNB channel is configured by $\mu = 2$ (SCS=30kHz) and a channel bandwidth of 40Mhz. Therefore, the total number of RBs is 106, as shown by Table 1.4, where the number of actually usable RB is 104 due to the use of 2 guard bands.

SCS	Maximum RB count												
	5	10	15	20	25	30	40	50	60	70	80	90	100
30 kHz	11	24	38	51	65	78	106	133	162	189	217	245	273

Table 1.4: Adapted from [15] - Maximum RB per channel bandwidth

Time Domain Structure

The timing unit in 5G is 1/64 of the LTE sample:

$$T_c = \frac{1}{4096 \cdot 2 \cdot SCS_{max}} = \frac{1}{4096 \cdot 2 \cdot 480Hkz} \approx 0.51ns \tag{1.1}$$

These units are grouped into frames. Each frame has a duration of 10 ms. At the same time, each frame is divided into 10 subframes, of 1ms each (1966080 T_c). Depending on the numerology, one subframe can have from 1 to 16 slots, where a slot contains 14 symbols [15].

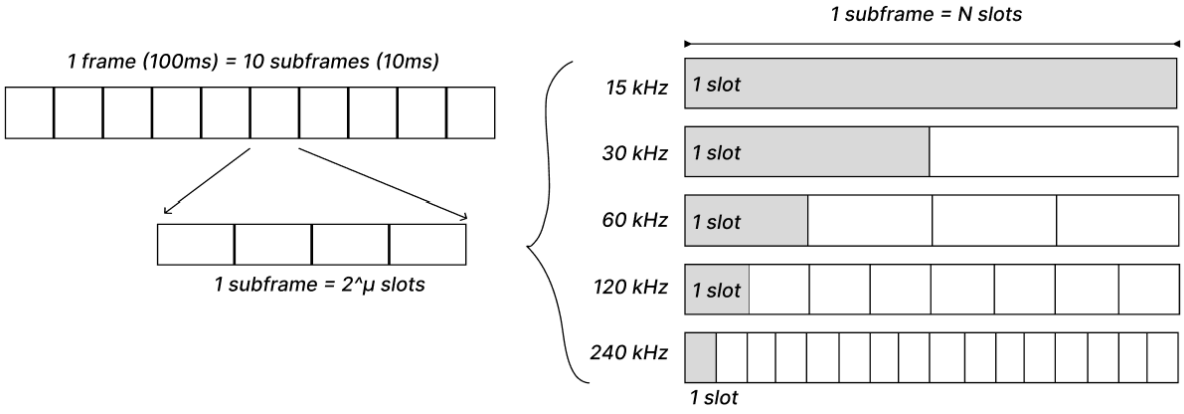


Figure 1.9: 5G TDD slot duration varies with the numerology

In resume, the 5G PHY can transmit uplink and downlink signals reserving RB of the RG for specific gNB-UE channels. These then, must negotiate the configuration to start transmitting based on their RF resources and RG grants.

1.2.2. 5G Reference Signals

One type of signal used in the gNB-UE negotiation is the reference signal. They act as fingerprints of channel quality, and hence they help to configure the communication. The

localization system leverages their properties to calculate the transmission delay between the UE and the gNB.

The reference signals are symbol sequences that, when transmitted over the air, provide data about the combined effect of channel multipath and power loss at the reception [9]. This makes them useful for measuring channel estimations, synchronization, and measurements in mobility and beam management. There are several of them in 5G, Table 1.5 lists them and their purpose. This study will focus on the SRS.

Signal	Name	Use	Direction
PSS	Primary Synchronization Signal	Acquisition	DL
SSS	Secondary Synchronization Signal	Acquisition	DL
CSI-RS	Channel State Information Reference Signal	Link adaptation	DL
SRS	Sounding Reference Signal	Link adaptation	UL
DM-RS	Demodulation Reference Signal	Demodulation	UL, DL
PT-RS	Phase-Tracking Reference Signal	Demodulation	UL, DL

Table 1.5: 5G Reference Signals and their use (source [15])

They work as follows:

1. Both endpoints, gNB and UE, have the same reference signal
2. One side, the UE for example, sends the signal over the communication channel
3. The other side, the gNB, receives the reference singal modulated by the channel. This means the received signal includes attenuation and time delay.
4. The receiver, the gNB, can detect the channel effects by comparing the received with the original one. This helps to calibrate the communication parameters to improve the RF transmissions based on the channel effects.

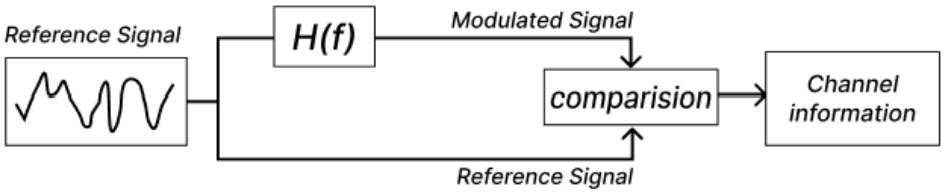


Figure 1.10: At the receiver, the reference signal modulated by the channel is compared with the original sequence to calculate the attenuation and delay parameters.

The SRS is one of the reference signals. It is UL which means the receiver is the gNB. The localization system exploits the channel detection properties to calculate the delay between the UE and the gNB. This is then translated into a physical distance to perform trilateration.

The reference signals are transmitted within specific BS of the RG: physical channels. It is crucial for the device and the BS to be synchronized to be able to recognize which channel occupies with resources, so they can communicate.

Channels

In the RAN, the transmissions are divided into categories or channels: logical, for transport, and physical.

Each physical channel has associated specific frequencies and time windows on RG and is dedicated to specific types of messages. They are broken down in terms of downlink or uplink and control or data type.

The reference signals are sent in control or shared channels in both uplink and downlink. More specifically, the SRS is sent in the Physical Uplink Control Channel (PUCCH). This channel is used to transmit Uplink Control Information (UCI) and has multiple formats [23]. If transmitted with data, it is sent via the Physical Uplink Shared Channel (PUSCH); otherwise, it is sent via the PUCCH.

Sounding Reference Signal

As explained, this study focuses on the SRS. As part of the reference signals, it helps the receiver to understand the channel effects. It leverages its code sequence properties to perform distance measurements from the UE to the gNB. It is exploited by the localization system considering two main facts:

1. With ranging techniques, it helps to estimate the transmission delay between the gNB and the UE
2. Only the gNB attached to the UE is able to understand when and how the UE sends the SRS signal

Overall, the UE sends this signal at regular intervals. Upon receiving it, the gNB is able to make intelligent decisions about the UP transmissions:

- the detection of the subcarriers with the strongest Interference-plus-Noise Ratio (SNIR). These would be the ones elected to communicate with the UE. The other

subcarriers will be left available to other devices.

- To define the uplink modulation, considering the ones the mobile can handle and the coding rate.

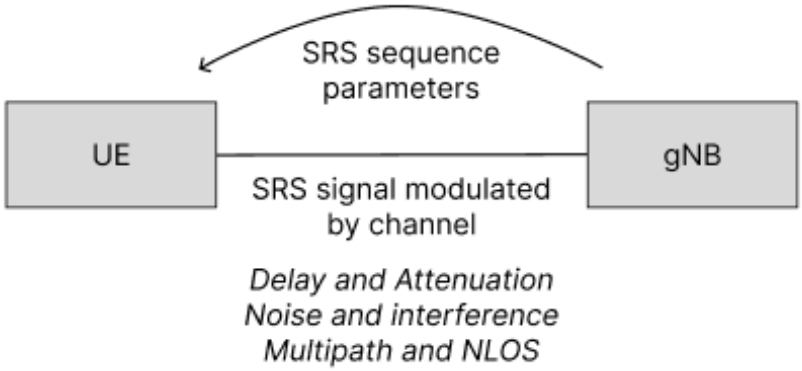


Figure 1.11: The receiver (gNB) can understand better the channel characteristics with the received (modulated) SRS.

During the attachment, the UE and the gNB negotiate the channel parameters via the RRC. This first attachment includes the SRS signals as part of the UL configuration. The gNB holds the parameters to generate a SRS sequence and then transmits them to the UE. By the end of this process, both entities have a copy of the same SRS and the UE sends it periodically to the gNB.

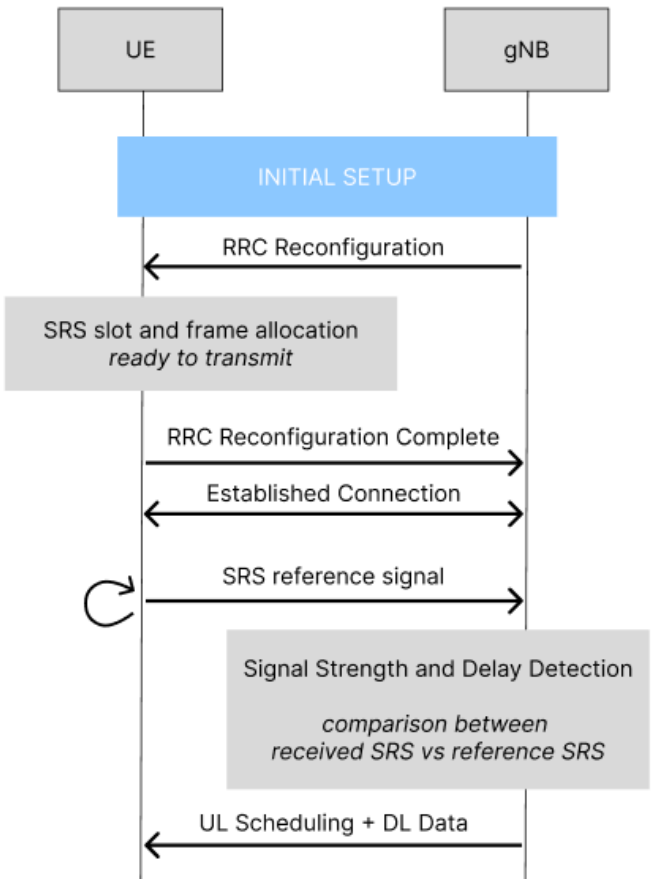


Figure 1.12: The UE sends periodic SRS to the gNB after the attachment procedure

The SRS code is computed with a Zadoff-Chu sequence [15]. The format depends on the cell and the device identifier, is complex-valued, and has an ideal periodic auto-correlation [21]. In other words, the periodic autocorrelation is zero for all non-zero time shifts. This property is useful for localization purposes as it will expose the time shift of the received signal. Joined with some localization functions, this shift can be translated into the distance from the gNB.

It has a specific allocation in the RG: each character, of the Zadoff-Chu code, is recorded in either 1, 2, or 4 consecutive Orthogonal Frequency Division Multiplexing (OFDM) symbols. It is sent over the PUSCH channel, spanning from the 8th to the 13th slot (time allocation). It is allocated in every 2 or 4 subcarriers (frequency allocation) [9]. Hence, a gNB can receive different SRS from several connected devices in different time slots and frequency frames.

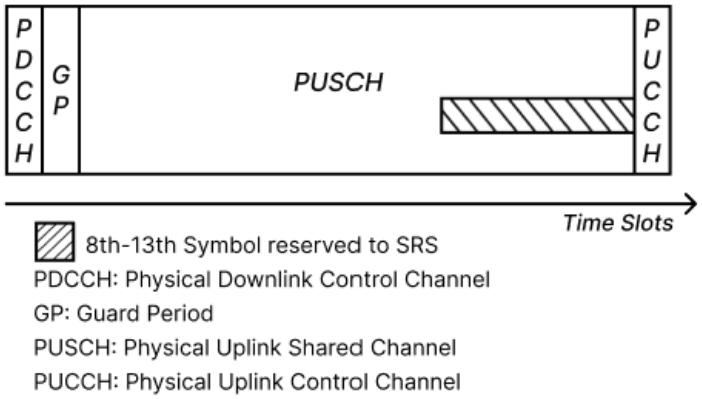


Figure 1.13: The SRS occupies the last symbols of the PUSCH channel.

Finally, its periodicity is configured by the RRC layer. In addition, the MAC scheduler is also able to adapt it dynamically based on the number of devices connected. If there are fewer, a lower periodicity can be used [9].

Timing Advance

As discussed, the SRS reveals the channel time delay between the UE and the gNB. This transmission time can vary based on the relative position of the UE to the station and may cause synchronization problems between them. The Timing Advance (TA) is the responsible mechanism to adjust them. It ensures that uplink transmissions from the UE reach the base station without interfering with previous or consecutive slots.

The TA dynamically adjusts the UE uplink transmissions based on the differential distance between the UE and the base station, which is critical as the UE moves. As a result, it prevents signal overlap by commanding UEs to transmit earlier or later, depending on their position concerning the one assumed during the previous uplink transmission.

Its value represents the delay correction for a UE message to reach the gNB. It is an integer in the range from 0 to 63 (8 bits), and the absolute transmission time in the uplink direction can be calculated as follows:

$$t_{UL}^{n} = t_{UL}^{n-1} + (val_{TA} - 31) \cdot 16[\mu s], \quad val_{TA} \in [0, 63] \tag{1.2}$$

where t_{UL}^{n} and t_{UL}^{n-1} are the current and previous time of the UE UL transmission, and val_{TA} is the TA value sent from the gNB to the UE.

Based on the last uplink transmission time t_{UL}^{n-1}, TA changes the time correction the user

might add or subtract to the next one. In this sense, the neutral value is 31, which won't apply any type of adjustment. As the UE moves further away from the gNB, the TA value will decrease to advance the transmission. When the UE approaches the gNB, the time it takes for the messages to reach it decreases, making the TA larger to force the UE a later transmission.

From the localization perspective, it is important to understand how the TA works, as it affects the time the UE might send its data. This will become more relevant after, in Section 3.2.4, where it is explained how the delay between the UE and gNB is used to perform trilateration.

1.3. O-RAN

The O-RAN Alliance, created in 2018, is a cooperative to standardize RAN interfaces, increase their programmability, improve interoperability, and overcome vendor lock-in. In 5G, with the new disaggregation of the base station in multiple components, CU, DU, and RU, it is key to establish a pattern of how they should interact. O-RAN takes care of their facades and also the one with the network controllers [42]. Their specifications are built on top of 3GPP releases.

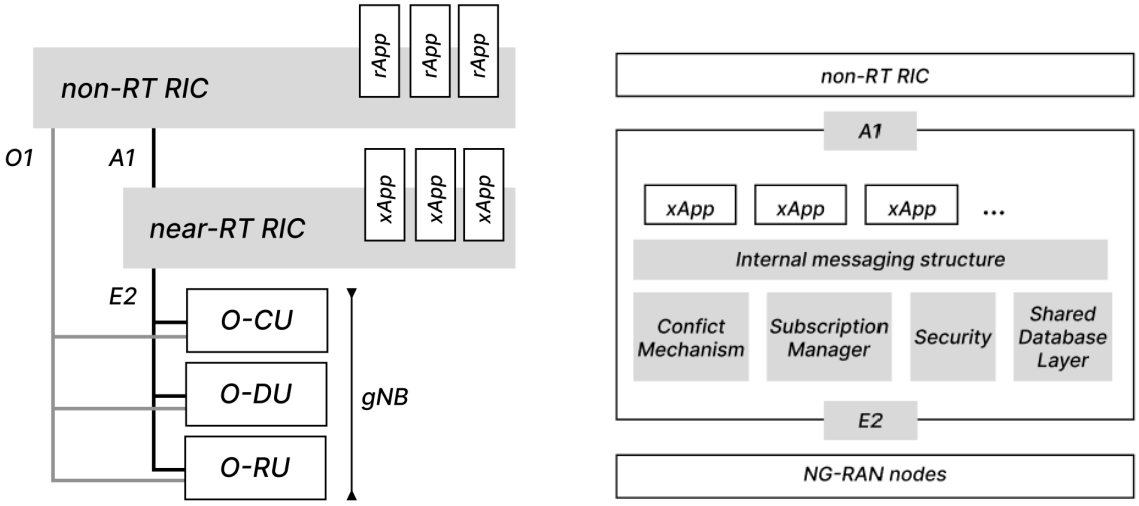
RICs are introduced as closed control loops that manage RAN nodes and collect data from them. RICs are capable of enhancing RAN features such as cutting, load balancing, scheduling, handovers, and policies thanks to their network automation potential [42].

O-RAN defines two types: the Near RealTime RICs (near-RT RICs) and the Non-RealTime RICs (non-RT RICs). While the first makes fast regional decisions and the second is slower and more global, see Table 1.6. They interact with each other and with the RAN nodes to collect data and perform actions.

Both RICs offer customization thanks to the xApps in the near-RT RIC and the rApps in the non-RT RIC . These applications are custom programs capable of interaction with the NG-RAN components to which the controller has access, see Figure 1.14. The interfaces between them are standardized and non-vendor-specific. O-RAN defines the protocols and their structure.

RIC type	Timescale	Coverage	Description
near-RT RIC	10ms to 1s	Between 100 and 1000 RAN nodes or UEs	Supports radio resource management, slicing, load balancing, handover, scheduling policies.
non-RT RIC	more than 1s	Thousands of devices	Connection with Service Management and Orchestration (SMO) and end-to-end Service Level Agreement (SLA).It manages ML models, applies more general policies, and has a more global vision of the network

Table 1.6: The two RICs offer different RAN control levels



(a) O-RAN RAN interfaces

(b) near-RT RIC

Figure 1.14: O-RAN and near-RT RIC simplified architectures. Adapted from [42]

1.3.1. E2 Interface

For this study, the most relevant interface is the E2 interface. It connects the near-RT with the gNBs. The implemented xApps are then able to read and control the RAN nodes.

The E2 functions are grouped in RIC Services (report, insert, control, policy, and query), which are based on functional procedures: RIC Subscription, RIC Indication, RIC Control, and others [38][1]. The E2 Application Protocol (E2AP) is the signaling protocol that builds them. It handles the lowest level actions: error reporting, setup, termination, and reset of the connection. Finally, the Service Models (SMs) are a combination of AP procedures joined to model common behaviors. For example, the E2SM KPM [39],

which reports performance metrics or the E2SM CCC [40], used to perform common gNB controls and reconfiguration.

E2 Service	E2AP message type	Description
Report	Indication of type report	The xApp specifies triggers and periodicity for the gNB to send requested telemetry
Insert	Indication of type insert	The xApp specifies only a trigger for the gNB to send requested telemetry. Used to request data without periodicity.
Control	Control Request	Used to modify the gNB behavior. It can be directly triggered by the near-RT RIC or as a consequence of an Insert.

Table 1.7: Some RIC services of the E2 interface

Embedding SMs into the E2AP packets, the xApps can request and subscribe to gNBs data, insert policies, and control their actions. In this manner, the xApps can read and manage NG-RAN nodes in a distributed and centralized way.

1.3.2. xApp

Emerging as a centralized solution to control RAN nodes, the xApps, in the near-RT RIC, offer a distributed vision of the network and can implement paramount control actions and policies.

For example, in [46], an ML-based xApp is developed to reallocate the resource grid dynamically based on dynamic traffic classes. They were able to estimate traffic demands in real-time with an accuracy of 85% and reallocate the resources as needed. In [11], Fredrik B. implemented a chain of xApps that controls the sleep time of the RU and reduces its energy consumption by 35%, increasing by 20% the number of sleeping RUs. These examples show their potential for network automation, programmability, and innovation.

Nevertheless, their development is challenging. They work in a complex microservice architecture, interacting with a wide range of protocols. In addition, MNOs support various hardware and software platforms, making standardization difficult [47].

In this context, the OSC is the main reference to develop consolidated xApps. Created in 2018 as an association between the O-RAN Alliance and the Linux Foundation, it is widespread in both the industry and academia. The open-source project creates templates

and O-RAN elements following its specifications. Finally, virtualization techniques, such as Docker, emerge as a portable solution to abstract their development from the environment details.

This study aims to push forward the xApp development with one functional implementation for UE real-time localization in a 5G SA deployment. Accordingly, the xApp was built based on the OSC standards, a Docker cluster running a near-RT RIC and Python Software development kit (SDK) from OSC [36].

2 | Localization Technologies and Trilateration Techniques

Localization is the process of finding the position of nodes. Several factors influence its quality, such as the available hardware, the medium access schemes, the accuracy of time synchronization, and the computation times [6] [16]. Hence, the position schemes need to balance their accuracy, energy consumption, and overall cost.

In this context, 5G offers a great infrastructure to leverage RF localization. The BS do not have energy limitations like in Wireless Sensor Networks (WSN); they already incorporate RF devices and directive antennas, their location is known, and the NG-RAN has high node density. Finally, the O-RAN programmability offers the possibility of combining different techniques and algorithms. All of these make 5G a great framework for performing localization.

2.1. Localization technologies

The positioning technologies normally are divided into *range-based* and *free-range*. The ranging, or the distance estimation between two nodes, is the base for many satellite and terrestrial systems, and it is based on RF hardware. All GPS, cellular-network-based, and indoor Wi-Fi rely on RF techniques to compute the distance between nodes [35]. Their energy consumption tends to be higher, and they offer a great position resolution. On the other hand, the range-free systems are expected to be more affordable, but as they are based only on node information and communication, they tend to be less precise [16].

With all said, the schemes to perform localization vary based on the physical and hardware resources, the positioning algorithm used, and whether they are based on range-free or range-based technologies. For this, it is hard to offer a rigid classification of all the possible combinations. Moreover, it is common to combine different stacks and techniques to improve the accuracy and balance their disadvantages. The section 2.1.1 lists the most common techniques and technologies used in localization schemes. In the next subsec-

tion 2.1.2, the range-based ones are more detailed due to how well they accommodate the 5G structure.

2.1.1. Localization technology comparison

- *GNSS*: it is the most popular outdoor method. However, it can become considerably costly and is not suitable for indoors and densely populated areas [4]. It requires precise synchronization between the transmitter and the receiver, as an error of microseconds can sum up to 500m [16]. The satellites involved have to be prepared for each localization attempt, which implies high power consumption and processing time. Additionally, the hardware in the receiver is expensive [6], and it requires Line of Sight (LOS) [16]. For all of these, the GPS offers great accuracy but is not scalable to implement with a larger number of nodes.

- *Received Signal Strength Indication (RSSI)*: the receiver measures the power lost in the transmission. The propagation loss can be transformed into a distance measurement thanks to the equation (2.1):

$$P_r(d) = \frac{p_t G_t G_r \lambda 2}{(4\lambda)^2 d^2} \tag{2.1}$$

  where P_t is the transmitted power, G_t and G_r are the gain of the transmitter and received antenna and λ is the wavelength of the RF[6]. It is quite used in fingerprinting schemes, where a reference heat power map is obtained, and the target elements modify it as they leave traces on it.

- *DV-Hop*: Distance vector range estimates the hop count between nodes. It is a range-free approach, where the nodes broadcast their coordinates across the network. In the end, each anchor node can get the distances to other nodes in terms of hops and meters [16]. It is simple to implement, but it has less accuracy compared to ranged-based approaches, and it is negatively affected by obstacles between the anchors [4].

- *Pedestrian Dead Reckoning (PDR)*: this approach uses native cellphone sensors, such as accelerometers and gyroscopes, to estimate the position of moving users. The sensors help to estimate the position based on the previous ones [16]. The downside of this approach is that the smartphone sensors embed drift errors, which, when accumulated, may affect the performance [43].

- *Wi-Fi based*: the home routers' easy access to UEs and increasing computing ca-

pabilities make Wi-Fi an exciting technology for localization schemes [16]. Several studies have been performed combining the fingerprinting techniques and RSSI. In [5], they proposed an RSSI fingerprinting scheme that, combined with deep learning techniques, was able to obtain an accuracy, at best, of 1.21 m. A hybrid approach with PDS is presented in [52], where they were able to reduce the drifting errors of the PDR. The downside of Wi-Fi used with fingerprinting is the pre-computation time needed to obtain the floor plan, which is limited mostly to indoor scenarios.

2.1.2. Range-based techniques

The ranging methods use RFs to compute the distance between an unknown point and the anchors (reference stations with known coordinates). There are several techniques used, based on different hardware, ranging signals and environment conditions:

- *Angle of Arrival (AoA)*: the angle between two nodes is computed with directional antennas that measure the incidence angle of the received signals [16]. Thanks to the beamforming techniques, the obtained angle defines an axis concerning the anchor; see Figure 2.1 and equation (2.2). Considering s as the anchor position and u as the target one, their arrangement defines the angle α used to define the line $\overline{us}$. [34]. With several anchors, these straight lines intersect to define the target's position. This approach is based on trigonometry: the laws of sine and cosines [6]. However, it suffers from reflection, diffraction, and scattering.

$$\alpha = \angle(u - s) = arctan\frac{u_y - s_y}{u_x - s_x}, \tag{2.2}$$

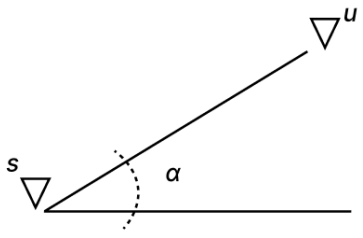


Figure 2.1: AoA ranging method.

- *ToA* (Time of Arrival): can be obtained by

$$d = \frac{c\tau}{2} = \frac{c(t_r - t_s)}{2}, \tag{2.3}$$

where c is the speed of light, and t_r and t_s are the reception time (measured) and the transmission time (inserted in the payload) respectively [35]. This procedure can also be referred to as *Lateration*, where the distance between two nodes feeds a *Trilateration* algorithm. This approach relies on synchronization between the anchors and the target object and is considered precise [16].

It uses the autocorrelation of signals to detect the delay between the received and the transmitted times (time of flight). Its major challenge is to mitigate the multiple delays received from the same signal due to multipath environments [4].

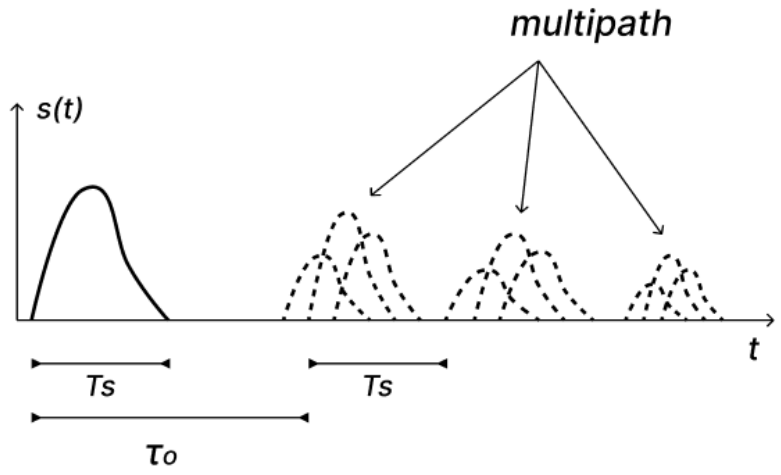


Figure 2.2: The signals are modulated by the channel in amplitude and pulse duration. Also, the multipath causes the same signal to be received multiple times.

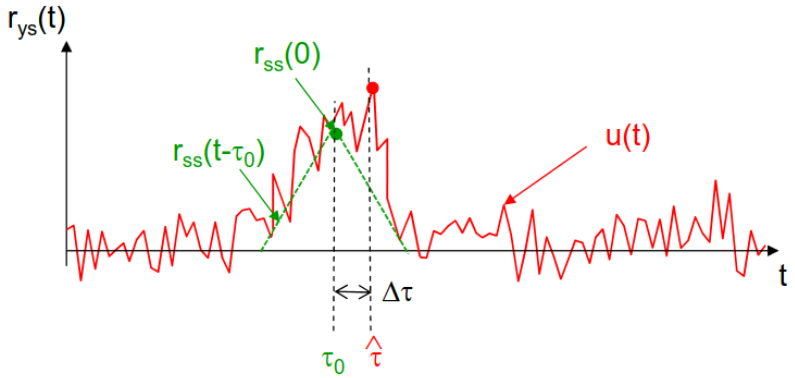


Figure 2.3: The Time of Arrival (ToA) estimate is the delay corresponding to the peak of $r_{ys}(\tau)$. This interval can be obtained with the correlation of the transmitted signal $r_{ss}(\tau)$ and the received one $r_{ys}(\tau)$. Source [34]

- *Time Difference of Arrival (TDoA)*: also called two-way ranging, computes the ToA difference between two anchors. This eliminates the user clock bias and, hence, the need for synchronization between anchors [34].

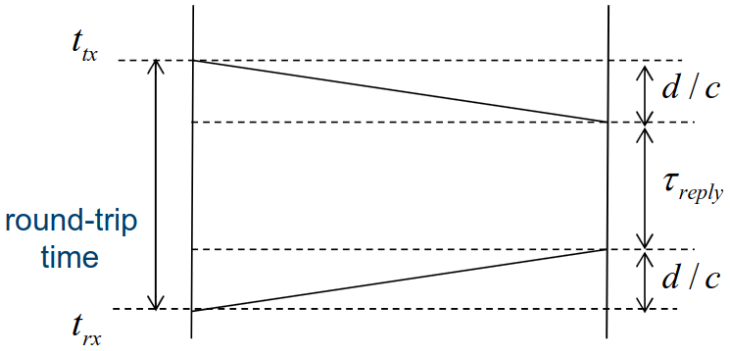


Figure 2.4: TDoA RRT avoids the need to have station synchronization. Source [35]

ToA and TDoA offer a high location accuracy, moreover, in wide-systems bands, which can be offered by 5G mmWave. In addition, both required synchronization between the anchor and the target, which in 5G is already offered by the synchronization between the UE and the gNB. Finally, AoA requires directive antennas and beamforming techniques already embedded into the gNBs. The negative side is that all of them suffer from (non-line-of-sight) Non-Line-of-Sight (NLOS) and multipath. However, 5G mmWave systems present the same challenges, which means their performance will evolve naturally with the development of 5G networks.

Technology	Cost	Accuracy	Downside
AoA	High	High	Complex hardware [4]
ToA	Medium	High, with precise time synchronization [4]	Depends on the synchronization accuracy
TDoA	Low	High	No need of time synchronization between anchors [4]

Table 2.1: Range-based localization techniques

This study will focus on range-based methods due to their feasibility in MRN. In this context, 3GPP has defined some pilot signals to be able to perform ranging [30]. For example, the reference signals in Table 1.5 can be used to perform ToA and TDoA ranging.

2.2. Trilateration

Trilateration is the technique used to localize a target based on N ToA, TDoA, or RSSI measurements from several stations. The accuracy of the algorithm is deeply connected to the signals' quality, which are translated into relative distances and then mapped to the coordinates of the target and the anchors as follows.

$$(x_1 - x_t)^2 + (y_1 - y_t)^2 = d_1^2 \tag{2.4a}$$
$$(x_2 - x_t)^2 + (y_2 - y_t)^2 = d_2^2 \tag{2.4b}$$
$$(x_3 - x_t)^2 + (y_3 - y_t)^2 = d_3^2 \tag{2.4c}$$

where (x_t, y_t) represents the calculated coordinates of the target at time t and (x_i, y_i) the ones from the i_{th} anchor. The distances obtained, d_1, d_2 and d_3, draw the circumference centered at the anchors coordinates. The intersection between all of them corresponds to the target position.

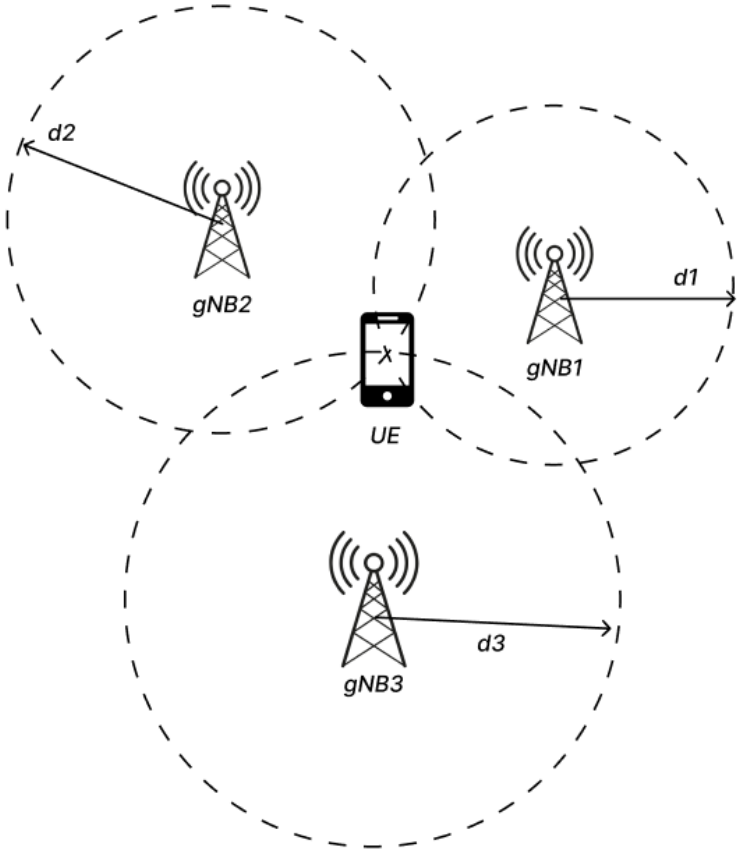


Figure 2.5: Trilateration example.

There must be at least 3 anchors to perform a $2D$ localization and 3 or more to perform a $3D$ one. For example, satellite systems required at least 4 satellites to perform a location attempt (they considered an extra time variable) [8]. With 3 anchors set up, the localization system is based on a trilateration algorithm responsible for translating the anchor measurements into the target position.

Nowadays, there are several options in the literature on trilateration algorithms within the mobile network context. For example, in [20] N. Heydarishahreza and N. Ansari provide a method to mitigate localization errors in a 5G network. They used a trilateration scheme joined with a Kalman filter, able to get an accuracy of 2 meters in a SA simulation. Also, new techniques have emerged to improve the already-known algorithms. In [30], Müller explores a skew-t distribution, rather than a normal one, to model ToA ranging errors of lower bandwidths in LTE. Then, the algorithms applied were shown to be more robust, improving the normal error quantiles by 43%. Overall, the algorithm must try to balance accuracy, time to converge, and computational complexity.

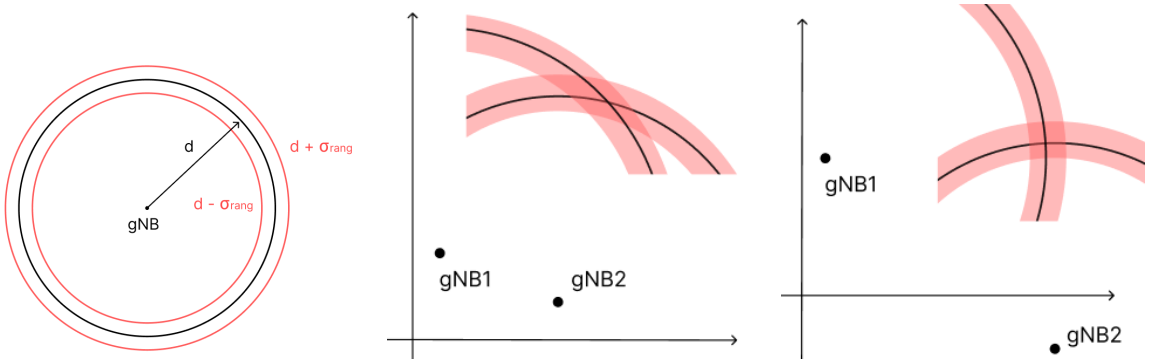
The proposed localization system is based on an implementation of ToA based on the SRS received by the gNB. ToA is a natural choice as it is considered precise. Its downside is the need for synchronization between the anchors and the target. However, this is already a requirement between the UE and the gNB. The process of ranging through SRS is detailed in further Section 3.1.1.

2.2.1. Error and Accuracy

Given that trilateration depends on anchor measurements, it is subject to their accuracy. In addition, the anchor's arrangement around the target also affects the performance. For the localization system implemented, the quality of the received SRS reflects on the time delay obtained by the xApp, which is translated into distance. The trilateration inaccuracy is then explained by two factors:

- Measurement accuracy (σ_{rang}): channel noise and interference affect the quality of the SRS received by the anchors. This error is then translated into time distortions in the delay, obtained from the SRS. Hence, it impacts the obtained ToA distance.
- Geometry factor (G): describes the arrangement of the gNBs, and defines the location error along a given axis. The best arrangement is a uniform distribution around the target [32].

Figure 2.6 shows how the ranging deviation σ_{rang} modifies the range circles in (a). In sequence, we present two cases with different gNB arrangements. The Geometric Dilution

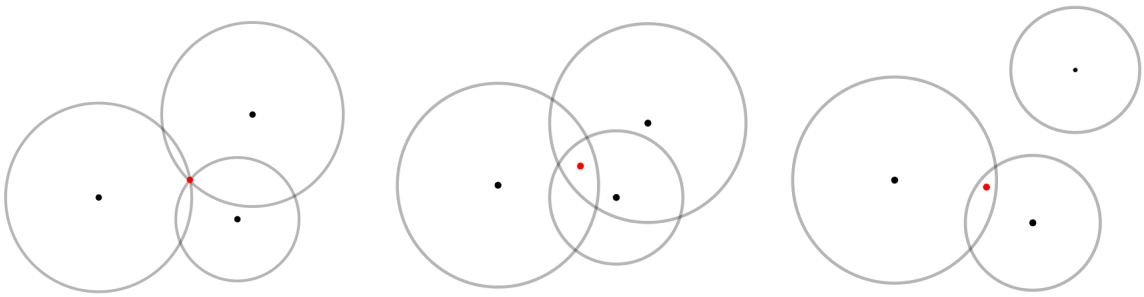


(a) Ranging error (b) GDOP 1 (c) GDOP 2

Figure 2.6: The measurement errors (a) and the geometric factor (b and c) affect the overall trilateration accuracy.

of Precision (GDOP) can be understood as the error impact on the trilateration accuracy due to the anchor's geometry around the UE. In the first case, the gNBs are closer, imposing a larger uncertainty in one of the dimensions. In the second case, the gNBs are more orthogonal, providing more equivalent errors between them. It is said that the precision in case (b) is diluted in comparison to (c).

In an ideal free error scenario, the distances calculated by 2.4 intersect at the exact target point. However, in reality, the trilateration suffers from the measurement variance of the anchors.



(a) Ideal (b) Overestimation of ToA. (c) Underestimation of ToA.

Figure 2.7: Ranging error impact in trilateration.

When the distance to the target is overestimated, case 2.7b, the uncertainty of the target position grows with the area drawn by the circle intersection. If big enough, the obtained target coordinates have a low confidence. In the second error scenario, the ranging underestimates the relative distances to the target. In this case, some trilateration algorithms might still work, but not in an optimal way.

Some articles explain how to deal with cases where there is no intersection. For example, in [25] Jiahong Li et al. offer a distance correction procedure to correct the lack of intersection of the ToA distances. Also, in [51], a range-based wireless localization based on a bounding-box algorithm, a modified version of the trilateration is applied to the case there is no intersection. Finally, in [12], an RSSI trilateration method is presented. By doing this weighed positioning, the location relies more on the measurements closer to the sender. They are also able to correct the cases where there is no point of intersection or when there are multiple ones.

Overall, the trilateration relies on the number of interactions between the circles of the ranging distances. In the perfect scenario, only one. When the quality of the measurements is not good enough, the trilateration accuracy is heavily impacted. Hence, it is natural to impose a minimum error ratio between the ranging distance and the error one. In [49], for example, S. Venkatraman et al. impose some constraints on the maximum measurement error accepted by their trilateration system. Considering the distance intersection between different anchors, see Figure 2.8, the measurement error couldn't be larger than the chords.

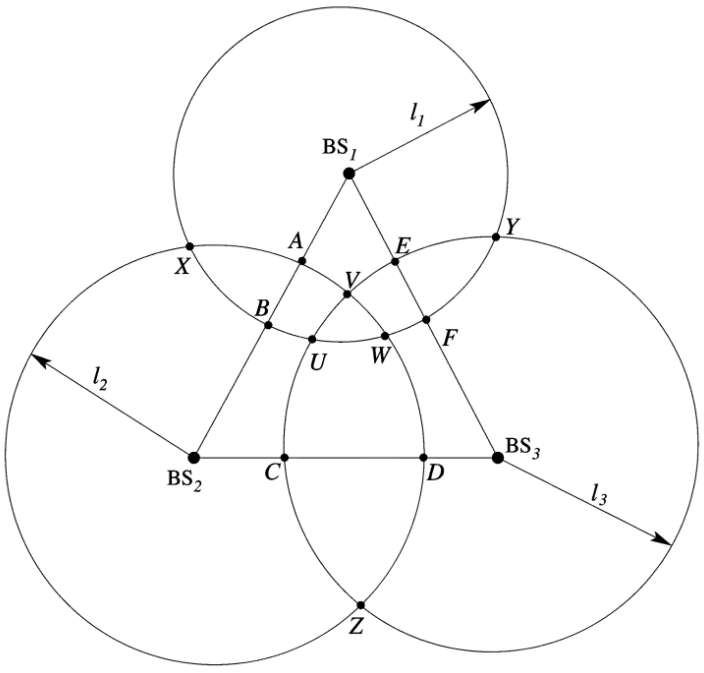


Figure 2.8: Trilateration distances. Source [49].

In this context, the NG-RAN reference signals provide a framework to implement either ToA ranging in a trilateration service. This study will focus on the SRS, already explained

in section 1.2.2. This uplink signal will be used to obtain the ToA distance between a UE and 3 gNBs. Finally, this data will be processed by an xApp, where the UE will be localized in real-time.

3 | O-RAN UE-Positioning Micro-Service

Considering all that has been discussed, this thesis aims to collaborate with the current 5G literature, providing a planned and scalable xApp with a trilateration technique based on SRS exchange. The Real Time 5G SA Localization System (Rt-5GLoc) proposed is a SRS-based trilateration platform built on top 3 gNBs emulators and an xApp running on a RAN controller.

Due to the difficulties in acquiring three physical SDRs and synchronizing them, the experiments were performed with virtual gNB emulators, with, however, real SRS signals captured by a single SDR. Thanks to emulation, Rt-5GLoc can perform trilateration experiments within various scenarios of geometries, abstracting the nuances of the physical layer and offering flexibility to build different conditions.

From the xApp perspective, the emulators act as real gNBs, and the system is prepared for a future substitution with real ones. Given that the trials were performed on real SRS data, they expect to anticipate the actual UE location obtained by real stations.

This chapter explains in detail how SRSs, joined with the TA correction, helps to situate UE and the structure built for this purpose.

3.1. Trilateration based on SRS

5G naturally embeds localization principles as it provides RF technologies, exploited by ranging techniques, and is set up by known positioned anchors, the gNBs. In this context, Rt-5GLoc leverages the properties of the SRS reference signals to implement a 5G ToA trilateration system.

SRS Database

Due to the physical and synchronization nuances of connecting 3 gNBs simultaneously, the data used in the Rt-5GLoc trials comes from real-world experiments with a single gNB. Its behavior was then scaled with gNB emulators (equipped with E2 terminations) fed by these signals.

In [7], V. Bernazzoli, E. Moro, and I. Filipini carried out a set of LOS experiments where a smartphone was positioned at different distances from a gNB. The testbed consisted of a single-cell 5G SA network assembled by: an open5gs CN, an OpenAirInterface (OAI) RAN, an Ettus Research SDR and an off-the-shelf smartphone with a programmable Subscriber Identity Module (SIM). In the trials, the UE sends periodic SRSs to the gNB, which embeds an E2 termination.

In resume, these trials provide a set of SRS signals and TA symbols sent and received at different distances from the UE to the gNB that Rt-5GLoc uses to feed emulators. Each emulator can be positioned at a different distance in the grid, based on the available SRS, and its angle can be changed to test different gNB-UE arrangements.

Again, it is important to highlight that, from the xApp perspective, the emulators act as real gNBs. This means that the xApp and its messaging are ready to be connected to real base stations.

3.1.1. Ranging through SRS

The SRS properties described in the previous Section 1.2.2 are paramount to understanding how these signals are used in the ranging estimation process.

The SRS is a sequence of symbols sent repeatedly by the UE to the BS to track the quality of the UL channel. The number of symbols it carries depends on the numerology and frequency configuration: it can be sent every 2 or 4 subcarriers. In other words, its definition can be doubled, increasing the band it occupies.

Rt-5GLoc SRSs were captured by a SDR with a channel bandwidth of 40MHz and an SCS of 30kH ($\mu = 2$). This leaves a total of 104 RBs available (see Section 1.2.1). The selected RF setting provides the maximum number of symbols per signal possible: a symbol every 2 subcarriers. In total, 6 SRS symbols per RB. This leaves a total of $6\frac{symbols}{RB} \times 104\frac{RB}{SRSsequence} = 624\frac{symbols}{SRSsequence}$ (in 1 time slot).

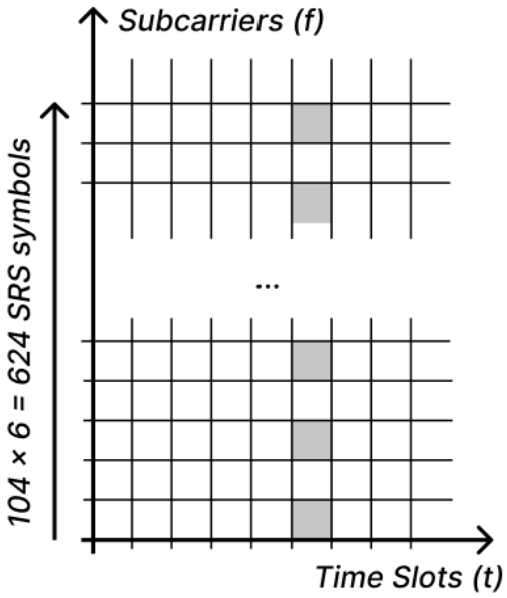


Figure 3.1: Each SRS is built of 624 symbols distributed every 2 subcarriers (40Mhz channel; 30kHz SCS; 2 guard RB)

When the UE sends the SRS, the gNB receives its symbols delayed and attenuated by the air channel.

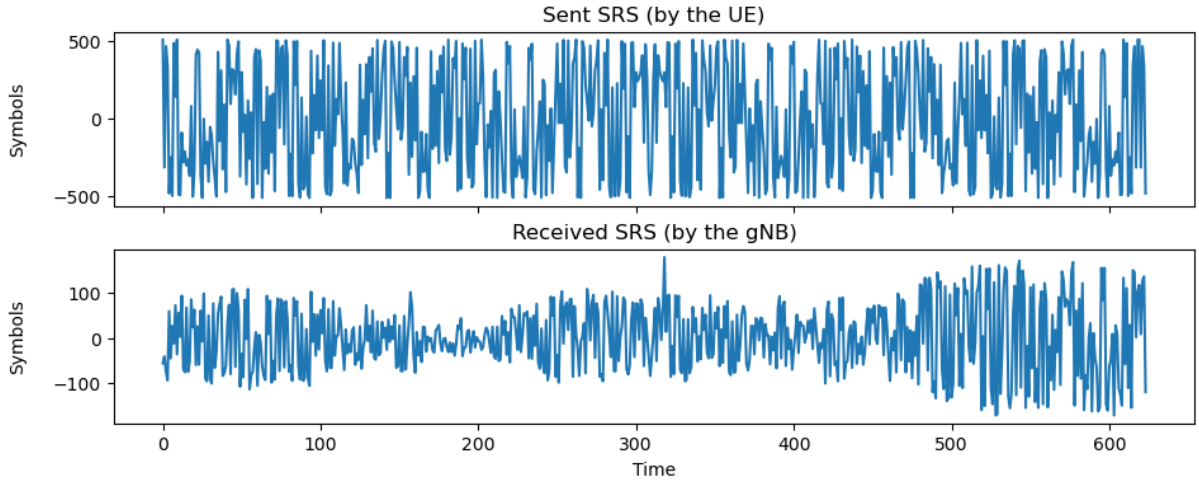


Figure 3.2: The channel effects of the SRS signal sent by the UE to the gNB

From now on, the *reference SRS* is defined as the original SRS sequence, and *received SRS* as the modified version. Given that the experiments were performed in LOS conditions, the received SRS collected represent a delayed and attenuated version of the reference one. With digital processing, the delay between them can be calculated with cross-correlation.

In other words, thanks to the SRSs sent by the UE, the xApp can obtain the time delay between each gNB and the UE. Finally, through ranging, this time variation is translated into the gNB-UE distance. The TA correction is applied, if needed, to each individual distance during the ranging function. The detailed version of this process is located in Section 3.2.5 below.

3.1.2. ToA and SRS

The ToA ranging can be generalized to estimate the distances in wireless communications between the transmitter and the receiver. In this scenario, the Time of Flight (ToF) of a message, to travel from the UE to the gNB, is calculated thanks to the SRS.

As explained, the ToA ranging computes their relative distance:

$$d = \frac{c \cdot \tau_0}{2} \tag{3.1}$$

where c is the speed of light and τ_0 the ToF, or the delay measured through the SRS.

In a LOS scenario between the UE and the gNB, the sampled reference SRS and the received one can be modeled by the following:

$$srs_{rec}(k\Delta t) = srs_{ref}(k\Delta t - \tau_0) + n(k\Delta t) \quad k = 1, 2, 3..., M \tag{3.2}$$

where Δt is the sampling window time, k is its index, $n \sim \mathcal{N}(0, \sigma^2_{rang})$ is a White Gaussian Noise (WGN) as the channel error, srs_{ref} is a Zadoff-Chu sequence (the reference SRS) and srs_{rec} the received one at the gNB . The signal srs_{rec} must be nonzero over its period, and the observation interval M should contain fully srs_{rec} and srs_{ref} [33].

In a more general scenario, the obstacles between the UE and the gNB impose nonlinearities in the model due to NLOS. For the SRS captured, the experiments were performed under LOS conditions; hence, equation 3.2 holds for this situation. The received SRS is then, fundamentally, a delayed and attenuated version of the reference one.

In sequence, the maximum likelihood ToA estimator $\hat{\tau}$, for this case, is the time index of the maximum peak in their cross-correlation [7] [33]:

$$\hat{\tau} = \Delta t \cdot argmax \sum_{k=-M}^{M} srs_{rec}(k) srs_{ref}(k + \tau_0) \tag{3.3}$$

$\hat{\tau}$ is the estimated ToA, Δt the sampling time, M is the length of the SRS sequence, and τ_0 is the propagation delay. During this process, if needed, the TA is applied, based on the equation 1.2, where the TA symbol corrects the time delay $\hat{\tau}$ calculated.

Joining 3.1 and 3.2, the distance d from the UE to the gNB can be computed by:

$$d = \frac{c \cdot \Delta t \cdot argmax \sum_{k=-M}^{M} srs_{rec}(k) srs_{ref}(k + \tau_0)}{2} \tag{3.4}$$

This process resumes how Rt-5GLoc obtains the gNB relative distance to the UE. In resume, the UE sends the reference SRS to the gNB that later acquires its delayed version and sends both to the xApp. Finally, the xApp calculates the distance based on equation 3.4, applies the TA correction, if needed, and scales this behavior to three gNBs emulators. The xApp is then the brain of the Rt-5GLoc structure, computing the localization as the signals arrive.

3.2. Infrastructure

Rt-5GLoc embeds three base stations for the UE position estimation. In our specific scenarios, real gNBs are substituted by gNB emulators, to ease the development and refinement of the localization accuracy of the system. The measurement of the UE position relies on the SRSs reception and cross-correlation with the reference SRS sequence, of which only the master gNB knows. In a non.O-RAN scenario, secondaries gNBs would not be able to retrieve and estimate the UE distance due to the lack of reference SRS knowledge. Hence, in this case, we are only able to calculate the gNB-UE distance from the *master gNB*.

Using O-RAN near-RT RIC, Rt-5GLoc programs other two *secondary gNBs* to listen to these SRSs. In other words, Rt-5GLoc connects to a *master* gNB, configures the other two *secondary*, and then requests and orchestrates the three parties to perform the UE localization estimation.

The localization system is then fully allocated on the NG-RAN, being independent of the CN.

3.2.1. Testbed architecture

Rt-5GLoc contains three main entities: the gNB, the RIC, and the xApp.

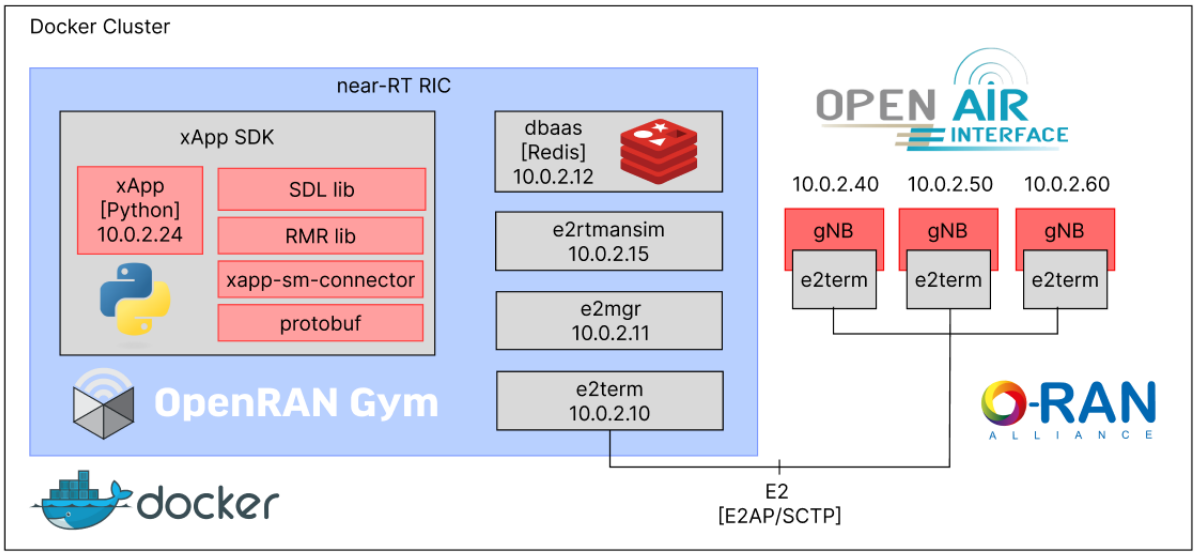


Figure 3.3: Rt-5GLoc structure: near-RT RIC represented by a set of components in the left blue box, including the xApp, and three gNBs, on the right, connected to it via the E2 interface.

The RIC is built on top of the OpenRAN Gym toolbox [10], which provides a set of template modules based on the OSC near-RT RIC [41]. Overall, the OSC and the Python xApp SDKs documentation is still at an early stage. However, it is one of the most popular RICs and offers flexibility with a modular design. The Python xApp, which runs on top of the near-RT RIC, holds all the ranging and trilateration functions. Finally, the gNB emulators are based on OAI E2 agents and terminations, fed by experimental SRS data.

Each component is a Docker container in a network with a proper IP; see Figure 3.3. The RIC internal components and communication are based on the OSC near-RT RIC, which have a timescale within 10ms to 1s, and the gNBs are Ubuntu 20.04 machines that run the OAI E2 components. The whole system is deployed by an automated script that prepares the localization experiments.

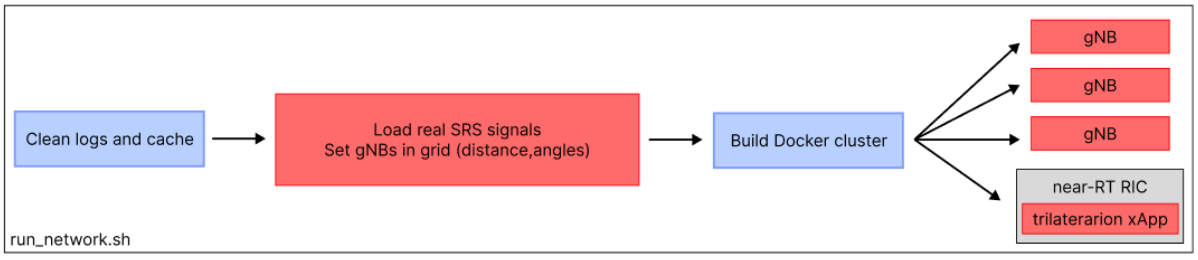


Figure 3.4: Flow to deploy the testbed

The gNBs connect with the RIC via the E2 interface; therefore, from now on, the gNBs are referred to as E2 nodes. The xApp, embedded in the RIC, is a Python application that controls and retrieves information from all three gNBs and where the localization logic is performed. Further details will be given in section 3.2.5.

3.2.2. near-RT RIC

The OSC near-RT RIC is a RAN controller compliant across several vendors and operators [48] and is a reference in both the industry and academia. At the same time, it is powerful but complex, as it runs on Kubernetes pods that can run multiple Docker containers. Assembled by microservices, its components communicate over an internal messaging structure with the xApps and the RAN terminations. They build the RIC actions, which, chained together, create O-RAN workflows.

As explained, the RIC implemented is based on the OpenRAN GYM toolbox, which offers a lighter version based on the OSC one. It runs a Docker cluster, where every container is either a RIC entity or a RAN termination. All of them run on the same network and have the IP and specific ports configured.

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
ff625229a76d	custom_gnb:latest	"tail -f /dev/null"	19 seconds ago	Up 17 seconds		gnb1
959734150b5f	custom_gnb:latest	"tail -f /dev/null"	19 seconds ago	Up 17 seconds		gnb2
12feca4a7ff1	custom_gnb:latest	"tail -f /dev/null"	19 seconds ago	Up 17 seconds		gnb3
efca16a8d7b2	custom_gnb:latest	"tail -f /dev/null"	19 seconds ago	Up 17 seconds		5g_automation_project-gnb-1
feb98d4c5eba	emoro/e2term:latest	"sh -c ./startup.sh"	19 seconds ago	Up 18 seconds	38000/tcp	e2term
c91869dbfc34	emoro/dbaas	"redis-server"	19 seconds ago	Up 18 seconds	0.0.0.0:6379->6379/tcp, :::6379->6379/tcp	db
9be1ef56eedd	custom_xappp:latest	"tail -f /dev/null"	19 seconds ago	Up 18 seconds	22/tcp	xapp
c0cba9d07518	emoro/e2mgr:latest	"sh -c './main -por…"	19 seconds ago	Up 18 seconds	3800/tcp	e2mgr
700314656be5	emoro/e2rtmansim:latest	"/bin/sh -c 'exec ./…"	19 seconds ago	Up 18 seconds		e2rtmansim

Figure 3.5: Localization testbed running on Docker containers.

The xApp, a Python application, uses libraries and plugins to abstract frequent and repetitive actions, such as serializing and sending information to the E2 Nodes (xapp-sm-connector and Protobuf), interacting with the database (Shared Data Layer (SDL) library) and the rest of the RIC entities (RIC Message Router (RMR) library), shown in Figure 3.6.

The rest of the RIC entities have specific functionalities.

- *dbass*: a Redis database. It holds all the E2 Nodes and the xApp information. It provides an SDL interface to the rest of the elements to access the UE and E2 Nodes data.
- *e2rtmansim*: the routing manager. It is responsible for distributing policies to the xApps and the rest of the components. It leverages the RMR protocol to route the messages from the E2 inside the RIC [10].

- *e2manager*: is the implementation of E2 manager, which controls the E2 connections and records the RAN information.
- *e2term*: provides connection between the RIC and the E2 nodes based on the E2AP protocol over Stream Control Transmission Protocol (SCTP). It communicates with the E2 termination of the RAN node.

The RIC Operations, RIC Subscription, Indication, and Control procedures are orchestrated by these containers. With them, the xApp can request information from the E2 Nodes, control which data and when they need to collect, and ask them to send it periodically.

3.2.3. Messaging

The content of the messages between the xApp and the E2 Nodes is based on Protobuf payloads [26]. This tool helps to build language and platform-agnostic payloads between systems. The message types are defined in *.proto* files with a key value notation similar to JavaScript Object Notation (JSON). Protobuf will scan them and prepare serialization functions in the target language, C and Python, in this case. Finally, the xApp, in Python, and the E2 Nodes, in C, will be able to read the same content.

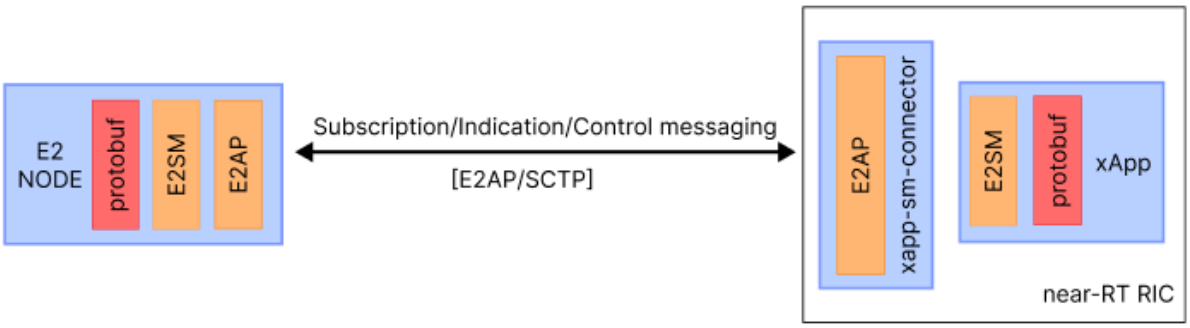


Figure 3.6: Information exchange between the E2 Nodes and the xApp.

The E2AP and the E2SM will abstract from the E2 Nodes and the xApp the connection setup, termination, and negotiation parameters. In the first interaction, the E2 Nodes announce their capabilities to the RIC, and the RIC will configure its services.

Section 1.3.1 explains how the Subscription, Indication, and Control services operate on top of the E2 interface, which offers a set of template messages to structure them. Based on these, the xApp leverages the Indication procedure to acquire SRS configuration data from the *master* gNB; the Control message, to prepare the *secondary* gNBs to receive

also the SRSs; and the Subscription message, to request these signals periodically to the three gNBs and perform trilateration.

The messages built for the system are detailed in the section ahead, at Table 3.7, while their trigger, response, and context are later explained with the xApp flow in Section 3.12.

3.2.4. E2 nodes (gNB emulators)

Some open-source projects have emerged to standardize the software of 5G gNBs. The OAI is an option for implementing a virtualized NG-RAN node, compliant with 3GPP and O-RAN specifications. It stands out for its open-source license, the OAI public license v1.1, and for being more computationally efficient than others [22] [50]. It provides software frameworks for the CU, DU, and RU that run on general-purpose platforms (x86) and SDRs.

For this setup, Rt-5GLoc E2 Nodes are built on top of the OAI project. As explained, these are emulators, but their connections with the RIC are the same as those in OAI gNB would use. Hence, they are considered equivalent to a physical gNB from the xApp perspective.

Each E2 Node is a Docker container running the OAI E2 agent and terminations in an Ubuntu machine; see Figure 3.3. The E2 termination is a C program, based on the OAI [37], that connects with the RIC through an E2AP/SCTP tunnel. The E2 Agent, also in C and based on OAI, reproduces the gNB behavior and communicates with the termination via the User Datagram Protocol (UDP) socket.

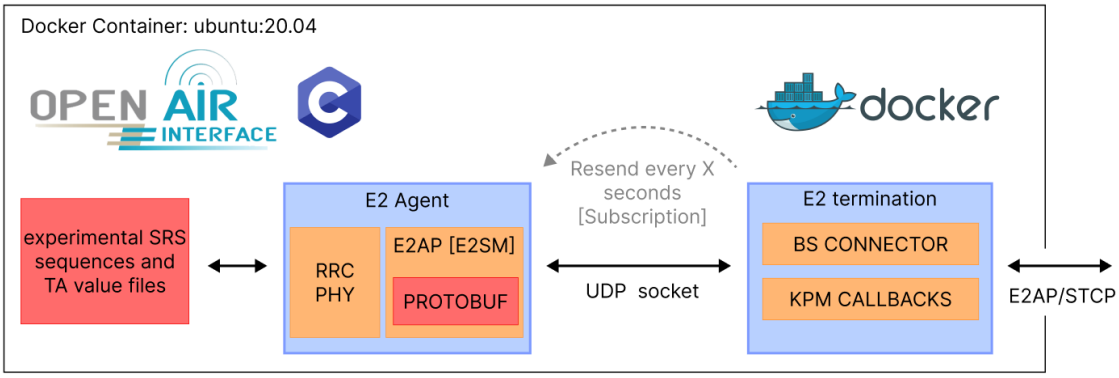


Figure 3.7: E2 Node implemented.

The *E2 termination* is the contact point with the rest of the network. It is based on the E2SM KPM, which embeds the Control, Subscription, and Insertion E2AP procedures.

These commands are then used by the E2 Nodes to communicate with the xApp. Its main functions are:

- To encode the E2SM payloads sent to the xApp and encapsulate them in an E2AP packet. The inverse process is done for incoming ones.
- To control the UDP and SCTP flows to communicate the E2 Agent with the xApp
- To define the timers to trigger periodic E2 Agent Subscriptions messages to be sent to xApp, or just a single one to send it once (Insertion Procedure).
- To set the identifier of the E2 Node for the RIC. Based on the environment variable *GN_ID*, that will translate into *gnb_ 734_ 733_ 0<GN_ID>000000* in the RIC database.

The *E2 Agent* is the gNB part that provides the network measurements. As the final endpoint of the xApp messages, it processes their payload and responds accordingly:

- Collects the SRS symbols and TA commands and sends them to the xApp via the E2 Termination
- Builds the payload of the Subscription, Indication, and Control messages
- Embeds the messages to the xApp into the Protobuf messages
- Sets the experiment time
- Has the SRSs slot and frame, and other configurations hard-coded, to send to the xApp

Both entities work to respond together to the xApp. While the E2 Agent is more domain-focused and performs the computations with the SRS and TA symbols, the E2 termination focuses more on the timers, the protocols, and the E2AP serialization.

In addition, different Action Types are defined for the xApp and the E2 Node connection. The Action Type is a header carried by the E2AP message next to the RAN Function ID.

Action Type	E2AP procedure	Periodicity	Description
1	RIC Subscription	Every 3 seconds	Used to send SRS symbols
2	RIC Indication	-	Used to carry specific, only once, information
3	RIC Subscription	Every 30 seconds	Used to send TA symbols

Table 3.1: E2AP procedures used

The interaction between the E2 Nodes and the xApp is based on these types of messages. They build the Subscription, Indication, and Control flows to send and receive SRS and TA symbols.

In this context, Rt-5GLoc performs ranging relying on real word SRS signals thanks to the RIC-gNBs communication. The E2 Nodes (gNBs) coordinates, and therefore their relative distance to the UE, can be changed and combined to perform different experiments. Finally, different SRS and TA experiment files can be uploaded to the system accordingly with the distances set. With this, before a localization experiment, the gNBs parameters must be set individually with environment variables:

- *SRS_LOAD_COUNT*: sets the number of SRS the E2 Agent will load. In other words, how much time will the localization experiment last, knowing the SRS periodicity (All E2 nodes should be set at the same number).
- *TA_LOAD_COUNT*: similarly to the SRS, it sets the number of TA the E2 Agent will load.
- *GNB_DISTANCE_TO_UE*: true distance from the UE to the gNB coordinates.
- *GNB_ID*: sets the E2 node identifier in the RIC.

Steps into a physical deployment

The OAI software alliance is developing a monolithic gNB project that offers all the CU and DU RAN functions: the RRC and PDCP CU layers, the PHY, MAC, and RLC in the DU ones, and finally, the RU. Once deployed and connected to a 5G CN and an SDR, they build an SA network.

The E2 Agent used is also from OAI. This means it is compliant with the OAI gNB. Actually, in this thesis, the E2 Agent is considered an emulated gNB. But, when connected

to OAI CU, it is a facade with the E2 termination. As it shows, the structure implemented by E. Moro et al. in [28].

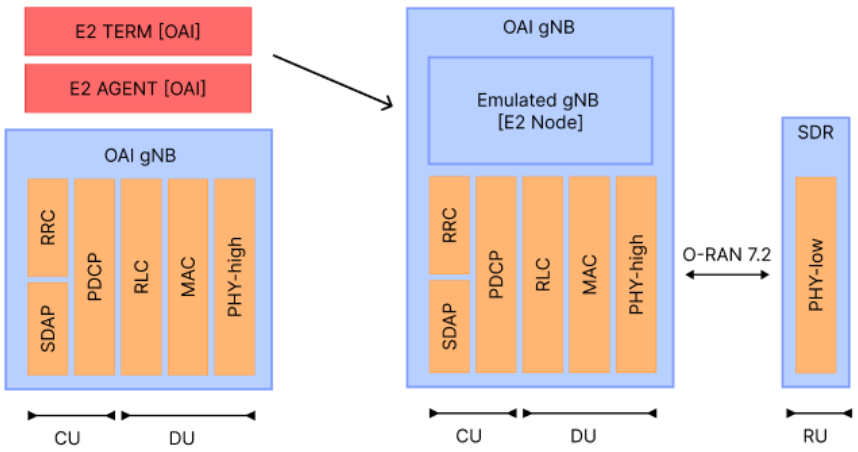


Figure 3.8: Transition to a physical gNB

The current system can be leveraged to build a final system with OAI gNBs. The Rt-5GLoc E2 Node emulator, described, should be modified to act as a facade between the OAI gNB and the near-RT RIC.

3.2.5. Trilateration xApp

The Rt-5GLoc xApp aims to shed some light on the xApp community, as setting up an xApp yet is an arduous task due to its context complexity. Some articles in the literature aim to clarify the virtualization structure of O-RAN and offer a context to start from the community [42] [31] [47] [48] [10]. Still, the O-RAN model core is quite elaborated as it is built on top of the software, virtualization, and the 3GPP specifications.

Based on a Python application, the Rt-5GLoc xApp transmits and receives messages with gNBs emulators to implement ranging and localization. It was designed with a focus on scalability, enhancing the existing trilateration approach, and supporting the addition of future implementations. Its software architecture is then divided into different Python modules layered with a specific division of responsibilities:

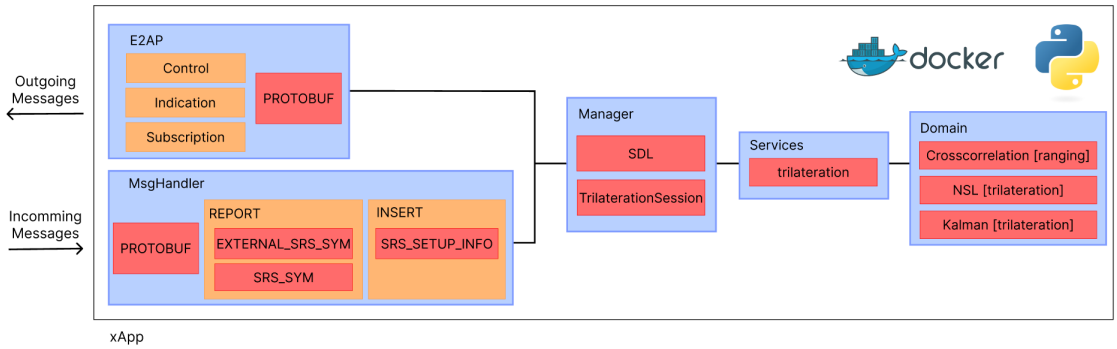


Figure 3.9: Rt-5GLoc trilateration xApp software architecture

Such structure, shown in figure 3.9, helps to visualize the tasks of each module and to map the dependencies between them. From right to left, the domain layer represents the localization algorithms and must not depend on any other. The services and managers hold the business logic that relates the domain with the network infrastructure (infra) details. Finally, the left layers, or infra, hold the NG-RAN protocol implementation details.

With this setup, to add a new localization algorithm, it would be necessary only to define a new function in the domain module and call it from the manager or service. In this sense, there is no need to change the E2AP logic as long as it is based on the message types already implemented (Table 3.1).

Each layer has a defined set of responsibilities:

- E2AP and MsgHandler: facade for the outgoing and incoming messages that abstract the E2AP logic. It is protocol-specific, and it serializes and deserializes the payloads with the Protobuf classes. The incoming messages are received by the Handlers, who deserialize the messages and trigger the correct manager or service. The outgoing messages go through the E2AP module, which will just embed the payload into a Control, Indication, or Subscription message and send it to the E2 Node.
- Managers: they act as either interfaces, such as the SDL manager (holds all the functions related to the RIC Database), or as state controllers. The state machines program the messaging between the xApp and the gNBs. The TrilaterationSession Manager (TSM) is the *central brain* of the reactive trilateration xApp. This class orchestrates the messages sent to the E2 Nodes. More about it in Section 3.2.5.
- Services: they handle the business logic specific to a localization function. For exam-

ple, they can implement whatever function is related to trilateration, triangulation, or multilateration. These functions are called by the state machine, and they help to divide the responsibilities. A specific state can call multiple services in sequence to trigger the next one.

- Domain: ranging and trilateration functions. These algorithms can then be tested outside the xApp and embedded easily. This layer must be kept independent to maintain this flexibility, it must not depend on the left ones.

In this scenario, this architecture is favorable to multiple implementations to build a general 5G localization framework. The following section explains how Rt-5GLoc leverages this structure to implement the ToA trilateration based on SRS and TA symbols.

xApp state machine

In this context, the xApp collects periodically SRS and TA values from the gNBs, requested by the TSM. The latter is the *brain* of the SRS trilateration: it holds the E2 Nodes identifiers, their coordinates, and controls when and from which station the symbols are requested.

The TSM then puts together all the parts seen so far (the E2 messages, the collected SRSs, and the trilateration algorithm) to build the localization system. Based on a state machine, it defines the triggers for the xApp-gNB requests and responses. The overall flow is described by Figure 3.10. In the first place, it configures the gNBs based on the trilateration session and then collects their signals to obtain the UE coordinates.

The flow is constructed by a set of states in a state machine that triggers different domain and infrastructure actions:

1. State 0: The xApp begins without any node registered. First, it connects to the master gNB. This is the only one attached to the UE, and therefore, the only one with the frame and slot information to listen to its SRS. By listening, it is intended that the RAN node is configured to recognize the reference signal, sent by the UE, in a specific time and frequency. Its slot and frame configuration is done at the RRC layer. The xApp then requests to the master, in an Indication message, the SRS RRC setup parameters; see Figure 3.11.

2. State 1: The master response is then received by the xApp. It holds the SRS frame, slot, and the reference SRS sequence in addition to other RRC values. The xApp then configures the other two gNBs to recognize the same signal, this is done by embedding the SRS parameters in a Control Message sent to the slave gNBs. At

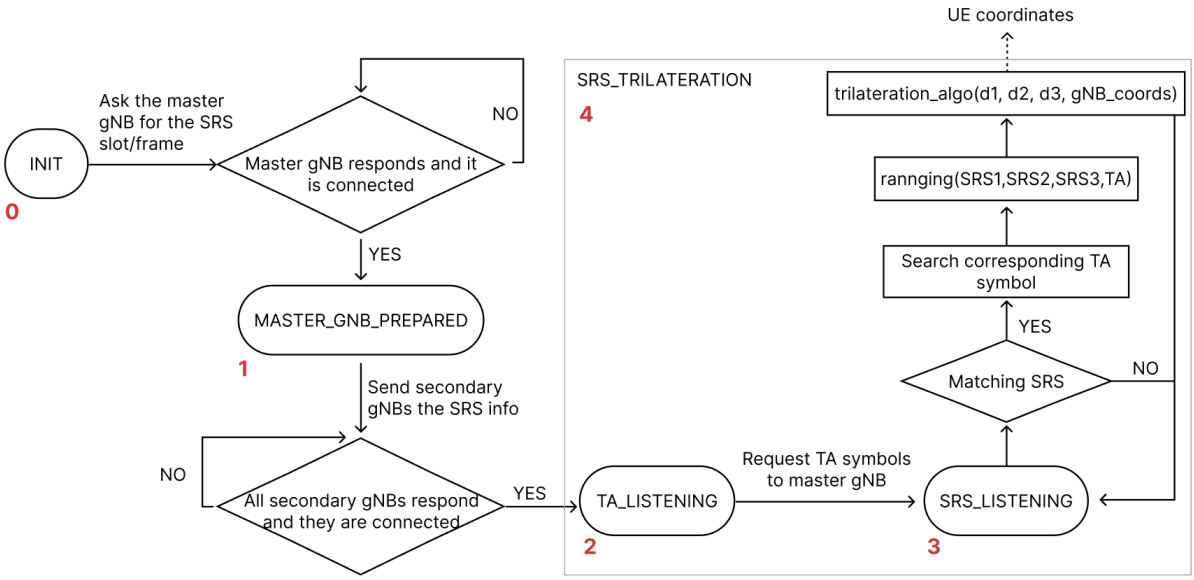


Figure 3.10: TSM state machine.

the reception, the secondary nodes will then configure their radio resources to listen to that specific time slot and subcarriers. Finally, all three gNBs are registered and capturing the SRSs sent by that specific UE. The reference sequence is then saved in the database to be used in the ranging algorithm.

3. State 2: with all the three gNBs ready, the xApp subscribes in the first place to the master TA commands, which will be sent to the UE. After receiving the Subscription message, the master will send the TA information periodically to the xApp.

4. State 3: the xApp emits Subscription messages to all the gNBs SRS received sequence. Each SRS signal later is used to estimate the specific time delay between the UE and the gNB that collects it. Then, cross-correlated with the reference SRS saved at the beginning, it will translate into a ranging distance.

5. State 4: This final step outputs the UE location. Based on a matching rule, the SRS from different gNBs are grouped to obtain 1 set of UE coordinates. In this case,the SRS are associated if they have the same timestamp (in milliseconds). When a triple of SRSs is detected, the xApp searches for the corresponding *master* TA symbol of that SRS timestamp. Then, the SRS ranging, defined at Section 3.1.1, is performed with the 3 signals. If needed, the TA adjustment is applied in the ranging function to each individual range. This results in 3 individual UE-gNB distances that are then used to execute the trilateration algorithm, which outputs the UE position. Every 3 matching SRS, this process is repeated.

```
message rrc_setup_srs_info {
    repeated srs_signal_m srs_reference=1;
    repeated int32 rnti = 2;
    repeated int64 timestamp = 3;
    repeated SRS_RCC_config_m SRS_params=4;
}
message SRS_RCC_config_m {
    required int32 srs_bandwidth=1;
    required int32 srs_hoppingbandwidth=2;
    required int32 freq_domain_postion=3;
    required int32 duration=4;
    required int32 srs_config_index=5;
    required int32 transmission_comb=6;
    required int32 cyclic_shift=7;
}
```

Figure 3.11: Protobuf SRS RRC parameters.

The interactions defined by the states between the nodes and the xApp are described by figure 3.12, which shows how the state machine in the TSM dictates the messages exchanged.

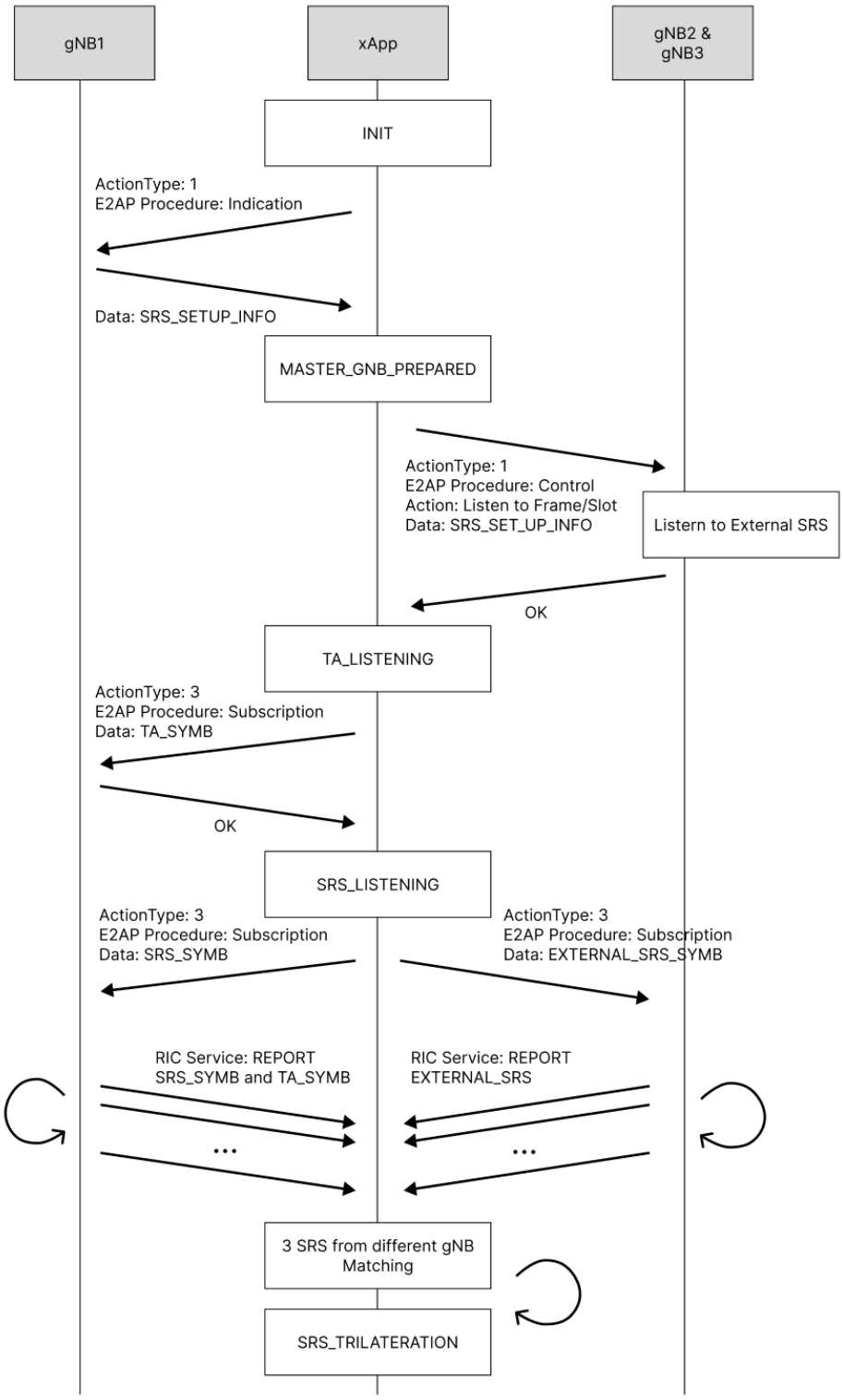


Figure 3.12: Messages exchanged by the xApp and the gNBs in a trilateration session.

3.2.6. Database

All the data is saved in the RIC database. The database is assembled by three entities: a *Session*, a *SRS resume* and the *SRS symbols*, see Figure 3.13.

The *Session* entity stores all the session information related to gNBs that is relevant to perform the trilateration algorithm; including gNB IDs and locations, SRS configuration and reference SRS, UE's last estimated location, which is dynamically adjourned. It also saves the final outputs through the process. The *SRS resume* is used by the trilateration services to track the SRS match within a certain time window. We call a triple, a set of three SRS delayed replicas of the UE's transmitted reference SRS, each of them received by a different gNBs. When a triple is found, the respective SRS symbols received from each gNB are retrieved to perform the algorithm. This information is saved in three tables: <gnb1>, <gnb2>, and <gnb3>, each one holding all the *SRS symbols* registered by the reference node since the beginning of the trilateration session. Hence, the only related entities are the *SRS resume* and the *SRS symbols*.

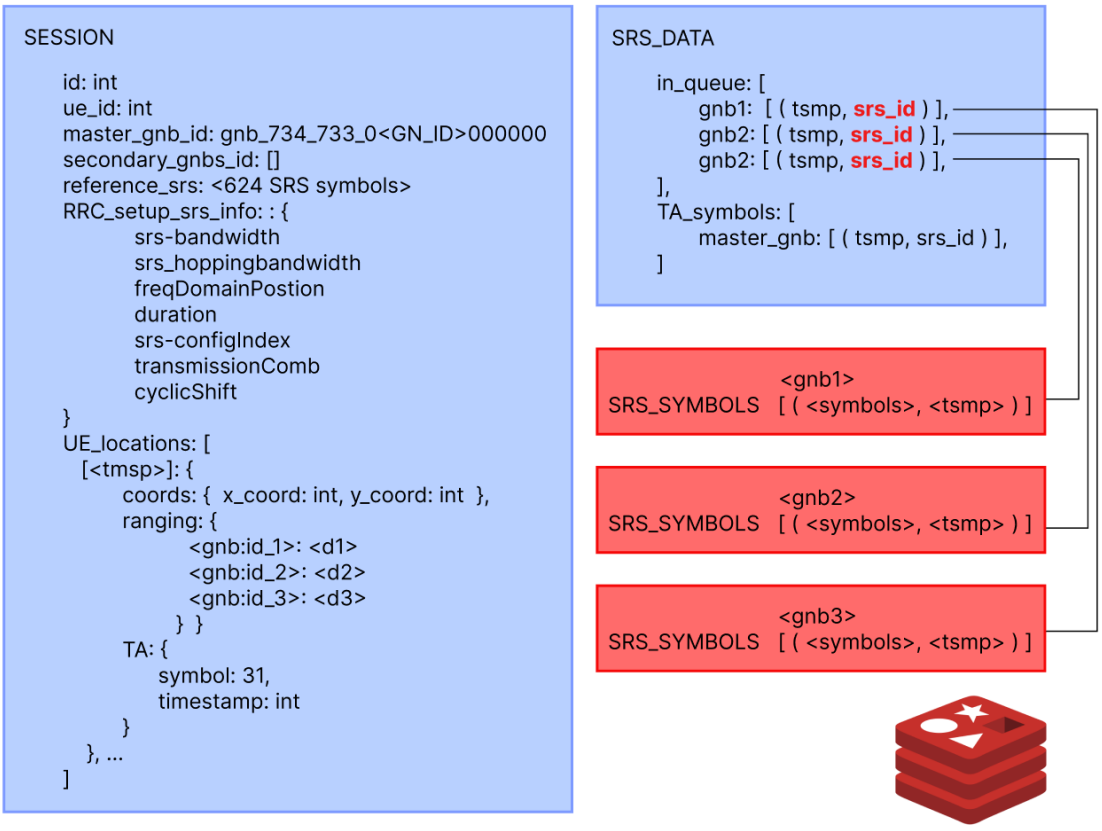


Figure 3.13: Database representation of a trilateration session.

3.3. Localization Algorithm

As explained above, the xApp periodically receives the SRS and TA symbols from the gNBs. Once it detects three SRSs from different gNBs, in a certain time window, it triggers the localization algorithm:

1. Detection of a *received SRS* triple, defined above, from different gNBs.
2. Perform cross-correlation of each sequence with the *reference SRSs* to obtain one ToF value per anchor (equation 3.3).
3. TA adjustment of the delay, if needed, with the matching master TA symbol.
4. Translation of the delay into a distance d_i (equation 3.1).
5. To obtain the UE coordinates, apply the ToA trilateration algorithm to the three distances acquired.

In this context, two different trilateration algorithms were implemented in the Rt-5GLoc framework. The first one, the NLS, is a non-recursive algorithm, while the second is a Bayesian tracking technique with a Kalman filter. The section ahead explains the definition of UE error assumed to compare their results.

3.3.1. Trilateration error

ToA trilateration is one of the most common methods used to perform localization based on three anchors [44]. In a 5G context, it estimates the UE position at the intersection of three circumferences centered at the gNBs locations, with radio equal to their respective ranging distance.

Based on the SRS ranging explained above, Rt-5GLoc can predict the distance from each gNB, and hence, perform trilateration. In this case, it is the Euclidean distance between the UE and each gNB.

The position error is defined as the distance between the true UE position, also called ground truth, and its estimate:

$$\Delta u = u - \hat{\mathbf{u}} \tag{3.5}$$

where $\hat{\mathbf{u}}$ is the estimator and u the real UE coordinates (x, y). Assuming $\hat{\mathbf{u}}$ is an unbiased estimator, the covariance matrix of a 2D trilateration is defined by equation 3.6a.

$$\mathbf{C} = \mathbb{E}\left[\Delta\mathbf{u}\Delta\mathbf{u}^T\right] = \begin{bmatrix} \sigma_x^2 & C_{xy} \\ C_{xy} & \sigma_y^2 \end{bmatrix}, \tag{3.6a}$$

$$\mathbf{C} = \sigma_{rang}^2 \left(\mathbf{H}^T\mathbf{H}\right)^{-1}, \tag{3.6b}$$

The diagonal values σ_x^2 and σ_y^2 represent the variance in the obtained UE position in each axes, and C_xy and C_xy covariance between them, which indicates how much the errors vary together. Moreover, the covariance can be rephrased as 3.6b, considering that the modeled channel error σ_{rang} (equation 3.2) from different gNBs is the same and independent [32] [24]. The matrix $\mathbf{H}$ is the Jacobian of the ToA trilateration system, which gives insights into the geometry's impact on position estimation accuracy, explained later in the next section 3.3.2.

The covariance can also be expressed in function of the gNBs arrangement around the target UE. This helps to understand how the variance in each axe changes with the geometry. Equation 3.7 calculates GDOP the matrix by normalizing the covariance with the ranging variance σ_{rang}^2.

$$\mathbf{G} = \frac{1}{\sigma_{rang}^2}\mathbf{C} = \begin{bmatrix} D_x^2 & G_{xy} \\ G_{xy} & D_y^2 \end{bmatrix}, \tag{3.7}$$

The resulting matrix represents the covariances of the UE position, as the ones from $\mathbf{C}$, but only in function of the geometry.

Overall, the accuracy of the trilateration is depicted by the ranging error combined with the geometry of the anchors and the target.

Rt-5GLoc offers a flexible framework to work with different ranging scenarios and geometries that help to understand the errors of a future physical deployment based on the same SRS scheme. The algorithms implemented then are run on the Rt-5GLoc and compared based on these metrics.

3.3.2. NLS algorithm

The NLS is a mathematical solution to find the most probable position of the UE $\hat{\mathbf{u}}$ given a set of measurements ρ_i from N bases stations. $N = 3$ in this case. It obtains the UE position that minimizes the residual distances between the measured ones ρ_i, and the predicted values $h_i(u)$, defined by function $F(\mathbf{u})$.

$$F(\mathbf{u}) = \sum_{i=1}^{N} \left(\rho_{\mathbf{i}} - h_i(\mathbf{u})\right)^2 = \|\boldsymbol{\rho_i} - \mathbf{h}(\mathbf{u})\|^2, \tag{3.8}$$

Where the measurement errors of $\rho_{\mathbf{i}}$, based on SRS and defined at equation 3.2, are Independent and Identically Distributed (IID) and Gaussian, with 0 mean and variance σ^2_{rang} [32]. This holds thanks to the fact that the SRSs were captured in individual experiments in the same setup. Then, it is assumed the errors from different gNBs are independent and distributed by the same distribution.

In conclusion, the NLS UE position estimator, based on the residual distances of equation 3.8, is defined as follows.

$$\hat{\mathbf{u}} = \arg\min_{\mathbf{u}} |\boldsymbol{\rho} - \mathbf{h}(\mathbf{u})|^2 = \arg\min_{\mathbf{u}} \sum_{i=1}^{N} \left(\rho_i - h_i(\mathbf{u})\right)^2 \tag{3.9}$$

An iterative solution for this equation can be defined by algorithm 3.1, defined in [32], which uses the Gauss-Newton numerical search method to find the NLS best UE estimation.

Algorithm 3.1 Iterative NLS for updating $\hat{\mathbf{u}}$

1: **Initialization**: Set $k = 0$, and $\hat{\mathbf{u}}^{(k)} = \hat{\mathbf{u}}^{(0)}$

2: **for** k = 1, 2, ... **do**

3: 1. Compute:

$$\mathbf{H}^{(k)} = \left.\frac{\partial \mathbf{h}(\mathbf{u})}{\partial \mathbf{u}}\right|_{\mathbf{u}=\hat{\mathbf{u}}^{(k-1)}}$$
$$\Delta\rho_i^{(k)} = \rho_i - h_i\left(\hat{\mathbf{u}}^{(k-1)}\right)$$
$$\Delta\boldsymbol{\rho}^{(k)} = \left[\Delta\rho_i^{(k)}\right]_{i=1}^{N}$$

4: 2. Inversion:

$$\Delta\mathbf{u}^{(k)} = \left(\mathbf{H}^{(k)T}\mathbf{H}^{(k)}\right)^{-1}\mathbf{H}^{(k)T}\Delta\boldsymbol{\rho}^{(k)}$$

5: 3. Update the solution:

$$\hat{\mathbf{u}}^{(k)} = \hat{\mathbf{u}}^{(k-1)} + \Delta\mathbf{u}^{(k)}$$

6: 4. Repeat until $\|\Delta\mathbf{u}^{(k)}\| < \varepsilon$ or $k = \text{num\_iter\_max}$

7: **end for**

Where $\mathbf{H} = \left.\frac{\partial \mathbf{h}(\mathbf{u})}{\partial \mathbf{u}}\right|_{\mathbf{u}}$ is the Jacobian matrix for the ToA localization. In this case, it is a $3x2$ matrix, as there are $N = 3$ gNBs in a 2D localization.

$$\mathbf{H} = \begin{bmatrix} \frac{x-x_1}{|\hat{\mathbf{u}}-\mathbf{s_1}|} & \frac{y-y_1}{|\hat{\mathbf{u}}-\mathbf{s_1}|} \\ \frac{x-x_2}{|\hat{\mathbf{u}}-\mathbf{s_2}|} & \frac{y-y_2}{|\hat{\mathbf{u}}-\mathbf{s_2}|} \\ \frac{x-x_3}{|\hat{\mathbf{u}}-\mathbf{s_3}|} & \frac{y-y_3}{|\hat{\mathbf{u}}-\mathbf{s_3}|} \end{bmatrix} \tag{3.10}$$

where $\hat{\mathbf{u}} = (x, y)$ are the estimation coordinates of the UE and $\mathbf{s_i} = (x_i, y_i)$ are the i_{th} gNB ones.

Finally, it was observed that in most cases the algorithm proposed offered a fluctuation of less than $1mm$ for more than 20 iterations, hence this is the maximum number.

3.3.3. Bayesian tracking and Kalman Filter

Bayesian statistical techniques the estimator is based on a probability density function (pdf). In this case, the estimator $p(u_t)$ represents the probable UE location at time t. It is based on a-priori information, which means the current estimation is based on past states of the system. For a 5G mobility scenario, the location is based on the past UE one and its moving velocity. The Bayesian estimate can be defined by:

$$p(u_t|\rho_{1:t}) = \Gamma_t \cdot p(u_t|\rho_{1:t-1}) \cdot p(\rho_t|u_t) \tag{3.11}$$

where $p(u_t|\rho_{1:t})$ is the posterior pdf, $p(u_t|\rho_{1:t-1})$ the prior pdf, $p(\rho_t|u_t)$ the likelihood, and Γ_t is a normalizing constant.

The prior pdf, is defined by the motion model 3.12a, which estates the current user position based on the last one, the motion function $f(\cdot)$ and error w_t.

The likelihood acts as a filter for the next future steps. It is defined as the measurement model 3.12b. Considering the user is at position u_t, it defines the possible set of positions for $t+1$ based on the current one and the measurements (ranging distances obtained through SRS) of $t+1$.

The deterministic functions $f(\cdot)$ and $h(\cdot)$, represented by matrixes $\mathbf{F}$ and $\mathbf{H}$, define the user motion and its probable position based on the measurements, respectively. They embedded Gaussian errors with zero mean and variances σ_{rang} and σ_n.

$$u_t = \mathbf{F}_t x_{t-1} + w_{t-1} \qquad w \sim \mathcal{N}(0, \mathbf{Q}_t), \quad \mathbf{Q}_t = \sigma_n^2 \mathbf{I} \tag{3.12a}$$

$$\rho_t = \mathbf{H}_t + n_t \qquad n \sim \mathcal{N}(0, \mathbf{R}_t), \quad \mathbf{R}_t = \sigma_{rang}^2 \mathbf{I} \tag{3.12b}$$

Finally, the posterior pdf adjusts the predicted state with the likelihood of the next step, as shown by equation 3.11. The UE estimator is the Minimum Mean Square Error (MMSE) of the last calculated posterior pdf:

$$\hat{\mathbf{u}}_t^{\text{MMSE}} = \arg\min_{\hat{x}_t} \mathbb{E}_{u,\rho}\left[\|u_t - \hat{u}_t\|^2 \big| \rho_{1:t}\right] \tag{3.13}$$

When the motion and the measurement models, $f(\cdot)$ and $h(\cdot)$, are linear and the errors, w_t and n_t, are Gaussian with known parameters, then the Kalman filter is a recursive procedure that can be used to calculate the mean and the covariance of posterior pdf. Then, with the Kalman filter, the mean of the posterior pdf represents the $\hat{\mathbf{u}}$ MMSE estimate and the matrixes $\mathbf{Q_t}$ and $\mathbf{R_t}$ are related to the trilateration accuracy [32].

The Kalman implementation, in this case within the Bayesian tracking, is divided into two steps:

- *Prediction*: calculation of the prior UE position and its accuracy (covariance) based on the motion model.

$$\hat{\mathbf{u}}_{t|t-1} = \mathbf{F}_t \hat{\mathbf{u}}_{t-1|t-1} \tag{3.14a}$$

$$\mathbf{C}_{t|t-1} = \mathbf{F}_t C_{t-1|t-1} \mathbf{F}_t^T + \mathbf{Q}_t \tag{3.14b}$$

- *Update*: new position estimation of the UE based on the prediction and the measurement model weighted by the Kalman gain matrix $\mathbf{G}$.

$$\hat{\mathbf{u}}_{t|t} = \hat{\mathbf{u}}_{t|t-1} + \mathbf{G}_t(\rho_t - \mathbf{H}_t \hat{\mathbf{u}}_{t|t-1}) \tag{3.15a}$$

$$\mathbf{C}_{t|t} = \mathbf{C}_{t|t-1} - \mathbf{G}_t \mathbf{H}_t \mathbf{C}_{t|t-1} \tag{3.15b}$$

$$\mathbf{G}_t = \mathbf{C}_{t|t-1}\mathbf{H}_t^T(\mathbf{H}_t \mathbf{C}_{t|t-1}\mathbf{H}_t^T + \mathbf{R}_t)^{-1} \quad \text{Kalman gain} K \times N \tag{3.15c}$$

The new variable $\mathbf{C}_{t|t-1}$ represents the covariance of the estimator, hence, the UE position squared error. Again, $\mathbf{H}$ is the ToA Jacobian matrix, $\hat{\mathbf{u}}_{t|t}$ the UE estimated position at time t and $\rho_\mathbf{t}$ the measurement matrix $1x3$, holding the ranging distances calculated through SRS of the 3 gNBs. Finally, the motion is defined by the matrix $\mathbf{F} = \mathbf{I_2}$ representing the Nearly-constant position (NcP) model, where the UE is in a static position during the trilateration.

In resume, the Kalman Filter implemented, based on Bayesian tracking, is a procedure to perform ToA trilateration estimates and update them recursively. For this case, the UE is assumed to follow a NcP motion model and the ranging measurements come from SRS signals. The parameters of the Kalman filter were then tuned for a static UE and

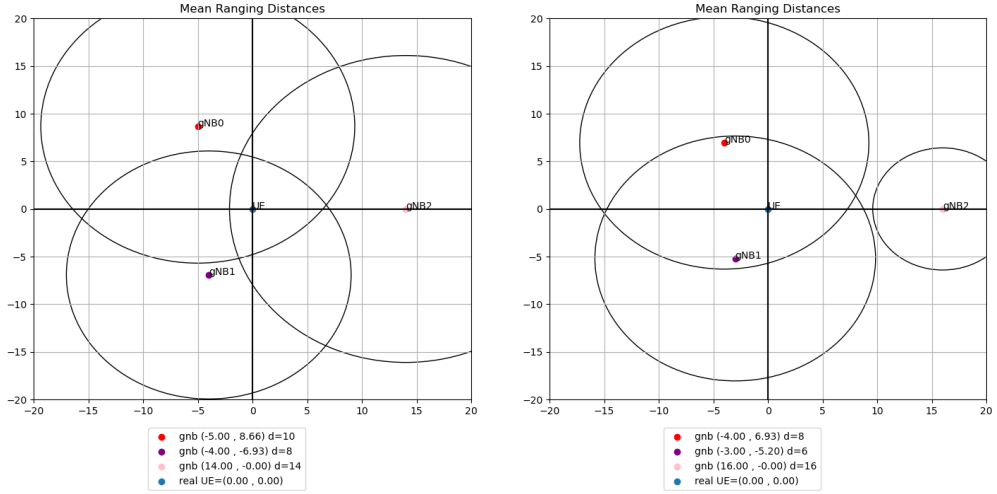
the mean ranging error of the SRS experiments.

Initial Position	[0,0]
Max step variance (σ_n^2)	3^2 m
Measurement Error (σ_{rang}^2)	3^2 m

Table 3.2: Initial Kalman Filter Parameters.

3.3.4. Localization system constrains

Considering that all SRSs were collected in the same experimental setup, all gNBs emulators should have the same ranging error. In reality, each SRS ranging experiment has a different variance, given that they were captured in individual experiments. The quality of the captured SRS affects the ToA circumference of that emulator, affecting the trilateration accuracy. If the SRS ranging error is high enough that the distance obtained is non-coherent, the trilateration results are not possible.



(a) Healthy case of intersection with ranging error

(b) Not realiable case, not enough intersections due to ranging error

Figure 3.14: Example of trilateration intersection cases. On the left, healthy case with ranging error, and on the right, the trilateration is not considered given the lack of intersection between all the ranges.

Section 2.2.1 shows cases where the ranging error affects the trilateration intersections and references some solutions to deal with the different cases where there is no intersection

between them. For this study, Rt-5GLoc measurements are defined as reliable as long as there are intersections between all range distances. Therefore, the cases where there is no intersection, as shown by the right case of Figure 3.14, are not considered for this study.

4 | Results

The experiments performed with Rt-5GLoc aim to analyze the impact of the ranging error and the gNB arrangement in different 5G SA trilateration scenarios. As explained, the system is based on 3 gNB emulators fed by real-world SRS signals, captured in a 5G SA environment. The gNB emulators act as real-world stations from the xApp perspective and avoid acquiring three gNBs, dealing with the nuances of the PHY and their synchronization. With all said, the results obtained expect to reproduce the analysis of a deployment with 3 real gNBs, help to analyze the final localization accuracy, the best geometry scenario and the ranging errors influence while implementing the SRS trilateration proposed.

The conditions where the empirical SRSs were collected are detailed in Section 3.1.1 and how they are translated into a time delay, between the UE and the gNB, and then into the distance are in Section 3.1.2.

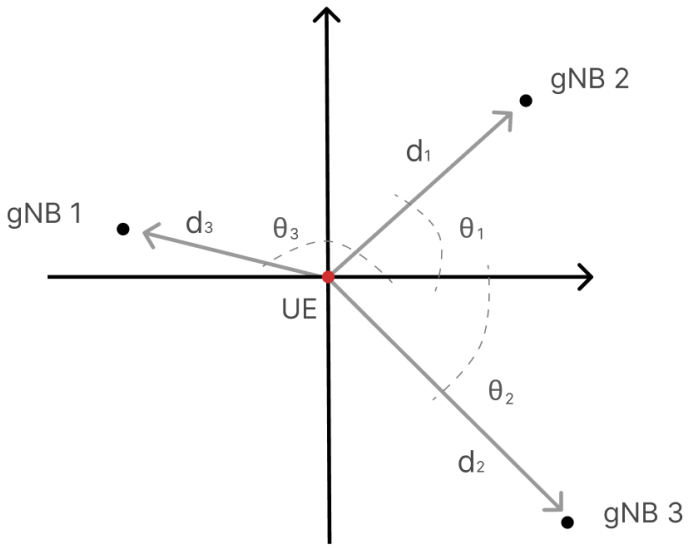


Figure 4.1: Rt-5GLoc SRS ToA localization experiment

In this context, a Rt-5GLoc experiment is assembled by 3 gNBs emulators and a set of

empirical SRSs, originated from the same reference signal and collected by a single station at different distances from the UE. The SRSs captured then define the possible distances d_1, d_2 and d_3 that the gNBs can assume in the experiment. In addition, the geometry can be changed based on the input angles θ_1, θ_2 and θ_3. Figure 4.1 shows how these 6 parameters are set to build a scenario.

4.0.1. Experiments with Kalman Filter

The SRS collected, at different gNB-UE distances, provided 3000 signals for each trial. When performing the SRS ranging, explained in section 3.1.1, the average results were the following:

Ranging experiment	**1**	**2**	**3**	**4**	**5**	**6**
Real d_{UE-gNB} [m]	6	8	10	12	14	16
Mean d_{UE-gNB} obtained through SRS ranging [m]	9,42	11,34	7,81	2,29	14,35	9,47
Error %	57,07	41,79	21,82	80,84	2,55	40,78

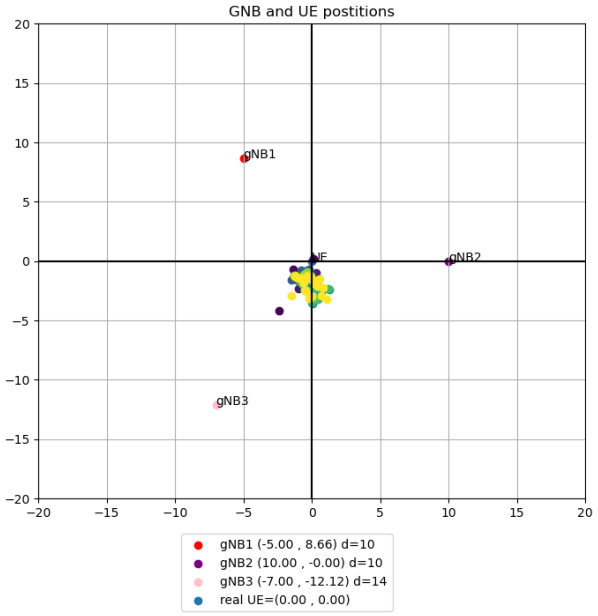
Table 4.1: Mean error of the ranging experiments

For a given Rt-5GLoc localization scenario, the gNB emulator fed with the sequences the i_{th} experiment provides the UE-gNB distance of that trial. In this way, these SRSs were used to set the distances d_1, d_2 and d_3 for each gNB, with the respective angles θ_1, θ_2 and θ_3. Accordingly, several trilateration experiments were performed to understand how the Kalman approach, explained in Section 3.3.3, performs in different circumstances of ranging errors and gNB geometries.

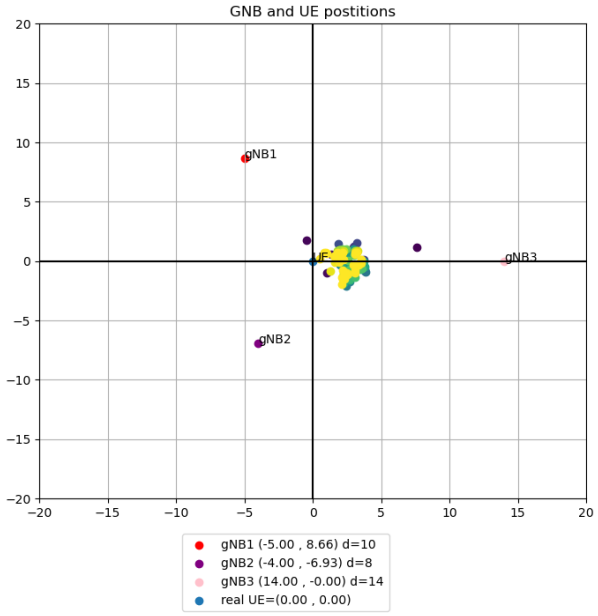
Impact of the SRS ranging error

In the trilaterations performed, the accuracy of the UE coordinates is the highest when the gNBs provided the more accurate SRS experiments, of Table 4.1. For this study, a ranging error around, or smaller than, 20% is considered a good ranging accuracy. This is, when the gNBs are set at 10m and 14m from the UE, 3_{th} and 5_{th} trials. A ranging error between 20% and 60%, experiments done at 6m,8m, and 16m, is categorized as medium quality ranging accuracy. Finally, the localization scenarios where a gNB is set at 12m from the UE, the 4_{th} trial, are considered to have pour quality ranging as 80% of distance error is seen too high to implement trilateration.

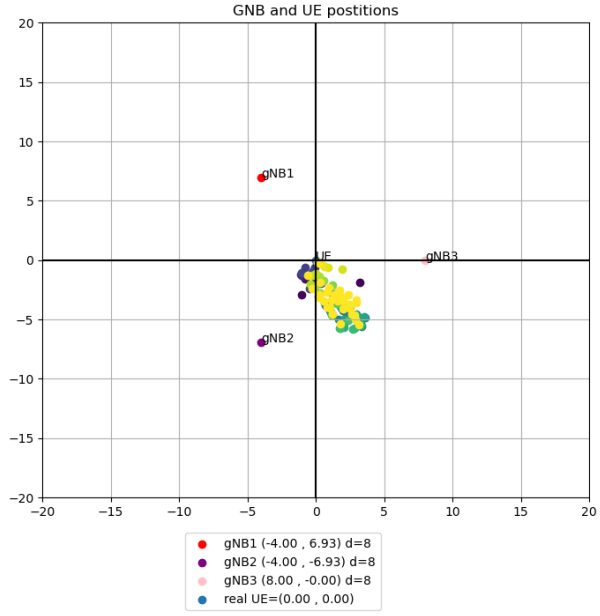
The trilateration scenarios based on good and medium SRS ranging qualities have the smaller Root Mean Square Error (RMSE) for the obtained UE position. This is expected as the trilateration error increases proportionally to the ranging error, shown by equation 3.6a.



(a) Good quality measurements



(b) One station with medium quality measurements



(c) Three stations with medium quality measurements

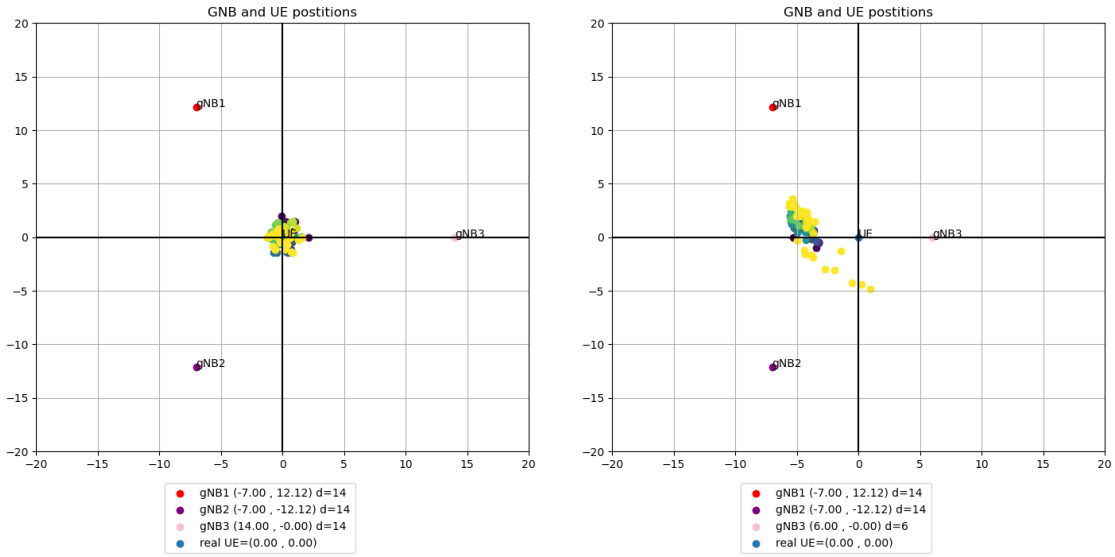
Figure 4.2: Examples of Rt-5GLoc ranging accuracy emulations, for the same geometry ($\theta_1 = 120°, \theta_2 = 240°$ and $\theta_3 = 0$)

Figure 4.2 shows 3 trilateration scenarios performed with 100 SRS ranging measurements each. They all have the same geometry: $\theta_1 = 120°, \theta_2 = 240°$ and $\theta_3 = 0$, varying only the individual gNB-UE distances, hence SRS trials, hence ranging accuracies. The 4.2a trilateration, where $d_1 = 10m, d_2 = 10m$ and $d_3 = 14m$, has only stations with good ranging. Scenario 4.2b substitutes one of the stations with a medium ranging quality, $d_2 = 8m$, and in 4.2c all three stations, with the 2_{th} SRS trial and $d_{UE-gNB} = 8m$. They show how the introduction of stations with worse ranging increases the spread of the obtained UE coordinates. The substitution of one station with medium ranging, in scenario 4.2b compared to 4.2a, increased the UE error by almost 1m. The case where all stations have medium-ranging quality, increased the total error to $3,5m$, as shown by Table 4.2.

$d_1 - d_2 - d_3$ **[m]**	**RMSE [m]**	**Variance [m^2]**	**Ranging accuracy class**
14-14-14	0,97	0,201	Good
14-14-10	2,05	0,344	Good
10-10-14	2,07	0,590	Good
10-8-14	2.73	0,681	Medium
6-8-10	2,58	3,765	Medium
10-14-6	3,10	0,483	Medium
14-14-6	4,68	0,718	Medium
8-8-8	3,53	2,953	Medium
6-8-8	2,64	4,125	Medium

Table 4.2: Rt-5GLoc trilateration experiments with same geometry, $\theta_1 = 120°, \theta_2 = 240°$ and $\theta_3 = 0$, and different gNBs-UE distances. The ranging accuracy class is defined by the lowest one between the 3 gNBs

Table 4.2 collects all the experiments performed with the same geometry but different d_{UE-gNB} distances. It lists their RMSE and their variance, this is the spread of the calculated UE coordinates. All the trilateration scenarios were simulated with 100 SRS in each gNB. Then, these experiments can be considered random trials within the scope of the 3000 total SRS collected for each distance. The smaller RMSE trilaterations are the ones including only stations with a ranging error smaller than 20% (good ranging). As stations with less accurate ranging are introduced, the obtained UE coordinates tend to have a higher spread and RMSE. Figure 4.3 shows the visual results of this case, where the UE coordinates tend to have a higher variance when the ranging of the stations is less accurate.

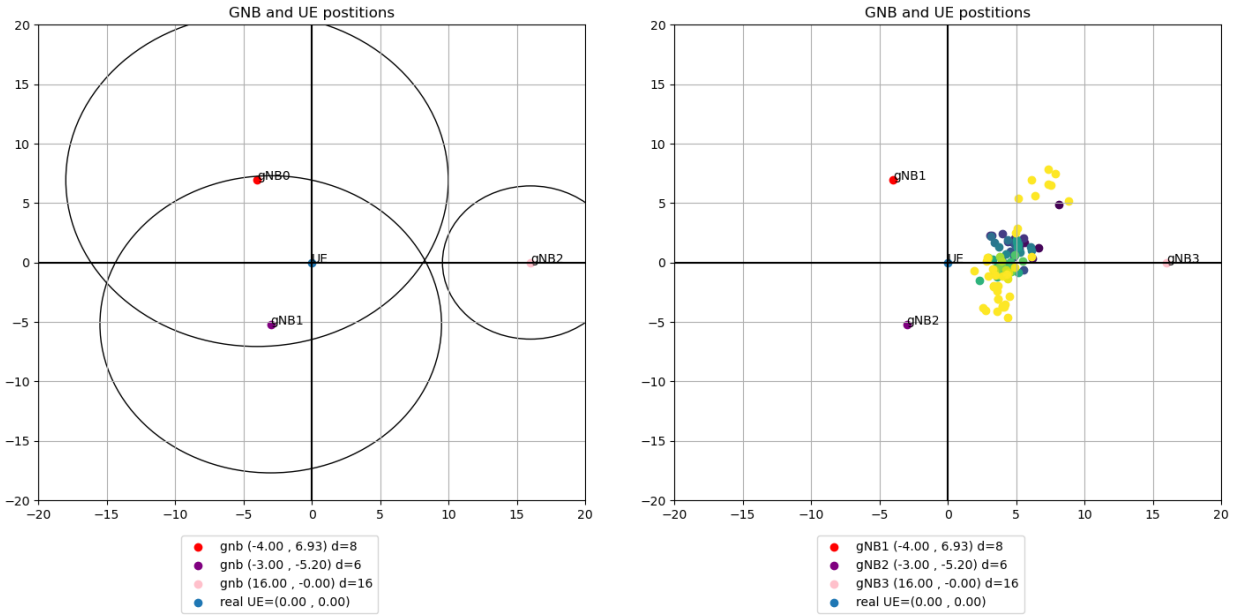


(a) Less spread results due to better ranging. SRS ranging error of each station: $gNB_1 = gNB_2 = gNB_3 = 2,55\%$

(b) Introduction of one gNB with worse ranging. SRS ranging error of each station: $gNB_1 = gNB_2 = 2,55\%$ $gNB_3 = 57,07\%$

Figure 4.3: Spread comparison between good and medium-ranging quality

In addition, in the scenarios where the gNBs were positioned farther away from the UE, 12m, and 16m, the ranging errors impacted the number of intersections.



(a) Medium ranging distances

(b) The obtained UE position is biased by gNB3

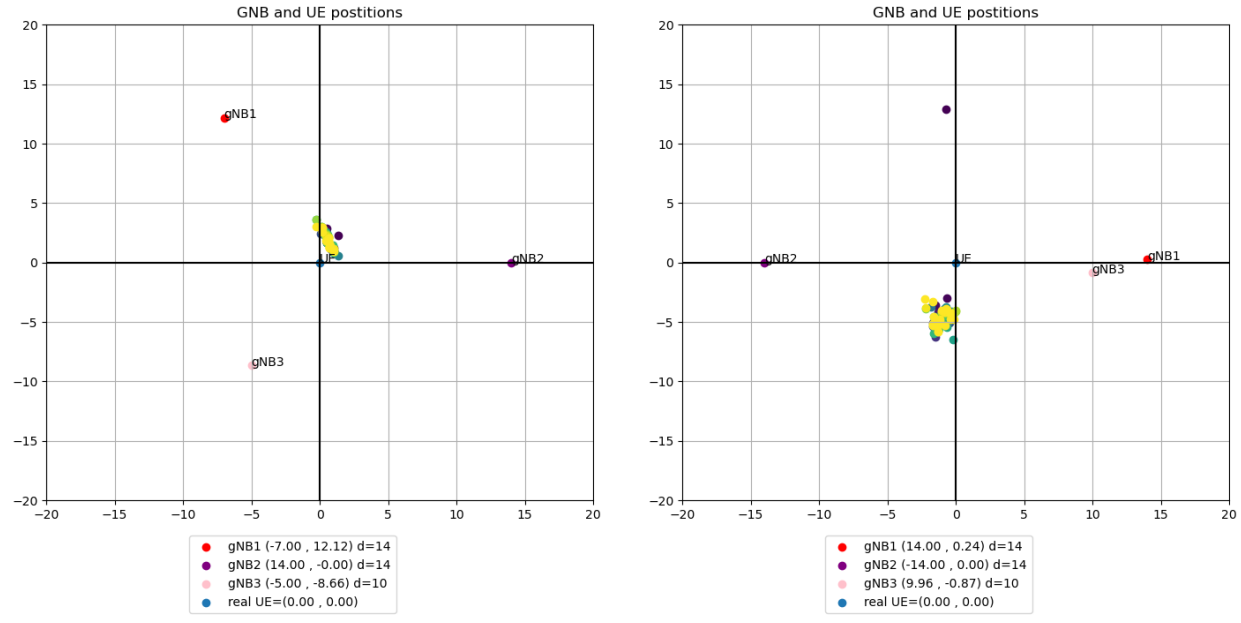
Figure 4.4: Experiment with not enough intersections

The cases where the stations were far and had enough ranging error did not cause an intersection, between all the station circumferences, the trilateration is biased and is not considered for this study. Figure 4.3 shows an example of this case, where $d_1 = 8m, d_2 = 6m$ and $d_3 = 16m$. The left figure shows the lack of intersections that cause a bad trilateration accuracy. For this example, the UE position RMSE is 5,0614m.

In resume, as expected, the experiments with best-ranging accuracy, gNBs at distances 10m and 14m from the UE, are the ones with the lowest spread, and RMSE for the obtained UE coordinates. These trilaterations reach a UE localization error equal to or smaller than 2,07m.

Impact of the gNB geometry

Another set of experiments was performed to analyze the impact of the gNBs geometry in the trilateration accuracy. As explained in section 3.1, more uniform geometries have a better performance. This means the best case is represented by 3 equidistant gNBs, to the target UE, with 120° between them. The previous combinations of SRS ranging distances were set under a *uniform* and a *linear* geometry. Figure 4.5 shows the angle definition for both.



(a) Uniform geometry

$\theta_1 = \theta_2 = \theta_3 = 120°$

$$GDOP_{uniform} = \begin{bmatrix} 0.667 & -7.40 \times 10^{-17} \\ -7.40 \times 10^{-17} & 0.667 \end{bmatrix}$$

(b) Linear geometry

$\theta_1 = 1°, \theta_2 = 180°, \theta_3 = 355°$

$$GDOP_{linear} = \begin{bmatrix} 0.41 & 3.68 \\ 3.68 & 158.92 \end{bmatrix}$$

Figure 4.5: Examples of Rt-5GLoc emulations with different geometry

In that scenario, the *linear* geometry has a larger variance over the y-axis $\sigma_y^2 = 158.92$ compared to the *uniform* geometry $\sigma_y^2 = 0.667$, as shown at the GDOP matrix. The uncertainty over the y-axis is explained by the lack of orthogonality of the gNBs positions. Table 4.2 compiles the results for all the experiments.

$d_1 - d_2 - d_3$	**RMSE [m]**		
	$geo_{uniform}$	geo_{linear}	$geo_{linear}/geo_{uniform}$
14-14-14	0,97	3,97	4,09
10-10-14	2,07	5,40	2,61
14-14-10	2,05	5,19	2,53
6-8-10	2,58	7,79	3,02
6-8-8	2,64	7,75	2,94
10-8-14	2,71	6,70	2,47
10-14-6	3,10	5,54	1,79
8-8-8	3,53	5,94	1,68
14-14-6	4,68	5,17	1,10

Table 4.3: Rt-5GLoc trilateration experiments and analysis on geometry accuracy

One thing to notice is how the relation between the most favorable geometry and the *linear* one decreases when gNBs with worse ranging error, 6m and 8m, are introduced. This means that, when the ranging error of the stations is high enough, the changes in the gNBs arrangement do not improve that much the localization performance. In addition, when the SRS ranging measurements are more accurate, the error between a disadvantageous geometry, *linear*, and a favorable one, *uniform*, decreases between 4 or 2 times.

In resume, the geometry of the gNBs around the UE considerably impacts the UE position estimation. In scenarios where the UE is positioned closer to the center of the arrangement, the results tend to be better, as long as the gNBs are not aligned. The introductions of one gNB with worse ranging in the SRS trilateration can add errors of more than 2m in the final UE mean position. In the best cases, Rt-5GLoc can calculate the UE position with an error equal to or smaller than 2,07m. Finally, the trilateration system proposed is sensible to the gNBs geometry but more even to the quality of the ranging measurements, this is how accurately the SRS can detect the delay between the UE and the station.

4.0.2. NLS vs Kalman

In all cases, the Kalman approach performed better than NLS. NLS is less suited for real-time tracking, as it generally requires iterative optimization over bigger datasets to minimize the sum of squared errors. For the localization experiments performed with 100 SRSs, the comparison between the two algorithms is the following.

$d_1 - d_2 - d_3$	**Kalman RMSE [m]**	**NLS RMSE [m]**
14-14-14	0,97	2,07
10-10-14	2,07	3,20
14-14-10	2,05	4,14
10-14-6	3,10	9,78
8-8-8	3,53	6,04
6-8-10	2,58	11,19
6-8-8	2,64	11,00
14-14-6	4,68	7,95
10-8-14	2,71	8,40

Table 4.4: Rt-5GLoc trilateration comparison of algorithms for the same geometry ($\theta_1 = 120°, \theta_2 = 240°$ and $\theta_3 = 0$) and ranging accuracies

Table 4.4 collects the result's comparison of the previous trilaterations performed with *uniform* geometry, $\theta_1 = 120°, \theta_2 = 240°$ and $\theta_3 = 0$, for the Kalman and the NLS approaches. For all trials, with 100 SRS, the Kalman model showed better accuracy. For the NLS, the RMSE rapidly increases when stations with worse ranging are introduced, reaching UE localizations errors up to 11m. Compared to the Kalman approach with good ranging accuracies, where the UE position can be obtained with an error up to 2,07m, the NLS reaches errors up to 4m for the same scenarios.

On the other hand, NLS calculations were faster to obtain the UE coordinates. NLS algorithm performs better over bigger datasets, compared to the Kalman that corrects recursively the results for each iteration. The NLS is not recursive, instead, it calculates the UE position that better reduces the RMSE for all the SRS ranging distance measurements available. Hence, for larger-ranging datasets, NLS might be a better option. For this study, nevertheless, the Kalman approach is a better fit.

Figure 4.6 shows the time it takes to calculate each iteration of 600 SRS ranging measurements for both models for a given trilateration scenario. In this case, the ranging

accuracies and their geometry are not relevant. Both, up and down figures, show the same scenario with different scales for the time axis for each iteration (y-axis). The first iterations of Kalman take the most time to be computed, while the NLS tends to be more constant.

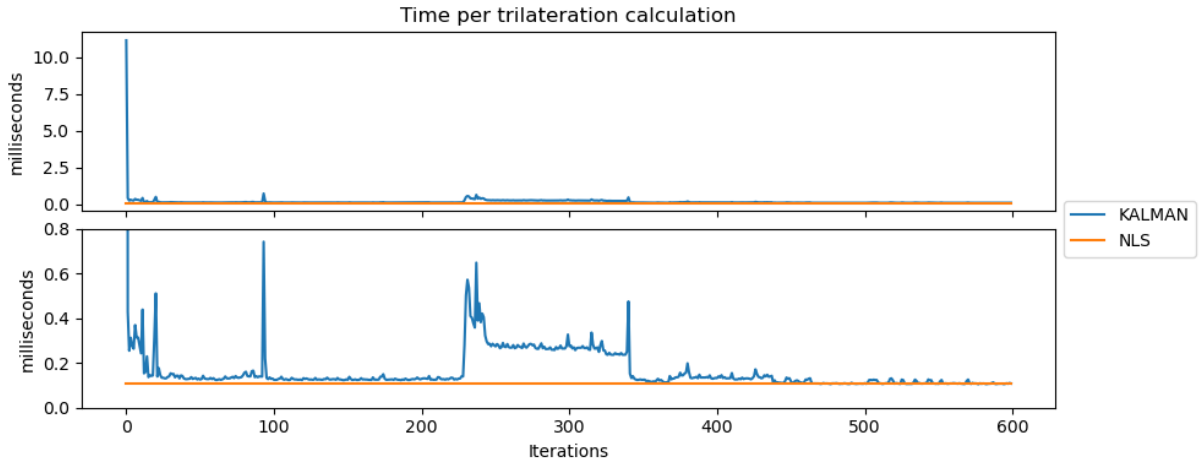


Figure 4.6: Example of 1 trilateration experiment with 600 SRS

NLS time to calculate (x,y) UE coordinates is more stable and lower for all the iterations. Kalman, however, takes more time for each iteration but performs better for all scenarios, being a better solution for the SRS localization proposed.

Overall, the Kalman filter based on Bayesian tracking has a better performance than NLS, the good ranging measurements, with a ranging error smaller or around 20% the gNB-UE distance, can obtain a trilateration accuracy up to 2,07m. Finally, the best performing arrangement is where the gNBs draw an equilateral triangle with the UE centered.

5 | Conclusions and Future Works

5.1. Conclusions

This thesis introduced a 5G localization approach that innovatively leverages SRS within the O-RAN framework, building from zero a trilateration xApp on the near-RT RIC to obtain real-time UE positioning. By employing real-world SRS signals, fed into gNB emulators that function as real-world stations within the O-RAN structure, the experiments revealed a final UE localization error around 2 meters, for gNB SRS ranging measurements errors smaller or equal to 20% their distance to the UE. Particularly, when the gNBs formed an equilateral triangle around the UE, the results had the smallest error and variance for the calculated UE position. Studying the behavior of Rt-5GLoc and improving its accuracy in response to different scenarios, we evaluated two algorithms through different signal quality: the NLS and the Kalman Filter. In conclusion, compared to the NLS approach, the Kalman filter provided higher accuracy, at the cost of increased computation time for each iteration.

This work highlights the innovation of using SRS within the O-RAN ecosystem, a step forward in integrating programmable network elements in MRN for precise and adaptable localization.

In conclusion, this study underscores the potential of 5G localization based on SRS signals, especially within the O-RAN framework and the new programmable approach of MRNs. With improvements in physical testbeds and synchronization techniques, future iterations of this system could enhance real-time, high-precision localization in dynamic network environments.

5.2. Future Works

Future work will involve deploying this 5G SA localization system with three gNBs. From this deployment we expect similar localization accuracy to the 2-meter error observed in the current study, affirming the reliability of SRS trilateration for UE positioning. Moving

to a real environment will also address the nuances of synchronization across the gNBs, introducing refinements in station synchronization and signal alignment at the physical layer.

This use of SRS-based trilateration, supported by the xApp within the O-RAN framework, represents an innovative approach to deal with localization in the NG-RAN. Overall, the system proposed supports collaborative innovation in open RAN localization systems, reinforcing the adaptability and robustness of 5G localization infrastructure for dynamic, real-time applications.

Bibliography

[1] O-RAN E2GAP Specification, Version 5.0. Technical Report O-RAN.WG3.E2GAP-R003-v05.00, O-RAN ALLIANCE e.V., Alfter, Germany, 2024. Copyright © 2024 by the O-RAN ALLIANCE e.V.

[2] 3rd Generation Partnership Project (3GPP). Release 18 description; summary of rel-18 work items. Technical report, 3GPP, 2024. URL `https://www.3gpp.org/specifications/work-plan`. TR21.918, specifies improvements and new topics for Release 18 of the 5G-Advanced system, including satellite access, IoT, MTC, and more. The document is in production with summary notes and is available through the 3GPP Portal. © 3GPP 2024.

[3] 3rd Generation Partnership Project (3GPP). Location and Positioning Technologies. `https://www.3gpp.org/technologies/location-and-positioning`, n.d. Accessed: 2024-11-12.

[4] A. M. Abba, J. Sanusi, O. Oshiga, and S. A. Mikail. A review of localization techniques in wireless sensor networks. *2nd International Conference on Multidisciplinary Engineering and Applied Sciences (ICMEAS-2023)*, 2023. doi: 10.1109/ICMEAS58693.2023.1042988.

[5] M. Abbas, M. Elhamshary, H. Rizk, M. Torki, and M. Youssef. Wideep: Wifi-based accurate and robust indoor localization system using deep learning. *IEEE International Conference on Pervasive Computing and Communications (PerCom)*, pages 1–10, 2019. doi: 10.1109/PERCOM.2019.8767421. URL `https://doi.org/10.1109/PERCOM.2019.8767421`.

[6] N. A. Alrajeh, M. Bashir, and B. Shams. Localization techniques in wireless sensor networks. *International Journal of Distributed Sensor Networks*, 2013:9 pages, 2013. doi: 10.1155/2013/304628. URL `http://dx.doi.org/10.1155/2013/304628`.

[7] V. Bernazzoli, E. Moro, and I. Filippini. 5g ranging: Towards flexible positioning services. In *Proceedings of the CoNEXT-SW '23*. Politecnico di Milano, Milan, Italy,

ACM, December 2023. ISBN 979-8-4007-0452-9. doi: 10.1145/3630202.3630239. URL https://doi.org/10.1145/3630202.3630239.

[8] S. C. Bhardwaj, S. Shekhar, A. Vidyarthi, and R. Prakash. Satellite navigation and sources of errors in positioning: A review. In *Proceedings of the IEEE Conference*, Dehradun, India, 2020. Department of Electronics and Communication, Graphic Era University, IEEE. ISBN 978-1-7281-9785-2. doi: 10.1109/CONFERENCE.2020.

[9] B. Bojović, S. Lagén, and L. Giupponi. Realistic beamforming design using srs-based channel estimate for ns-3 5g-lena module. In *Proceedings of the 2021 Workshop on Ns-3*, WNS3 '21, page 81–87, New York, NY, USA, 2021. Association for Computing Machinery. ISBN 9781450390347. doi: 10.1145/3460797.3460809. URL https://doi.org/10.1145/3460797.3460809.

[10] L. Bonati, M. Polese, S. D'Oro, S. Basagni, and T. Melodia. OpenRAN Gym: An Open Toolbox for Data Collection and Experimentation with AI in O-RAN. In *IEEE WCNC 2022 Workshop on Open RAN Architecture for 5G Evolution and 6G*. Institute for the Wireless Internet of Things, Northeastern University, 2022. URL https://arxiv.org/abs/2202.10318v1. arXiv:2202.10318v1 [cs.NI].

[11] F. Borg. Increasing energy efficiency in o-ran compliant radio access networks using ric and xapps. Advanced level thesis, 20 credits, Luleå University of Technology, Department of Computer Science, Electrical and Space Engineering, 2023. URL https://urn.kb.se/resolve?urn=urn:nbn:se:ltu:diva-98204. Supervised by Professor Karl Andersson; External cooperation with Tietoevry.

[12] V. Cantón Paterna, A. Calveras Augé, J. Paradells Aspas, and M. A. Pérez Bullones. A bluetooth low energy indoor positioning system with channel diversity, weighted trilateration and kalman filtering. *Sensors*, 17(12), 2017. ISSN 1424-8220. doi: 10.3390/s17122927. URL https://www.mdpi.com/1424-8220/17/12/2927.

[13] A. Capone. Mrn - 4g. PowerPoint presentation, 054324 - Mobile Radio Networks, 054328 - Wireless Networks [Sezione A] [2022-23], 2023. Accessed: August 23, 2024.

[14] A. Capone. Mrn - 5g. PowerPoint presentation, 054324 - Mobile Radio Networks, 054328 - Wireless Networks [Sezione A] [2022-23], 2023. Accessed: September 02, 2024.

[15] C. Cox. *An Introduction to 5G: The New Radio, 5G Network and Beyond.* John Wiley & Sons Ltd, Chichester, West Sussex, UK, 2021.

[16] L. L. de Oliveira, G. H. Eisenkraemer, E. A. Carara, J. B. Martins, and J. Monteiro.

Mobile localization techniques for wireless sensor networks: Survey and recommendations. *ACM Transactions on Sensor Networks*, 19(2):36:1–36:39, April 2023. doi: 10.1145/3561512.

[17] Ericsson. 5g sa deployment: Moving beyond embb. Technical Report EAB-22:005355 Uen/5, Ericsson, SE-164 80 Stockholm, Sweden, June 2022. URL `https://www.ericsson.com/mobility-report`. © Ericsson 2022.

[18] Ericsson. Ericsson mobility report. Technical Report EAB-24:004434 Den Rev D, Ericsson, SE-164 80 Stockholm, Sweden, June 2024. URL `https://www.ericsson.com/49ed78/assets/local/reports-papers/mobility-report/documents/2024/ericsson-mobility-report-june-2024.pdf`. © Ericsson 2024.

[19] Global mobile Suppliers Association. 5g standalone: Global status update. Technical report, Global mobile Suppliers Association, P.O. Box 6092, Sheffield, S6 9HF, UK, July 2023. URL `https://gsacom.com`. © Global mobile Suppliers Association. 2023, Published on 28th July 2023.

[20] N. Heydarishahreza and N. Ansari. Mobile node localization in wireless networks: Path-loss model, trilateration, and error mitigation in a 5g sub-6 ghz scenario. *Journal of Networking and Network Applications*, 3(3):129–138, December 2023. Corresponding author: Navid Heydarishahreza.

[21] M. Hua, M. Wang, K. W. Yang, and K. J. Zou. Analysis of the frequency offset effect on zadoff–chu sequence timing performance. *IEEE Transactions on Communications*, 62(11):4024–4039, 2014.

[22] F. Kaltenberger, A. P. Silva, A. Gosain, L. Wang, and T.-T. Nguyen. Openairinterface: Democratizing innovation in the 5g era. *Computer Networks*, 176:107284, 2020. ISSN 1389-1286. doi: https://doi.org/10.1016/j.comnet.2020.107284. URL `https://www.sciencedirect.com/science/article/pii/S1389128619314410`.

[23] L. Kundu, G. Xiong, and J. Cho. Physical uplink control channel design for 5g new radio. In *2018 IEEE 5G World Forum (5GWF)*, pages 233–238, 2018. doi: 10.1109/5GWF.2018.8517042.

[24] R. B. Langley. Dilution of precision. *GPS WORLD*, 10(5), 1999. URL `http://www.gpsworld.com`.

[25] J. Li, X. Yue, J. Chen, and F. Deng. A novel robust trilateration method applied to ultra-wide bandwidth location systems. *Sensors*, 17(4), 2017. ISSN 1424-8220. doi: 10.3390/s17040795. URL `https://www.mdpi.com/1424-8220/17/4/795`.

[26] G. LLC. Protocol buffers overview. `https://protobuf.dev/overview/`, 2024. Accessed: 2024-09-19.

[27] MEF. Slicing for shared 5g fronthaul and backhaul. White Paper 202003, MEF Forum, April 2020. © MEF Forum 2020. Any reproduction of this document, or any portion thereof, shall contain the following statement: "Reproduced with permission of MEF Forum.".

[28] E. Moro, M. Polese, A. Capone, and T. Melodia. An open ran framework for the dynamic control of 5g service level agreements. *arXiv preprint arXiv:2309.07508*, 2023.

[29] I. Motorola. Long term evolution (lte): A technical overview. Technical white paper, Motorola, Inc., 2007. URL `http://www.motorola.com`. © Motorola, Inc. 2007.

[30] P. Müller, J. A. del Peral-Rosado, R. Piché, and G. Seco-Granados. Statistical trilateration with skew-t distributed errors in lte networks. *IEEE Transactions on Wireless Communications*, 15(10):7114–7127, 2016. doi: 10.1109/TWC.2016.2597836.

[31] M. V. Ngo, N.-B.-L. Tran, H.-M. Yoo, Y.-H. Pua, T.-L. Le, X.-L. Liang, B. Chen, E.-K. Hong, and T. Q. Quek. Ran intelligent controller (ric): From open-source implementation to real-world validation. *ICT Express*, 2024. ISSN 2405-9595. doi: 10.1016/j.icte.2024.02.001. URL `https://www.sciencedirect.com/science/article/pii/S240595952400001X`. Peer review under responsibility of The Korean Institute of Communications and Information Sciences (KICS). Open access article under CC BY-NC-ND license.

[32] M. Nicoli. Radio localization: Algorithms for position estimation. Presentation, Localization, Navigation and Smart Mobility, Telecommunication Engineering, 2023. Course: 054309 - Localization, Navigation and Smart Mobility, Politecnico di Milano.

[33] M. Nicoli. Ranging: Time of arrival estimation, localization, navigation and smart mobility. Presentation, Telecommunication Engineering, 2023. Course: 054309 - Localization, Navigation and Smart Mobility, Politecnico di Milano.

[34] M. Nicoli. Radio localization: Parameters. *Lecture Notes from Localization, Navigation and Smart Mobility, Telecommunication Engineering*, Politecnico di Milano, 2023. Course code: 054309 - Localization, Navigation and Smart Mobility [2022-23].

[35] M. Nicoli. Ranging: Time of arrival estimation. Presentation, Localization, Navigation and Smart Mobility, Telecommunication Engineering, 2023. Course: 054309 - Localization, Navigation and Smart Mobility, Politecnico di Milano.

[36] O-RAN Project. *xApp Python Framework: Framework Overview*. O-RAN Software Community (O-RAN SC), 2023. URL `https://docs.o-ran-sc.org/projects/o-ran-sc-ric-plt-xapp-frame-py/en/latest/`. Revision 13030cc3.

[37] O-RAN Software Community. sim-e2-interface repository. `https://github.com/o-ran-sc/sim-e2-interface`, 2022. [Online; accessed 19-Sep-2024].

[38] O-RAN Work Group 3 (Near-RT RIC and E2 Interface). E2 general aspects and principles (e2gap). Technical Specification O-RAN.WG3.E2GAP-R003-v05.00, O-RAN ALLIANCE e.V., Buschkauler Weg 27, 53347 Alfter, Germany, 2024. Copyright © 2024 by the O-RAN ALLIANCE e.V.

[39] O-RAN Working Group 3. O-RAN near-real-time RAN intelligent controller E2 service model (E2SM) KPM 2.0. Technical Specification ORAN-WG3.E2SM-KPM-v02.00, O-RAN, Alfter, Germany, Jul 2021.

[40] O-RAN Working Group 3. O-RAN E2 service model (E2SM), cell configuration and control 1.0. Technical Specification O-RAN.WG3.E2SM-CCC-v01.00, O-RAN, Alfter, Germany, Oct 2022.

[41] OAIC. *O-RAN Near-Real Time RIC Installation Guide*, 2022. Built with Sphinx using a theme provided by Read the Docs. Accessed: Sep. 17, 2024.

[42] M. Polese, L. Bonati, S. D'Oro, S. Basagni, and T. Melodia. Understanding o-ran: Architecture, interfaces, algorithms, security, and research challenges. *IEEE Communications Surveys and Tutorials*, 2023. doi: 10.1109/COMST.2023.3239220.

[43] A. Poulose, O. S. Eyobu, and D. S. Han. A combined pdr and wi-fi trilateration algorithm for indoor localization. In *2019 International Conference on Artificial Intelligence in Information and Communication (ICAIIC)*, pages 072–077, 2019. doi: 10.1109/ICAIIC.2019.8669059.

[44] S. Pradhan, S. Shin, G.-R. Kwon, J.-y. Pyun, and S.-s. Hwang. The advanced toa trilateration algorithms with performance analysis. In *2016 50th Asilomar Conference on Signals, Systems and Computers*, pages 923–928, 2016. doi: 10.1109/ACSSC.2016.7869184.

[45] Professor Andy Sutton, CEng FIET. 5g ran architecture evolution. Technical report, Architecture & Strategy, Principal Network Architect, January 2019. Presentation on topics including Antennas, Radio Wave Propagation, Mobile Technology, UMTS 3G Systems, LTE, 4G and 5G Technologies, and Network Security.

[46] M. M. H. Qazzaz, L. Kulacz, A. Kliks, S. A. Zaidi, M. Dryjanski, and D. McLer-

non. Machine learning-based xapp for dynamic resource allocation in o-ran networks. *arXiv preprint arXiv:2401.07643*, page 6, 2024. doi: 10.48550/arXiv.2401.07643. 6 figures, 2024 IEEE International Conference on Machine Learning for Communication and Networking (ICMLCN).

[47] J. F. Santos, A. Huff, D. Campos, K. V. Cardoso, C. B. Both, and L. A. DaSilva. Managing o-ran networks: xapp development from zero to hero. *arXiv preprint arXiv:2407.09619v2 [cs.NI]*, Aug 2024. URL `https://arxiv.org/abs/2407.09619v2`. This work has been submitted to the IEEE for possible publication. Copyright may be transferred without notice, after which this version may no longer be accessible.

[48] H.-T. Thieu, V.-Q. Pham, A. Kak, and N. Choi. Demystifying the near-real time ric: Architecture, operations, and benchmarking insights. In *IEEE INFOCOM 2023 - IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, pages 1–8, 2023. doi: 10.1109/INFOCOMWKSHPS57453.2023.10225852.

[49] S. Venkatraman, J. Caffery, and H.-R. You. A novel toa location algorithm using los range estimation for nlos environments. *IEEE Transactions on Vehicular Technology*, 53(5):1515–1524, 2004. doi: 10.1109/TVT.2004.832384.

[50] D. Villa, I. Khan, F. Kaltenberger, N. Hedberg, R. S. da Silva, S. Maxenti, L. Bonati, A. Kelkar, C. Dick, E. Baena, J. M. Jornet, T. Melodia, M. Polese, and D. Koutsonikolas. X5g: An open, programmable, multi-vendor, end-to-end, private 5g o-ran testbed with nvidia arc and openairinterface, 2024. URL `https://arxiv.org/abs/2406.15935`.

[51] Y. Zhao, X. Li, Y. Wang, and C.-Z. Xu. Biased constrained hybrid kalman filter for range-based indoor localization. *IEEE Sensors Journal*, 18(4):1647–1655, 2018. doi: 10.1109/JSEN.2017.2768556.

[52] A. Zou, Z. Chen, H. Jiang, L. Xie, and C. Spanos. Accurate indoor localization and tracking using mobile phone inertial sensors, wifi and ibeacon. *IEEE International Symposium on Inertial Sensors and Systems (ISISS)*, pages 1–4, 2017. doi: 10.1109/ISISS.2017.7935650. URL `https://doi.org/10.1109/ISISS.2017.7935650`.

List of Figures

List of Tables

Acknowledgements

I dedicate this work to Enrique B., a writer; to Maria Begoña A., a teacher; and to Beatriz B. and Luan C., the students who guided me to this point.