INSTINCT

D4.1

Specification of Demonstrators

BOSCH

Revisions

Work package	WP4
Task	T4.1
Due date	30-06-2025
Submission date	30-06-2025
Deliverable lead	Bosch
Version	V1.0
Responsible author	Alexander Artemenko (Bosch)
Reviewers	Amélie Hennequart (GW)

Contributing partners

BARKHAUSEN INSTITUT GGMBH (BI)

UNIVERSITY OF PIRAEUS RESEARCH CENTER (UPRC)

ROBERT BOSCH GMBH (Bosch)

AALTO KORKEAKOULUSAATIO (AAL)

FRAUNHOFER HHI (FHG-HHI)

GREENERWAVE (GW)

NEC LABORATORIES EUROPE GMBH (NEC)

INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE (Inria)

INSTITUT NATIONAL DES SCIENCES APPLIQUEES DE LYON (INSAL)

FUNDACIÓ PRIVADA i2CAT, INTERNET I INNOVACIÓ DIGITAL (i2CAT)

OULUN YLIOPISTO (UOulu)

CENTRALESUPELEC (CS)

TELEFONICA INVESTIGACION Y DESARROLLO SA (Telefonica)

Document editing history

Version	Date	Description	Contributor(s)
v0.1	01.02.2025	ToC, first draft	Bosch, UPRC
v0.5	01.05.2025	Main content	All partners
v0.9	06.06.2025	Final contributions	All partners
V1.0	30.06.2025	Final review and revision	GW, all partners

Document information

Nature of the deliverable*:		R
Dissemination level:		
PU	Public, fully open. e.g., website	✓

* Deliverable types:

R - Document, report

DMP – Data Management Plan

DEM – Demonstrator, pilot, prototype

Disclaimer

Funded by the European Union. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or SNS JU. Neither the European Union nor the granting authority can be held responsible for them.

Executive Summary

Deliverable Overview

This deliverable outlines the hardware and software specifications for three hardware (HW) demonstrators and one software (SW) simulation platform. It also presents preliminary test and measurement specifications for the upcoming evaluation campaigns.

The demonstrators include:

- **Hardware Demonstrator I**: *JCAS in a sub-6 GHz network infrastructure*
- **Hardware Demonstrator II**: *RIS-aided JCAS using a mmWave link (28 GHz)*
- **Hardware Demonstrator III**: *UAV-supported 5G & GNSS fusion-based localization framework for enhanced search and rescue (S&R) operations*
- **Software Simulation Platform**

Use Case Alignment

The development of these demonstrators is driven by the use cases defined in Deliverable D1.1. Specifically, they address the following two primary use cases:

1. **Indoor Navigation**: Sensing- and RIS-aided navigation of automated guided vehicles (AGVs) within factory environments.
2. **Outdoor Traffic Management**: Sensing- and RIS-aided traffic control on outdoor campus roads.

Development and Evaluation Framework

The main components of the demonstrators are being developed within WP2 and WP3. These components will be configured to support the demonstration of project objectives and to extract relevant Key Performance Indicators (KPIs), as outlined in Section 1.2.

The demonstrators serve as proof of concept to both showcase the feasibility of the proposed techniques and enable controlled measurements in a laboratory setting. A dedicated link-level simulation tool, incorporating Joint Communication and Sensing (JCAS) and Reconfigurable Intelligent Surfaces (RIS), will be developed. This tool will support KPI analysis in broader scenarios that may not be physically replicable in the lab.

Feedback and Validation

Both measured and simulated KPIs will be fed back into the technical work packages to validate assumptions, refine system models, and enhance algorithm performance. The defined scenarios will include detailed specifications of the required hardware, its physical placement, and the software needed for operation and performance data collection.

A comprehensive test protocol will be created, specifying the measurement procedures and performance metrics. Additionally, the software simulation tool will be designed to support the modeling of key scenarios and the derivation of target KPIs. An initial outline of the planned simulation campaigns will also be included.

Keywords

RIS, ISAC, JCAS, demonstrators, KPI/KVI, hardware, software

Table of Contents

List of Tables

List of Figures

1 Introduction into the Deliverable

With this chapter, we introduce the results presented in this document.

1.1 Scope of the document

WP4 entitled "Demonstrations, Simulations and KPI/KVI Evaluations" focuses on the system integration, testing and experimentation, building upon the achievements of WP1, WP2, and WP3. The use cases defined in WP1, the technology enablers developed in WP2, and the algorithms and network architectures studied in WP3 form the basis for the demonstrators of WP4.

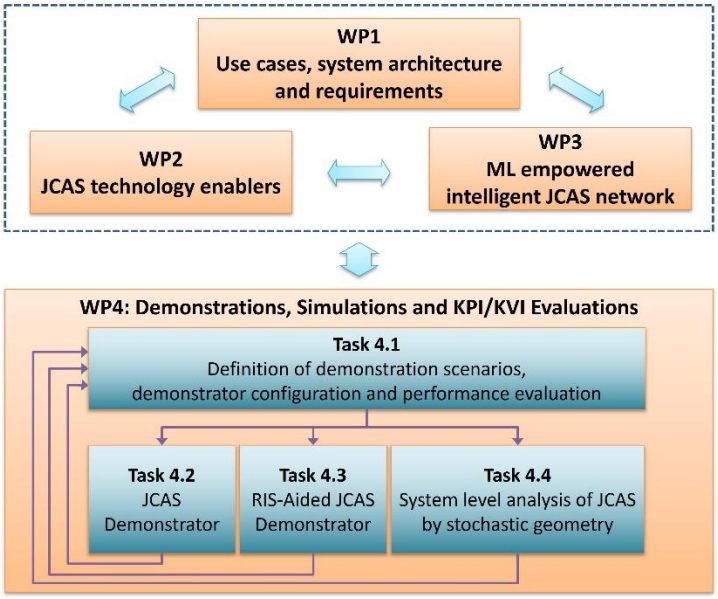

Figure 1: Interfaces between WP1, WP2, WP3, WP4, and interaction among the different tasks of WP4

Figure 1 depicts the interfaces between the work packages WP1, WP2, WP3 and WP4, as well as how the individual tasks within WP4 interact, with the aim to design the three hardware and one software demonstrators.

The individual tasks are structured to serve the specific objectives of WP4, which are summarized below:

1. To define demonstration scenarios and specifications addressing the three INSTINCT pillars
2. To build three hardware proofs of concept to demonstrate the pillars and project objectives in laboratory and even real-world environments
3. To build a software simulation platform to extend the HW demonstrations to more complex scenarios
4. To evaluate the relevant KPIs for different scenarios

In preparation towards this deliverable, all partners came up with an overview of partner individual wishes and plans for WP4 work. The following input has been collected:

1. Main technological advances to be demonstrated
2. Main KPIs to be demonstrated
3. Use cases to be demonstrated
4. HW/SW demo
5. Tools to be used
6. Current state of work (basis for the work)
7. System requirements
8. Details on the setup (e.g., HW/SW components, tools used, etc.)

1.2 Addressing Three Pillars and Project Objectives

INSTINCT aims to bring together advanced radar and sensing technologies, breakthrough intelligent material science and combined sensing and artificial (cyber-physical) intelligence into a unified framework and a coherent multifunction system concept leveraging novel key technologies, system concepts, and network architectures such as:

- Intelligent surfaces and algorithm design for co-design of sensing and communications
- RIS-as-a-sensor and network-as-a-sensor
- Unified intelligence framework combining co-designed sensing and communications and ML-based approaches

INSTINCT will bring together these novel technologies and concepts into a unified framework addressing connectivity *beyond just the transmission of bits*. This innovative wireless communication architecture will be demonstrated through hardware and software proof-of-concept demonstrations, which will showcase selected scenarios that reflect the three pillars of INSTINCT:

Pillar I: The objective of **sense-to-communicate** is to take advantage of the communication aspect, to transform the wireless network into a smart "*sense-the-world*" platform, providing a wireless radio capable of sensing, detecting, mapping, and understanding semantics. The topics, which address this pilar and will be approached by the partners, are listed below:

1. Hierarchical beam-tracking and ranging (Estimate the user location with controllable accuracy and low overhead)
2. The hardware/software platform provided by BI supports the development of a path towards integrating sensing services into existing communication infrastructure, specifically network base stations. It models a single communication link consisting of a spatially separated full-duplex MIMO base station and a MIMO terminal, both nodes being represented by a software defined radio. Researchers are enabled to rapidly prototype and evaluate digital signal processing algorithms for sensing, communications and their various levels of integration via the platform's native Python interface in either a physical hardware testbed or in full-scale simulation campaigns deployable to high-performance computing clusters.

3. Propose a beam tracking and localization approach using RIS through a feedback link operating at low frequency. In particular, the UE can report its measured RSRP at mmWave and report the information at low frequency to the RIS. The RIS controller would adjust its beam direction and capabilities according to this information.
4. Work on developing an ML-based radio coverage model that is able to model radio propagation in 3D environments. In particular, the model will *sense* the 3D geometric environment and will extract contextual information that will later be used to estimate the radio signal propagation through a real-world scene.
5. Stochastic Geometry evaluation of what sensing can bring to communication. This is in particular addressed in the simulation tool of Task 4.4 where the prediction of obstructions by the sensing of moving object is used to improve communication rate.
6. Providing and adapting the CorteXlab testbed making it more useful and easier to use for all partners.
7. The ray tracer tool under development in UOulu is planned to be used in Task 4.4 to cooperate in the statistical geometry study. The detailed plan for this cooperation will be finalized in later phase of the project.

Pillar II: The objective of **communicate-to-sense** is to take advantage of sensing information, to substantially improve radio spectrum usage and support novel beyond-communications use cases. The topics, which address this pilar and will be approached by the partners, are listed below:

1. Opportunistic ISAC with RIS (Extract location information through an AI architecture w/o interfering with communication operation at the RIS)
   Integrating sensing into existing networks presents challenges, often requiring additional hardware, protocol changes, or dedicated sensing time slots. Hence, it introduces complexity and potential reduction to communication performance. To overcome such limitations, we propose an opportunistic ISAC approach that opportunistically infers user locations from the configuration of RIS operating solely for communication purposes.
2. Dimensionality reduction for Sensing and RIS control
3. AI-based Optimization of 5GNSS for Robust 5G-NR and GNSS Localization
4. Deployment and operation strategies
5. High-resolution sensing through wideband transceiver hardware and single-carrier communication schemes
6. Providing and adapting the CorteXlab testbed making it more useful and easier to use for all partners.

Pillar III: The objective of **multi-functional JCAS network intelligence** is to leverage AI-based optimization of wireless ISAC systems and algorithm design, together with data-driven ML approaches, to optimize the architecture, resources allocation, propagation modeling and waveform design. The topics, which address this pilar and will be approached by the partners, are listed below:

1. Stochastic geometry modeling and performance evaluation. The activity was focused on two main items, the first of which is complete, while the second is in progress. *As the first*

activity, we evaluated the downlink performance of cellular networks in terms of coverage and electromagnetic field exposure (EMFE), in the framework of stochastic geometry. The model is constructed based on datasets for sub-6 GHz macro cellular networks but it is general enough to be applicable to millimeter-wave networks as well. On the one hand, performance metrics are calculated for β-Ginibre point processes which are shown to faithfully model a large number of motion-invariant networks. On the other hand, performance metrics are derived for inhomogeneous Poisson point processes with a radial intensity measure, which are shown to be a good approximation for motion-variant networks. For both cases, joint and marginal distributions of the EMFE and the coverage, and the first moments of the EMFE are provided and validated by Monte Carlo simulations using realistic sets of parameters from two sub-6 GHz macro urban cellular networks, i.e., 5G NR 2100 (Paris, France) and LTE 1800 (Brussels, Belgium) datasets. In addition, we analyzed the impact of network parameters and discussed the achievable trade-off between coverage and EMFE. *As for the second activity*, we aim to examine the number of communication modes, that is, the degrees of freedom (DoF), in a wireless setup comprising a small continuous linear intelligent antenna array in the near field of a large one. The framework allows for any orientations between the arrays and any positions in a two-dimensional space assuming that the transmitting array is placed at the origin. Therefore, apart from the length of the two continuous arrays, four key parameters determine the DoF and are hence considered in the analysis: the Cartesian coordinates of the center of the receiving array and two angles that model the rotation of each array around its center. The work starts with the calculation of the deterministic DoF for a generic geometric setting, which extends beyond the widely studied paraxial case. Subsequently, a stochastic geometry framework is proposed to study the statistical DoF, as a first step towards the investigation of the system-level performance in near field networks. Initial results applied to millimeter wave networks reveal the large number of DoF provided by near-field communications and unveiled key system-level insights.

2. Co-design and joint optimization of radar sensing and communications waveforms. Allocation of frequency resources (multicarrier waveforms) and spatial resources (multiantenna systems) to different communications and sensing tasks.
3. Ray-traycing-tool can be used to model both indoor and outdoor environments. However, in INSTINCT the emphasis is on the cellular environment.

1.3 Driving Use Cases that Motivate Demonstrations

Concluding the first chapter, we would like to address one of the main outcomes from [1]: two use cases, which have been selected in relevance to the three pillars of INSTINCT. The demonstrators will be developed to serve the following two “driving” use cases of the project:

1. Use case 1: (Indoor) Sensing- and RIS-aided navigation of AGVs in factory environments
2. Use case 2: (Outdoor) Sensing- and RIS-aided Traffic Control in Outdoor Campus Roads

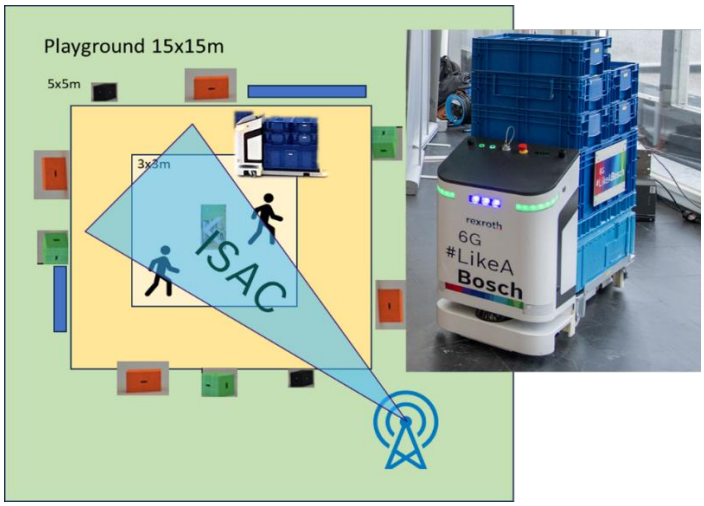

Figure 2: Use case 1: (Indoor) Sensing- and RIS-aided navigation of AGVs in factory environment

These use cases are depicted in Figure 2 and Figure 3. They motivate our work in WP4 as described in this document. Further details on the use cases can be found in [1].

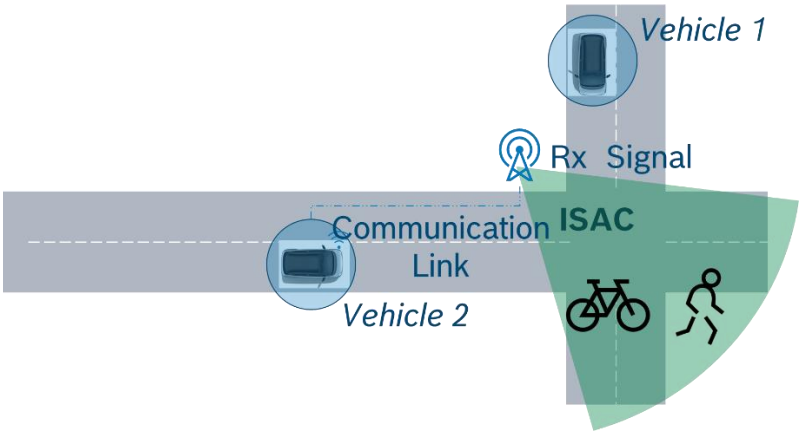

Figure 3: Use case 2: (Outdoor) Sensing- and RIS-aided Traffic Control in Outdoor Campus Roads

1.4 Structure of the Document

The objective of deliverable D4.1 is to provide details about the four proof-of-concept demonstrators that are planned to be implemented in INSTINCT. The organization of the remainder of this document is as follows:

1. Section 2: Hardware Demonstrator I "JCAS in a sub-6GHz network infrastructure"
2. Section 3: Hardware Demonstrator II: "RIS-aided JCAS using mmWave link (28 GHz)"
3. Section 4: Hardware Demonstrator III "UAV-supported 5G&GNSS fusion-based localization framework for enhanced S&R operations"
4. Section 5: Software Simulation Platform

Each section describes the implementation details of the corresponding demonstrator and presents the scenarios that will be demonstrated based on the developed software and hardware.

2 Definition of Hardware Demonstrator I "JCAS in a sub-6GHz network infrastructure"

The demonstration for joint communication and sensing in sub-6GHz network infrastructure is based on two complementary hardware platforms provided by Insa-Lyon/INRIA and Barkhausen Institut showcasing the integration of sensing functionality into existing communication infrastructure on both a macroscopic scale, featuring up to 40 individual communication nodes, and a microscopic scale, consisting of a single communication link between two nodes. It is designed to demonstrate the pillars and project objectives in both controlled laboratory and real-world environments.

2.1 Demonstrator Description

CorteXlab is hosted by INRIA in Lyon, France, whereas the HermesPy JCAS Testbed is hosted by Barkhausen Institut in Dresden, Germany. The following table provides a comparison of key indicators:

Table 1: Key indicators of the first HW demo

	CorteXLab	JCAS Testbed
Location	Lyon, France	Dresden, Germany
Number of network nodes	40	2 - 4
Number of streams per node	1x1, 2x2, 4x4	4x4
Maximal bandwidth	160 MHz	400 MHz
Carrier frequency range	10 MHz – 6 GHz	1 MHz – 8 GHz
Remote access	Yes	No
Intended Ranges	≤ 20m	30cm – 10m

2.1.1 Description of CorteXlab

Hosted at the telecommunications department of INSA-Lyon and managed by INSA-Lyon and INRIA, the Cognitive radio testbed and eXperimentation lab (CorteXlab) facility, was inaugurated in 2014, created to foster experimentation for future multi-node radio systems. Its main characteristic is the installation itself, comprised of an electromagnetically isolated experimentation room, of roughly 180 m^2 (see the Figures 4 and 5). The experimentation room hosts 40 software defined radio (SDR) nodes, each paired with a computer, able to perform (near)-real time signal processing for SDR radio transceivers in code. The electromagnetic isolation of the experimentation room creates a controlled propagation environment which can be exploited by experimenters. It also guarantees that, neither radio signals produced inside the experimentation room can create interference to outside devices, nor that radio signals from outside the experimentation room can interfere with experiments running inside. These

important features allow experimentation in any frequency band without concern to restricted bands, as well as reproducible experiments, a critical characteristic for proper scientific advancement in the field. It is important to highlight that CorteXlab is one of the few radio testbeds in the world with these characteristics, at this scale.

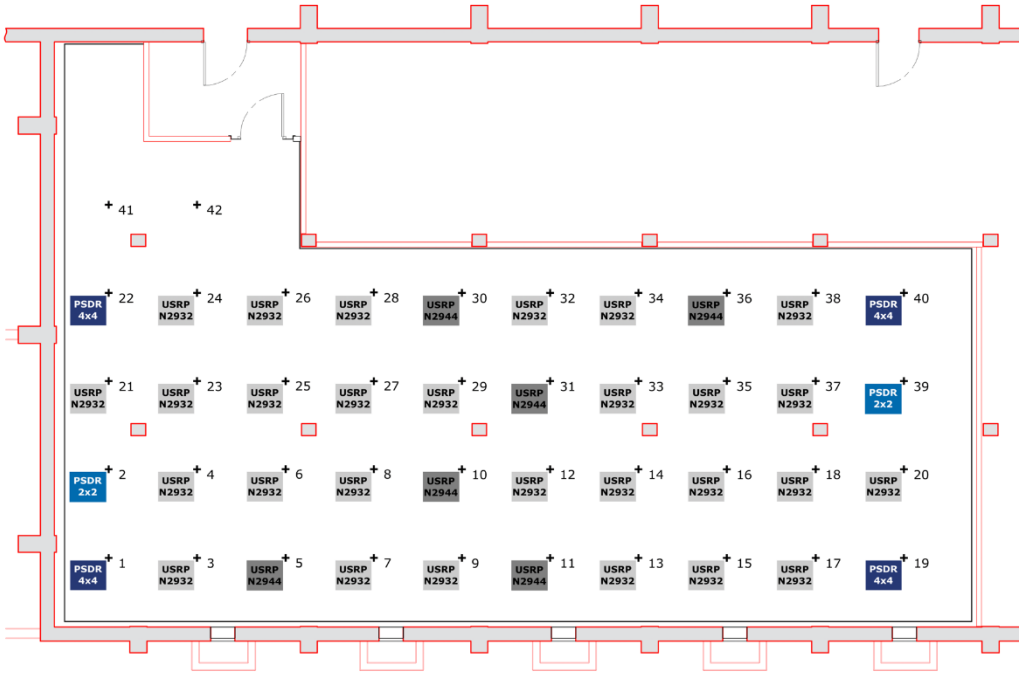

Figure 4: CorteXlab nodes mapping

Figure 5: CorteXlab inside view

CorteXlab counts with:

1. 28 USRPs N2932 - Frequency range: 400 MHz to 4.4 GHz – BW: up to 20 MHz – Single Input Single Output
2. 6 USRPs N2944R - Frequency range: 10 MHz to 6 GHz – BW: up to 160 MHz – 2x2 Multiple Input Multiple Output, phase synchronous

3. 6 PicoSDRs - Frequency range: 300 MHz to 3.8 GHz – BW: up to 20 MHz – 2x2 and 4x4 Multiple Input Multiple Output
4. 2 Turtlebots with one USRP B210 each for experiments requiring mobility - Frequency range: 10 MHz to 6 GHz – BW: up to 56 MHz – 2x2 Multiple Input Multiple Output
5. 5 Octoclocks (1 master, 4 slaves) for precise timing and alignment of oscillator frequencies (only for fixed USRP devices), PPS and 10 MHz reference wiring as seen in Figure 6.
6. 1 controllable Greenerwave's FR1 3,4-3,8GHz RIS device
7. Various support and measurement equipment: DC-6GHz Agilent portable spectrum analyzer, 2 TinySA ULTRA+ DC-6GHz portable spectrum analyzers (remotely controllable), SigFOX base station, localization equipment for precise location of the Turtlebots, etc.

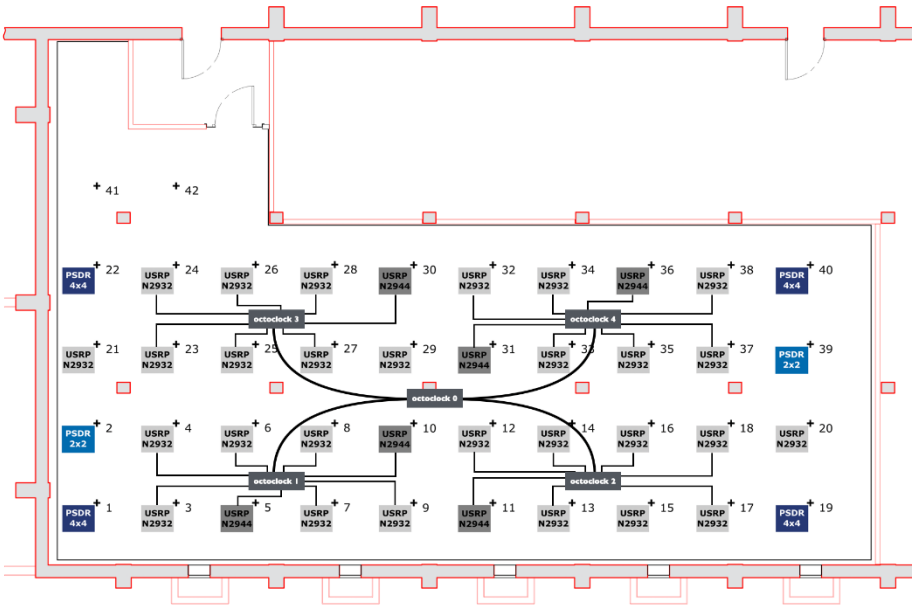

Figure 6: CorteXlab networking configuration example

What makes CorteXlab stand out when compared to similar testbeds in the world is its middleware that controls all aspects of the experiment. Indeed, while it is easy to control the experiment over a few SDRs, scaling up to tens of SDRs can be logistically challenging. CorteXlab can be used remotely over the internet through its internal control software stack, that manages reservation, scheduling, turning on (and off) nodes, experiment deployment in the selected nodes and results recovery, all automatically, based on an experimental scenario description.

Finally, the SDR nodes (through their associated computers) can be programmed using any software that can run in a Docker container under Linux, and that uses the UHD driver as interface. Being a physical layer centric testbed, the SDR toolkit of choice is GNU Radio and containers for several versions up to 3.10 are available under Dockerhub, but it can also be used with other SDR software stacks like OpenAir Interface or BI's HermesPy, for example.

2.1.2 HermesPy JCAS Testbed

BI's joint communication and sensing testbed is built around the Heterogeneous Radio Mobile Simulator Python (HermesPy), a physical layer Python framework designed to accelerate the

early-stage research and development cycles of novel digital signal processing algorithms for electromagnetic sensing and communication applications. Any DSP algorithm integrated into HermesPy's framework can easily be deployed to either hardware testbeds consisting of multiple software defined radio MIMO nodes or Monte-Carlo style simulation campaigns considering detailed channel and radio-frequency hardware models.

In this specific setup, two Ettus x410 USRPs are connected to a respective array of horn antennas and configured to operate at a carrier frequency of 6 GHz with a bandwidth of ~ 400 MHz. With the two arrays facing each other, any object placed within the general area between the two arrays will result in backscattering of electromagnetic radiation towards receiving antennas when illuminated by transmitted antennas, effectively creating a barebone radar channel (see Figure 7).

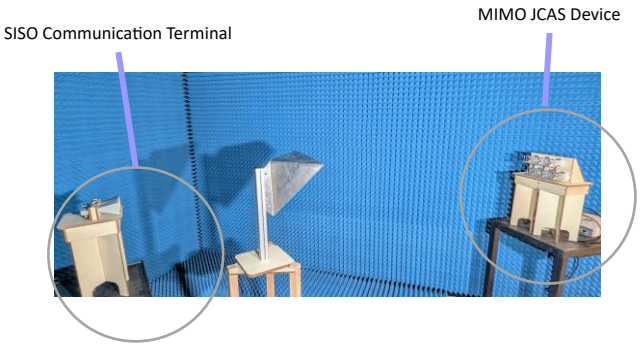

Figure 7: JCAS MIMO Testbed

The testbed can be extended to up to 4 dedicated USRPs time- and frequency-synchronized by an Octoclock.

2.1.3 FR1 RIS by Greenerwave

In the framework of the project, GW has developed a RIS in the band FR1 intended to be used by the partners and deployed in the testbeds for both cases "Communicate to sense" and "Sense to communicate". In the sequel, we provide a description of the developed RIS and its main characteristics.

2.2 Implementation Details

This section provides the details for the implementation of the FR1 demonstrator.

2.2.1 CorteXlab

Being a support platform for experimentation, CorteXlab itself will not introduce any JCAS in the sub - 6 GHz band (or above for that matter). However, a tutorial that introduces the main blocks that could be used by partners wishing to use CorteXlab for their own experiments was put in place to speed up their development. This tutorial was made available in the CorteXlab wiki page as well as introduced to all partners during the general meeting in Lyon in May 2025.

The demonstrator provides a very simple set-up where the CorteXlab experiment room is divided into two non directly accessible sections. In the interface of these two sections a Greenerwave

RIS device in the sub-6 GHz band was installed and can be used to reflect radio signals from one section of the room to the other. Furthermore, two USRPs are used, one as a transmitter, sending pilot signals, and another as a receiver, estimating channel coefficients. The main idea is to show that channel coefficients do indeed change significantly when the RIS device provides a meaningful reflection from the transmitting device to the receiving one.

Figure 8: Partition wall inside CorteXlab

2.2.2 HermesPy JCAS Testbed

Properly configured, investigators interested in evaluating the performance of communication and sensing waveforms and algorithms can exploit HermesPy's framework to jointly run the same Python code over the hardware testbed and within various simulation campaigns. The demonstrator is able to jointly collect communication performance indicators such as modulation error vector magnitudes and bit-, block-, or frame error rates of the communication link between the two displayed devices, as well as sensing performance indicators such as radar operation characteristics, probabilities of detection and false alarm, and receiver operating

characteristics and store collected measurement data and performance metrics in the HDF5 file format (see Figure 9).

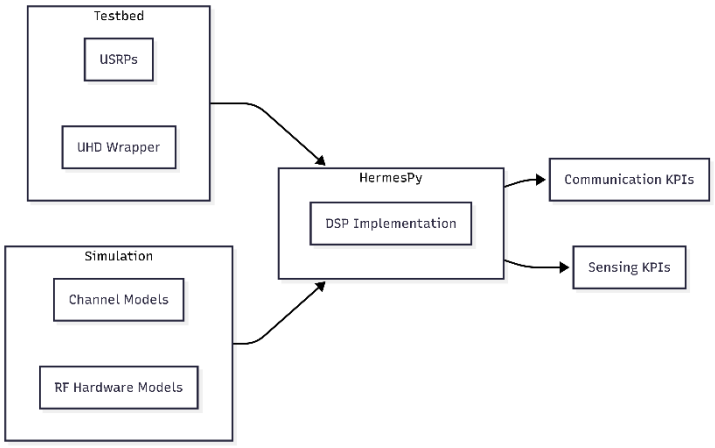

Figure 9: Work protocol of HermesPy

2.2.3 FR1 RIS

The FR1 RIS consists of 64 elements, each with 2 switches to reflect each incoming polarization respectively. Shown in Figures 10,11 and 12, the RIS is made of a bottom metasurface, containing the pixels capable of intelligent reflection, as well as a top metasurface which interacts with the bottom and filters the undesired waves.

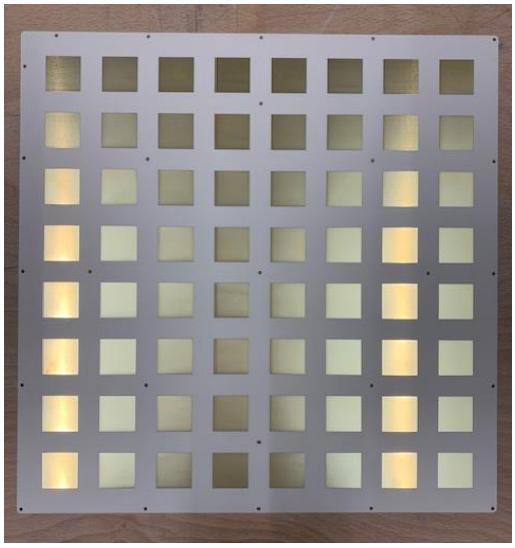

Figure 10: Top of GW RIS FR1

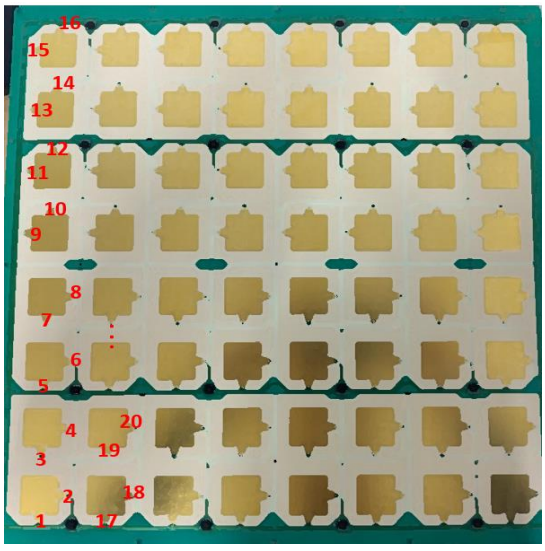

Figure 11: Bottom of GW RIS FR1

The control board of the RIS, responsible for configuring the RIS by turning switches to a 0 or a 1 state, is shown in below figure.

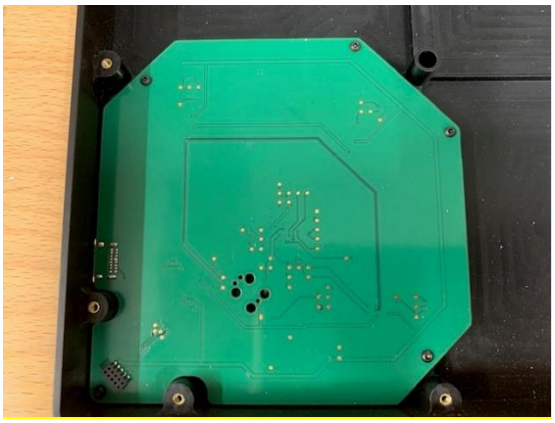

Figure 12: Control Board of the RIS FR1

The RIS works specifically at 3.4-3.8 GHz, tailored to be integrated in sub 6 GHz networks. Due to the utilization of switches to reconfigure the surface, the RIS FR1 consumes extremely minimal power. No additional power supply is required. It can be supplied directly by the computing processor used to control the RIS.

Illustrated in Figure 13, the following setup was used to perform early measurements to validate successful optimization of the RIS. The TX horn antenna was positioned on the left almost at boresight from the RIS, and the RX horn was placed at a 32° in azimuth and −6° in elevation. The RIS, placed on the right wall, is controlled by a standard laptop. The Pixel By Pixel (PBP) optimization algorithm is run on this laptop and requires the S21 measured from the VNA to find the best configuration possible for the RIS to correctly reflect the incident signal and beamform towards the RX antenna.

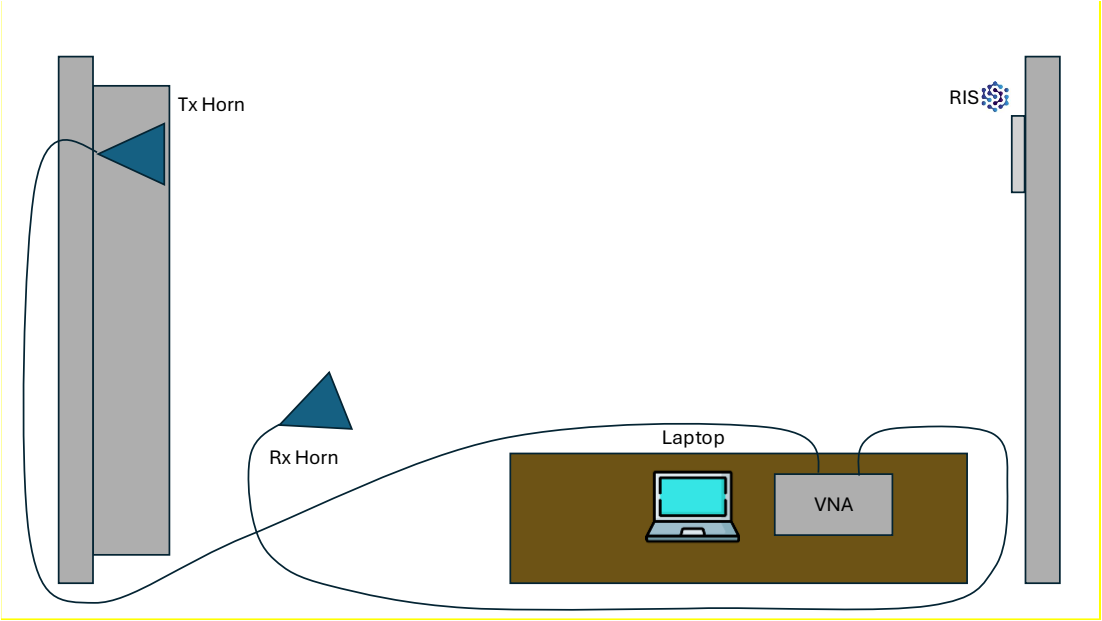

Figure 13: Bird's Eye View of the Measurements Setup

During these measurements, the RIS offered a clear improvement in gain, compared to the same environment without the RIS.

Figure 14 exemplifies this by comparing the S21 in the 3.7-3.9 GHz band with the optimized RIS, the RIS with all states equal to 1 (or ON), and the RIS with all states equal to 0 (or OFF). We can observe a gain in S21 of at least 6 dB when the RIS is configured at optimization, which is considerable in FR1. In a best-case scenario at around 3.85 GHz, we observe a clear S21 improvement of almost 16 dB. This shows the clear utility in different use-cases for the RIS FR1, encouraging its integration and utilization in networks.

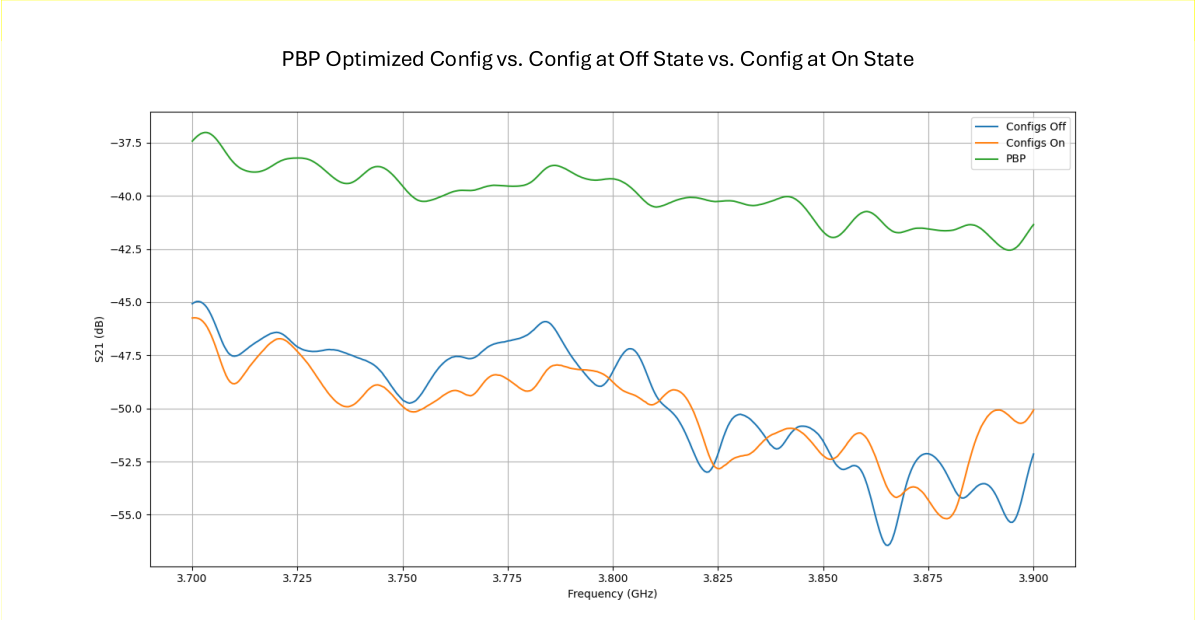

Figure 14: Comparison of S21 between optimized RIS and RIS with states at all off and all on

In order to work with the RIS, a user guide is provided showing how to install it and also presenting the pixels indexing mapping. In custom configuration mode, user can send an array of states (1:On, 2:Off) for each cell's switchs. The array will be applied to the RIS Metasurface in column by column starting from the bottom to top, left to right. This mapping can be seen on Figure 11.

In addition, a Python API can be used to control the RIS and to run the optimization algorithm which results are described just above. For this, a feedback system (VNA for instance) is required.

2.3 Testbed Implementation Plans

For the CorteXlab tutorial described in Section 2.2.1, several modifications were carried on in the experiment room to support the kinds of scenarios necessary for sub-6GHz JCAS experiments by the partners. These changes were made based on the expected characteristics required by the partners but are not ultimate or rigid. As soon as the other partner's experimental scenarios become clearer, CorteXlab can be further adapted to fulfill these new features.

- New isolating partitioning walls (3 piece, each 1.20m x 1.75m) were made from a conductive material (electromagnetic isolation) and covered with absorbent foam to minimize excessive reflections
- A Greenerwave RIS panel (sub-6 GHz, 40 cm x 40 cm x 4 cm) was installed
- A ready-made tutorial experiment job that can be modified by partners.

2.4 Test and Demonstration Scenarios

The CorteXlab demonstration constitutes a separation of the CorteXlab experiment room into two non-directly accessible sections. Orthogonal to the interface of these two sections, a Greenerwave RIS panel in the sub-6 GHz band can be activated to reflect radio signals from one section of the room to the other. Two USRPs, each in one section of the room act as 1) a transmitter, sending continuously pilot signals, and 2) a channel sounder, estimating channel coefficients and storing them to a file.

- **RIS channel improvement in none-line-of-sight communication scenarios**
  Due to the placement of the reflective intelligent surface at the edge of a wall separating two halves of CorteXlab, the link improvement of a properly configured RIS can be evaluated and demonstrated by selecting a pair of nodes separated by the wall and refining the RIS element weights.

- **RIS in sensing scenarios**
  Both monostatic and bistatic sensing might benefit from a properly configured RIS. The CorteXlab setup, including its autonomous robot platforms, can be deployed to investigate potential benefits.

- **MIMO sensing by communication infrastructure**
  Existing MIMO communication infrastructure may be reconfigurable to add sensing functionalities to communication signals. Within both the CorteXlab and BI MIMO setups, traditional communication-centric waveforms such as OFDM, OTFS or OCDM can be investigated in terms of their monostatic sensing capabilities. CorteXlab's autonomous robot platform can be deployed as a potential sensing target in both monostatic and bistatic configurations. Barkhausen Institut's MIMO testbed contains multiple corner reflectors and spheres as potential sensing targets that can be added to the existing link model.

- **RIS in physical layer secret key generation**
  With the emergence of quantum processing, traditional public-private key authentication methods governed by a central certification authority are under imminent threat of obsolescence. A promising solution to enable wirelessly linked IoT devices to exchange a quantum-secure encryption key and thus establish a safe connection without the need for third-party involvement is deriving said key from the unique frequency selectivity of the linking channel. However, in static scenarios with strong line-of-sight connection, the channel might evolve very slowly, leading to a suboptimal key generation rate. RIS might be deployed in such a static scenario to introduce a dynamic random element and improve the key generation rates. CorteXlab is a perfect example of such a static environment and can be used to investigate secret key rate algorithms in combination with RIS.

3 Definition of Hardware Demonstrator II "RIS-aided JCAS using mmWave link (28 GHz)"

This part of the document describes our second demonstrator.

3.1 Demonstrator Description

Here, we give an overview of the scope and general concept of the demonstrator.

3.1.1 Demonstrator concept of the FR2 sensing setup

The second hardware demonstrator explores technologies and characteristics slated for implementation in the next generation(s) of wireless networks. This includes operation in the mmWave band, with a carrier frequency set at 28 GHz, as well as the inclusion of a state-of-the-art Reconfigurable Intelligent Surface (RIS) as a secondary JCAS node. Full-fledged, beamforming-capable JCAS transceivers act as fixed base station (BS) and mobile user equipment (UE) communicating with each other and sensing the environment, while the RIS augments their capabilities by providing additional sensing data and extending or redirecting the communication link and thereby increasing the robustness of the network. The scope of the demonstrator is limited to a single point-to-point (P2P) communication link and monostatic or bistatic sensing, depending on which of the demonstration scenarios, which are detailed below, is chosen. This allows the consortium to optimize the performance of JCAS functionality on the physical link level first and proceed from there.

3.1.2 AI-based opportunistic ISAC with RIS

Our demonstrator is planned to include a RIS device capable of dynamically manipulating electromagnetic waves through programmable configurations. The aim is to validate the opportunistic localization strategy described in [2], which leverages RIS configurations intended for communication purposes only, to perform localization without disrupting ongoing communication operations.

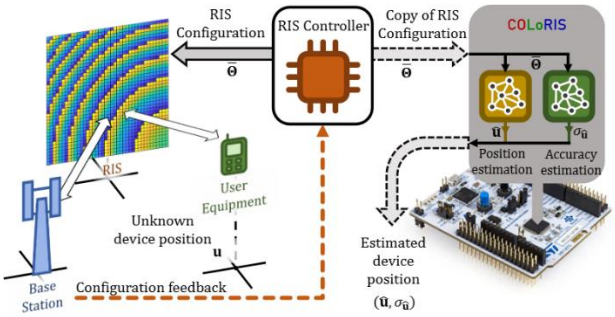

Figure 15: Workflow of the proposed opportunistic ISAC approach (dubbed as COLoRIS)

To this end, the demonstration will require a RIS, an access point or BS, and a mobile user. The RIS will enhance the communication link between the access point (or BS) and the mobile user. A configuration controller will select and apply the optimal RIS configuration. Once the RIS is configured, its current configuration status will be collected and processed by a separate entity

running the opportunistic ISAC framework, which performs localization based on this information. The system architecture will resemble the one depicted in Figure 15.

Importantly, our design assumes no access to the internal processing of the RIS. Instead, we focus on extracting useful sensing information from the externally observable configurations used during regular communication tasks.

3.1.3 FR2 RIS by Greenerwave

GW provides in the project RIS operating at mmWave (26 GHz-29 GHz).

It consists of 1600 elements or pixels, each with 2 diodes which allows independent polarization control of the incident wave. RIS FR2 is composed of 4 GW metasurfaces each with 800 total diodes, resulting in a total of 3200 diodes (see Figure 16,17).

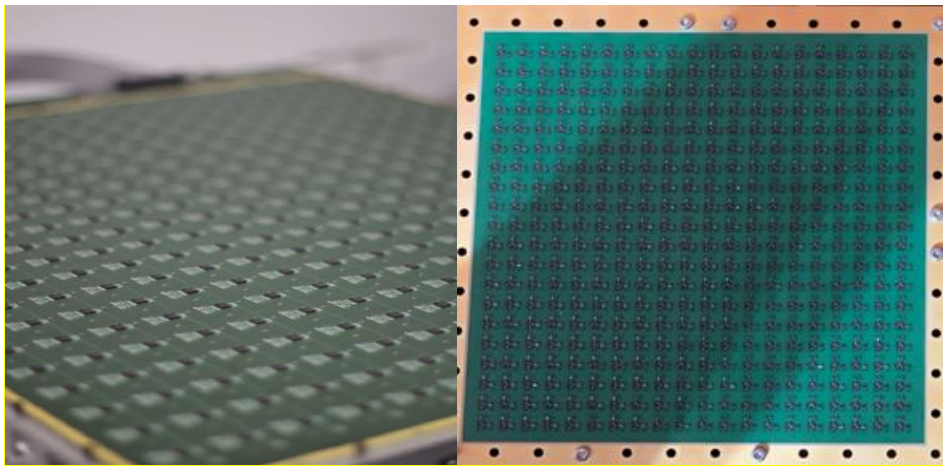

Figure 16: FR2 RIS metasurface by Greenerwave

Figure 17: GW FR2 RIS mounted on a wall

The RIS is reconfigured by the control board which controls the diodes to redirect the signal towards the Rx user equipment (UE) antenna. The control board is placed inside the RIS casing and necessitates a power supply, as well as a microUSB to connect to the laptop responsible for running the beamforming algorithms.

3.2 Implementation Details

Here we give details about the hardware and software specifications.

3.2.1 Hardware specifications (FhG-HHI)

Novel transceiver hardware forms the technical backbone of the demonstrator. The BS and UE nodes are each represented by a 4x4 mmWave Phased Array Antenna Module (PAAM) operating at 28 GHz carrier frequency and offering 2 GHz of impedance bandwidth. The array geometry allows for two-dimensional beamforming of the transmitted and received signal with full phase and amplitude control per antenna element. A complementary RIS provided by GW with more array elements, but more coarse control over each reflecting element's phase acts as an additional node in the JCAs network. Wideband DAC and ADC equipment provided by HHI ensures reconfigurability and flexibility of the setup, because the physical link is agnostic to waveform, modulation format, beamforming codebook etc. With minimal hardware reorganization, many different software solutions can be employed and tested.

Finally, a collaborative robotic arm (Cobot) introduces dynamic elements into the setup when needed. The small, lightweight transceivers can be attached to the tool flange of the Cobot to simulate a mobile UE scenario, a blockage can be maneuvered along a path to partially or fully occlude the line-of-sight (LOS) path between transmitter (Tx) and Receiver (Rx), or reflective surfaces can be positioned to arrange a more or less detrimental multi-path propagation scenario. The concrete configuration of these elements depends on the explored use case.

3.2.2 AI-based opportunistic ISAC with RIS (NEC)

The objective is to infer the user's position solely through the analysis of RIS configurations applied during standard communication operations. To achieve this, we require three key elements: the known positions of the transmitter and receiver, the RIS configurations used in each interaction, and the corresponding received power measurements or other communication-related KPIs. In the intended setup, the positions of both the access point or BS and the RIS are fixed, while a mobile user moves within the service area. The RIS operates continuously to enhance communication performance between the access point and the mobile user.

Received power measurements are used to guide the selection of RIS configurations (i.e., to optimize the RIS). At the same time, the mobile user's position, the received power, and the active RIS configuration are recorded and stored. This process is repeated for various positions of the mobile user within the area, gradually building a dataset. This dataset is then used to train and evaluate an AI agent responsible for performing opportunistic localization.

3.2.3 Tracking RIS-based demonstrator

GW RIS FR2 is intended to be integrated in FhG-HHI FR2 setup in Q3 2025 and in Arena2036 from Bosch in Stuttgart in Q4 2025 for indoor tests in real environment. This is considered as a toolset that can be used by the partners to run scenarios described in paragraph 3.3.2.

Monthly meetings are in place among partners involved in this FR2 test setups in order to precisely define scenarios as well as all equipment needed to showcase them.

3.3 Testbed Implementation Plans

Here, we give details on the implementation of the demo.

3.3.1 Two-stage approach

The implementation of the demonstrator takes place in several phases, encompassing two scopes of deployment. A downscaled, i.e. miniaturized version is first set up in the FhG-HHI lab under controlled conditions. This iteration of the testbed is housed in a rectangular 2000 mm × 1200 mm × 800 mm radio cell constructed from low-reflection cage profiles [Figure 18]. The vertical orientation of the setup reflects the geometry of short- to medium-range urban environments, where small-area radio cells are serviced by a BS located at a vantage point.

For further testing, the setup is subsequently relocated to Bosch facilities in Stuttgart. Field tests can be performed on a bigger scale and in realistic, application-centric environments.

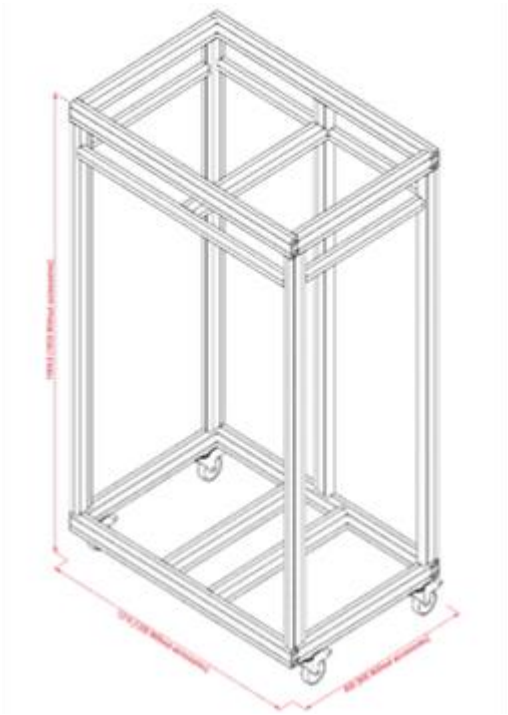

Figure 18: Isometric rendering of the radio cell

3.3.2 AI-based opportunistic ISAC with RIS

The testbed will feature a RIS installed in a realistic indoor environment, such as the CorteXlab, or outdoor. The RIS controller will operate independently, continuously optimizing configurations to support communication as users move within the area. The system will log the selected configurations and build a dataset used to train the opportunistic ISAC agent. This same dataset can also serve for evaluation purposes. Moreover, a previously trained agent may be deployed to perform real-time user localization based on live RIS configurations, demonstrating the feasibility and effectiveness of the approach in a dynamic and operational environment.

3.4 Test and Demonstration Scenarios

Here, we list the various scenarios, in which the demonstrator will be tested. The detailed protocols for the testing and measurement campaigns will be presented in the next deliverables of WP4.

While an extensive discussion in the consortium has produced a wide spectrum of envisioned use cases – more than could feasibly be explored over the course of this project – categorizing, clustering, and simplifying them revealed the major differentiator between them, namely whether they presuppose and indoor or outdoor environment. Deploying the demonstrator in one representative scenario for each environment type allows for the extensive study of associated phenomena and behaviors with minimal redundancy and on a realistic time scale.

3.4.1 Indoor testing

An **indoor** scenario in a factory-of-the-future setting in Bosch's Arena2036 covers the first set of use cases, where ambient conditions, such as temperature and humidity, are assumed to be

relatively static. However, sources of reflection are more common and multi-path propagation is more pronounced due to the spatially segmented nature of indoor spaces, employed building materials, as well as dynamic elements, e.g. foot traffic and small autonomous ground vehicles (AGVs). The environs demand a high sensing resolution and degree of accuracy from the JCAS system, as well as separating clutter from signals-of-interest will be a major challenge in an electromagnetically busy space such as a factory.

As mentioned above, our first testing campaign is planned to be performed in Arena2036 (see Figure 19). As a starting point, there are three scenarios to be tested first: (a) direct sensing of targets, (b) indirect sensing of targets facilitated by RIS, (c) combined sensing including direct and indirect sensing facilitated by RIS. All three scenarios are depicted in Figure 20. As targets, we consider usage of a autonomous transport system ACTIVEShuttle created by Bosch Rexroth and adapted to the testing purposes by the project partner Bosch. Additionally to the autonomously moving platform, further static (bulky objects like metallic shelves) and dynamic (persons/workers) targets will be added to the sensing scene. The ground truth will be provided by the on-board lidar sensor of the transport system, which is used to do the self-localization of the moving platform in the environment with an accuracy of about 1 cm.

Figure 19: Arena2036 indoor environment representing industrial manufacturing scenario

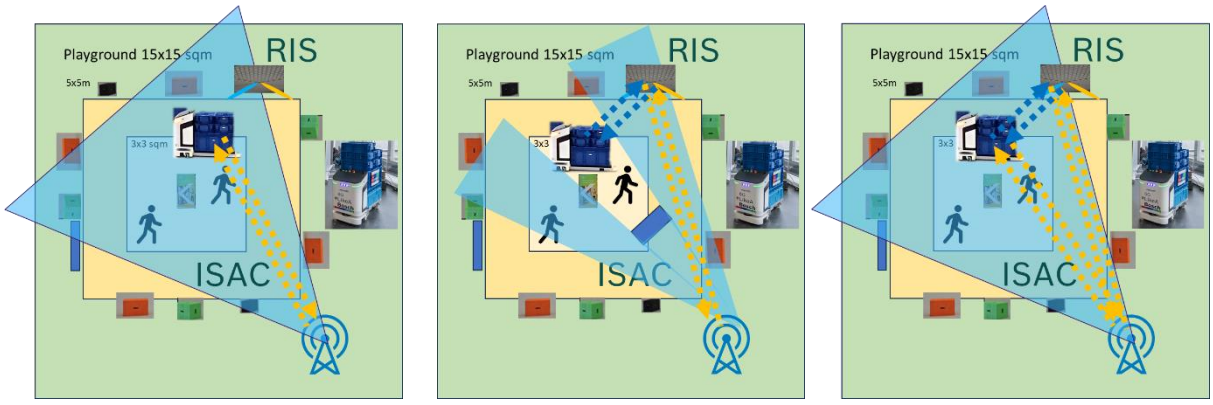

Figure 20: Three testing scenarios for indoor environment
(left) direct sensing of targets, (middle) indirect sensing of targets facilitated by RIS, (right) combined sensing including direct and indirect sensing facilitated by RIS

3.4.2 Outdoor testing

Outdoor scenarios are represented by a campus area in another one of Bosch's locations. The area is characterized by intersecting roads for car traffic, little to no vertical spatial segmentation, but a more expansive horizontal scale, as well as unmitigated exposure to weather effects. Optimizing the link budget is a prerequisite for covering the larger distances, as is coping with the requirements for high sensing accuracy at increased distances and optimizing sensing rates and adaptation speed enough to be able to handle cars travelling at typical speeds.

The testing area is depicted in two sub-plots in Figure 21. Main targets will be represented by real test vehicles as well as pedestrians and cyclists. The ground truth will be provided by lidar and radar sensors installed close to the intersection. The exact moving scenarios for the outdoor testing will be discussed after the first results from the indoor testing become available.

Figure 21: Testing area in the campus of Bosch in Renningen representing the outdoor scenario

3.4.3 RIS-as-a-sensor

The concept of RIS-as-a-sensor is based on the idea of leveraging the communication between the RIS and the receiver (Rx) in order to locate the Rx, using solely information about the received power at the Rx, i.e. without the need to process the back-scattered wave [3]. Using the RIS to steer the incident wave towards prescribed directions, the incident wave can sense the angle of the Rx with respect to the RIS. The RIS can subsequently reshape the incident beam into a focused beam with controllable focal distance, to identify the distance of the Rx from the RIS. With knowledge of both the Rx-RIS angle and the distance, the location of the Rx can be fully retrieved. The formation of the respective beams is depicted in Figure 22 (a).

From an algorithmic perspective, the procedure is divided into two phases that follow a hierarchical and binary tree structure [3], as depicted in Figure 22 (b). In phase 1, beam tracking is performed with RIS-assisted beam forming, in order to estimate the Rx's direction with respect to the RIS. At the n^{th} hierarchical level of phase 1 of the algorithm, the angular domain is divided into 2^n sectors, using the 3dB-angular beam width. Hence, as the algorithm proceeds to higher levels, finer beams are used and the Rx's direction is estimated with finer resolution. In phase 2, ranging is performed with RIS-assisted beam focusing, in order to estimate the Rx's distance from the RIS, along the direction identified in phase 1. At the m^{th} hierarchical level of phase 2 of the algorithm, the radial domain is divided into 2^m sectors, using the 3dB-extent of the focal areas.

Hence, as the algorithm proceeds to higher levels, beams with narrower focal areas are used and the Rx's distance is estimated with finer resolution.

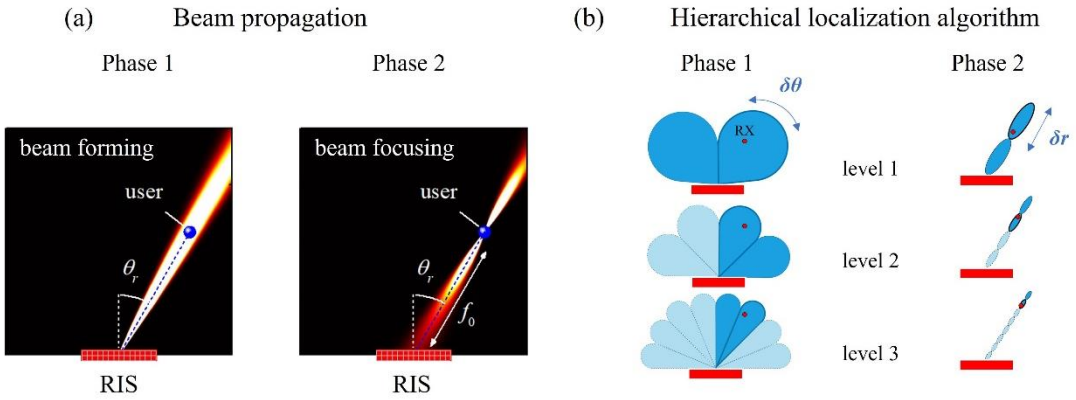

Figure 22: RIS-as-a-sensor. (a) Beam propagation. (b) Schematic representation of the hierarchical localization algorithm [1].
The beams in phase 1 and focal areas in phase 2 represent all codewords that are stored at each level of the corresponding codebook. Their dark-toned versions represent example codewords that are used in the hierarchical approach, to locate the RX shown in this example. The resolution in phase 1 is determined by the 3-dB beam width, $\delta\theta$, and in phase 2 by the 3-dB extent, δr, of the focal area.

With information about both the direction and the distance, the location of the Rx is fully determined. Essentially, the area in front of the RIS is represented by a polar grid, with the RIS at its origin. The angular resolution is determined by the minimum angular beam width, $\delta\theta$, and beam separation in phase 1 and the radial resolution is determined by the minimum size of the focal areas, δr, formed in phase 2 at each hierarchical level, as depicted in Figure 22 (b).

The concept of RIS-as-a-sensor unifies the outcomes of WP1, WP2 and WP3 in a unique way; it combines advanced wavefront engineering aspects (studied in WP2) with algorithm design for sensing (explored in WP3) into a scenario (selected in WP1) that falls within the *communicate-to-sense* pillar.

The RIS-as-a-sensor concept involves several challenges that we aim to address in HW Demonstrator II. One major challenge is that the RIS that will be used is 1-bit, i.e. it involves only two phase levels, and it cannot provide access to the entire $(-\pi, +\pi)$ range. This is expected to limit the resolution capability (with respect to an ideal RIS) and also cause unintentional side-lobes. Additionally, it should be taken into account that the beam focusing efficiency depends on the size of the RIS, i.e. larger RIS panels promote sharper focal points [4]. Therefore, to ensure beam focusing and to enhance the beam focusing capabilities, we will need to combine several GW FR2 RIS's to extend the RIS area adequately. Depending on the resolution capabilities of the HW setup, the algorithms will be modified and tailored to the experimental configuration.

A comparison between an ideal RIS with access to the entire $(-\pi, +\pi)$ range (full phase) and a 1-bit RIS is demonstrated in Figure 23. In this example the operation frequency is 28 GHz, and the RIS consists of 80 × 80 elements ($\lambda/2$ separation), yielding a RIS area of ~43 cm × 43 cm. The necessary phase for achieving focusing at 2 m from the RIS is shown in Figure 23 (a). In Figure 23 (b) the focused beam generated by each RIS is shown explicitly. In Figure 23 (c) the beam power along its center (on-axis beam power) is calculated for the two cases, as well as for an

intermediate case using a 2-bit RIS. These results give promise for efficient focused beam formation with the hardware specifications of HW Demonstrator II.

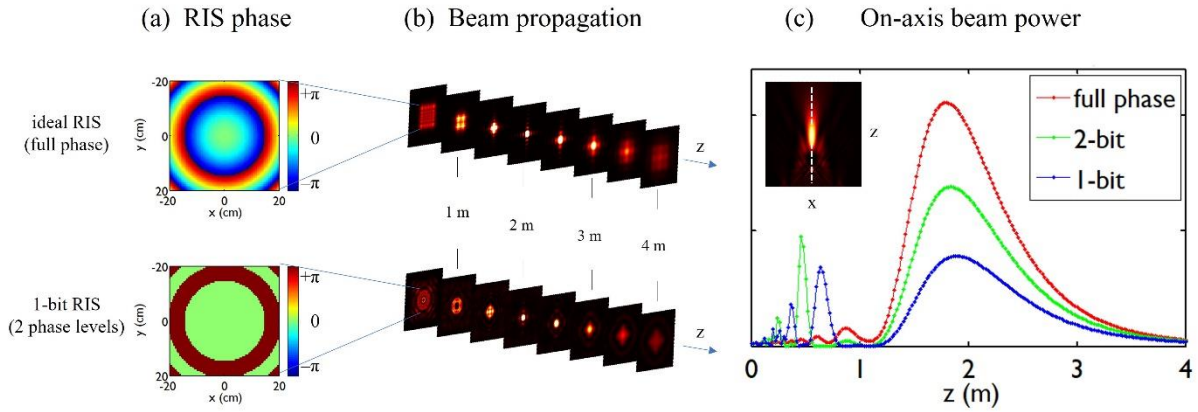

Figure 23: RIS Example implementation of RIS-as-a-sensor phase 2 using an ideal RIS and a 1-bit RIS. (a) RIS element phase distribution, required to focus a beam at 2 m in front of the RIS. (b) Propagation of focused beams, generated using the RIS phase in panel (a). (c) On-axis beam power for the two examples, also showing the expected on-axis beam power for a 2-bit RIS.

4 Definition of a Hardware Demonstrator III "UAV-supported 5G&GNSS Fusion-based Localization Framework for Enhanced S&R Operations"

The details of the third hardware demonstrator are presented below.

4.1 Demonstrator description

This demonstrator will showcase a two-stage system leveraging the fusion of 5G New Radio (5G-NR) and GNSS signals for precise positioning, and applying it to UAV-based search and rescue (S&R) missions. Developed within INSTINCT, this hardware demonstrator will integrate AI-based optimization techniques to address real-world challenges in positioning accuracy, availability, and robustness.

4.1.1 Stage 1 – AI-Based Optimization of 5GNSS for UAV Localization

The first stage will demonstrate 5GNSS [28], a system that fuses GNSS and 5G-NR signals for enhanced localization performance, with the focus on deployment aboard outdoor UAV platforms. This stage will also validate the robustness of the AI-optimized 5GNSS system and its application in UAV platforms operating under varying environmental constraints. GNSS offers global coverage but suffers in urban and indoor settings, while 5G-NR provides enhanced accuracy in dense environments. Their combination, optimized using AI, is expected to mitigate the limitations of each technology individually. Key features of the demonstrator will include:

- **AI-based adaptive fusion of 5G-NR and GNSS signals,** accounting for signal quality, multipath, and non-line-of-sight effects.
- **Error modeling and correction through AI to** handle measurement noise, clock offsets, and asynchronous data sources.
- **Software-defined testbed** integrating USRP with OpenAirInterface (OAI) for 5G-NR data collection and GNSS Logger for satellite data, using a single Android phone to ensure temporal alignment.

4.1.2 Stage 2 – 3DSAR: UAV-Based Victim Localization Using Passive 5G-NR Signals

Building upon the self-localization capability from Stage 1, the second stage will demonstrate 3DSAR [29], a UAV-based S&R system that enables passive localization of victims via their 5G smartphones without requiring any user-side configuration. This stage targets accuracy-sensitive emergency scenarios where traditional localization techniques (e.g., GNSS-based app sharing) are not feasible. It showcases the ability of autonomous UAVs to navigate and locate individuals on the ground using only 5G network presence. Core capabilities will include:

- **Non-cooperative localization** using control-plane 5G-NR signals from victim devices.
- **UAV-mounted 5G-NR sensors** to passively estimate distance and/or angle to detected smartphones.

- **Onboard integration with 5GNSS**, ensuring accurate UAV positioning while performing victim detection.

4.2 Implementation Details

The setup will integrate software-defined radio (SDR) platforms, commercial smartphones, and open-source 5G and GNSS toolchains, and will be designed to support both controlled lab experiments and mobile outdoor trials.

The 5G-NR implementation will comprise two core components: the 5G core network and the Radio Access Network (RAN). The 5G core will be based on Open5GS, an open source 5G SA (standalone) core network implementation. The test setup includes a reprogrammable SIM card embedded in a commercial smartphone, ensuring that the UE can reliably attach and register with the testbed network. Custom SIM parameters (e.g., IMSI, MCC/MNC) will be configured to match those expected by the core.

The gNB will be implemented using the OAI software stack, compliant with 3GPP Release 16. OAI supports deployment on both Intel and ARM architectures and interfaces with common SDR platforms such as USRP and BladeRF. We will use the Ettus USRP B210 as the SDR hardware, chosen for its compact form factor and full compatibility with OAI.

Key RAN configuration parameters will include: i) operating band n78 (~2.5 GHz), ii) subcarrier spacing of 30 kHz, iii) transmission bandwidth of up to 40 MHz, and iv) uplink SRS of 1 ms transmitted by the UE during every subframe, enabling ToF-based ranging at 100 Hz using channel impulse response estimation. To enable outdoor field trials, the RAN setup will be deployed on a portable Intel NUC (i7, Ubuntu 22.04 LTS), powered by a high-capacity external battery. This design ensures full mobility and independence from fixed infrastructure.

At the user side, a Google Pixel 8 smartphone running Android 12 will be used. The smartphone will be equipped with a Sysmocom reprogrammable SIM, allowing network-specific parameters (e.g., PLMN, IMSI) to be configured to match the custom 5G core. This will ensure consistent registration to the test network and enable collection of uplink SRS measurements for passive localization in 3DSAR.

GNSS data will be collected using the GNSS Logger application from Google. This tool allows for the collection of raw GNSS measurements (e.g., pseudoranges, Doppler shifts, satellite IDs) from all visible satellites across multiple constellations (e.g., GPS, Galileo). The GNSS data is time-synchronized with 5G data using the co-located smartphone setup, ensuring precise alignment for fusion and AI-based processing.

4.3 Testbed Implementation Plans

The demonstration of 5GNSS and 3DSAR systems will be carried out in the DroneLab, an UAV research and testing facility of the Universitat Politècnica De Catalunya (UPC) located in the Baix Llobregat Campus in Castelldefels, Spain. DroneLab is a pioneering facility in Spain and Europe, dedicated to the research, testing, validation, and training of UAVs.

The DroneLab supports a wide range of UAV-related research domains, including autonomous navigation, communication systems, and integration with 5G networks. Moreover, it provides advanced testing capabilities: as it is equipped with state-of-the-art technology to conduct comprehensive experiments and validate UAV systems under various scenarios.

Finally, the DroneLab is a flight facility segregated from general airspace and not restricted to aviation regulation. It is equipped with technology to conduct experiments, research and teaching related to the subject. Measuring 80 x 40 x 15 meters, the netted enclosure guarantees safety and will allow any type of professional application or technology development related to drone-based logistics, automatic operation and the exploitation of 4G, 5G and the Internet of Things (IoT) communications to be validated.

4.4 Test and Demonstration Scenarios

To validate the effectiveness of the 5GNSS and 3DSAR systems, several real-world inspired scenarios are designed, focusing on non-cooperative victim localization using UAV-mounted 5G-NR receivers. These scenarios aim to assess system accuracy, responsiveness, and robustness under diverse environmental and operational conditions:

For both stages of the demonstrator, the considered scenario is an open field rescue scenario:

1. **Objective:** Establish a performance baseline for passive 5G-NR-based victim localization in optimal conditions.
2. **Setup:** A single UAV flies over an open area with clear line-of-sight to one or more smartphones emitting standard 5G control signals.
3. **Evaluation focus:** Localization accuracy, altitude impact, UAV scanning trajectory, latency from detection to localization.

The main results of the demonstrator will include the public showcasing of the 5GNSS and 3DSAR prototypes at the European Conference on Networks and Communications (i.e., EuCNC 2025 and EuCNC 2026, respectively). Each subsystem will be demonstrated in real-world conditions at the DroneLab facility, with both video recordings and dedicated demonstration videos produced to illustrate their operation and performance. Additionally, a rigorous experimental performance evaluation will be conducted based on the collected measurement data, allowing for quantitative assessment of positioning accuracy, system responsiveness, and robustness under different environmental and operational conditions.

5 Definition of a Software Simulation Platform / SW Demonstrator

We build a software simulation platform to extend the HW demonstrations to more complex scenarios. Core is the Software Tool from Inria.

5.1 Demonstrator Description

Here we give an overview of the scope and general concept of the demonstrator.

5.1.1 Realistic AI-based Coverage Map Prediction

In this section we describe the software demonstrator that is being developed at TID. Specifically, we are working towards an ML-based tool for coverage map prediction in real-world environments. This tool is designed to effectively capture the 3D characteristics of buildings and estimate how the radio signal propagates within the environment. Compared to traditional ray-tracing methods, our approach offers lower computational cost and the ability to learn directly from real-world measurements, a key feature that enables realistic coverage map estimation.

To train the ML-based model, we initially rely on Sionna RT [14] to create synthetic datasets for training the model. The results obtained from the simulator are used as ground truth labels for training the ML model using a supervised-learning approach. These datasets need to be representative enough to enable generalization to different antenna placements, configurations, etc.

The software demonstrator is composed of two main modules:

1. **Environment digitalization:** To model radio signal propagation in urban environments, we convert the continuous 3D real-world environment into a discretized digital equivalent space. This digital representation includes information on the building location, size, orientation, and geometric characteristics, all obtained from OpenStreetMap (OSM) [10]. The objective is to leverage a ML model to estimate the coverage map of a given real-world scene. To do that, the digital environment is converted to a graph G that captures the geometric characteristics of the buildings, the antenna configuration and the receiver position.
2. **Machine Learning Model**: Once we have built the graph representation of the real-world environment, we use a Graph Neural Network [27] to model the radio propagation in the 3D environment. The graph representation G and the carefully designed edge/node features capture accurately the geometric characteristics of the scene. In addition, the features provide meta-information that helps capture the orientation of the antennas with respect to the receivers and the other buildings. This is critical to learn radio propagation and to enable learning natural physical phenomena such as diffraction, scattering and reflection.

Model Inputs

As mentioned previously, we leverage OpenStreetMap to obtain the digital representation of the real-world environment. In addition, we also use a dataset of real-world antennas deployed in the United Kingdom (UK). This dataset contains the latitude and longitude coordinates of each antenna of a mobile network operator in the UK, together with the antenna configuration. Combining this information with OSM, we create a dataset of different digitalized environments. In other words, for each antenna deployed, we extract the 3D building information from OSM, creating a digital representation of the real-world, also referred to as scene.

Once the scenes have been downloaded and the dataset has been created, we proceed to convert them to a point cloud representation. This means that instead of representing the 3D geometry of each building using 2D planes, we leverage a point cloud that captures the geometrical information of the buildings. In addition to the geometric information, each scene includes a transmitter and multiple receivers. The transmitter is placed according to the relative positions in the real-world environment and the receivers are uniformly distributed across the scene in a 2D plane at a height of 1.5 meters. Then, the elements of the scene are converted to a point cloud, where each point is converted to a node of the graph. We interconnect the nodes by adding edges in a way that resembles the potential paths the electromagnetic waves follow from the transmitter to a given receiver.

Each node and edge have a feature vector with features associated to the orientation of the nodes, distances between nodes and the angles between edges and node orientations. We build a graph for each receiver, changing the connectivity of the graph accordingly. Then, each graph is processed by our GNN model, outputting the predicted value of interest for the corresponding receiver. This procedure is repeated for all receiver points from the point cloud representation. Figure 24 shows a general overview of the entire software demonstrator, including the digitalization of the real-world environment, the graph representation and the ML model.

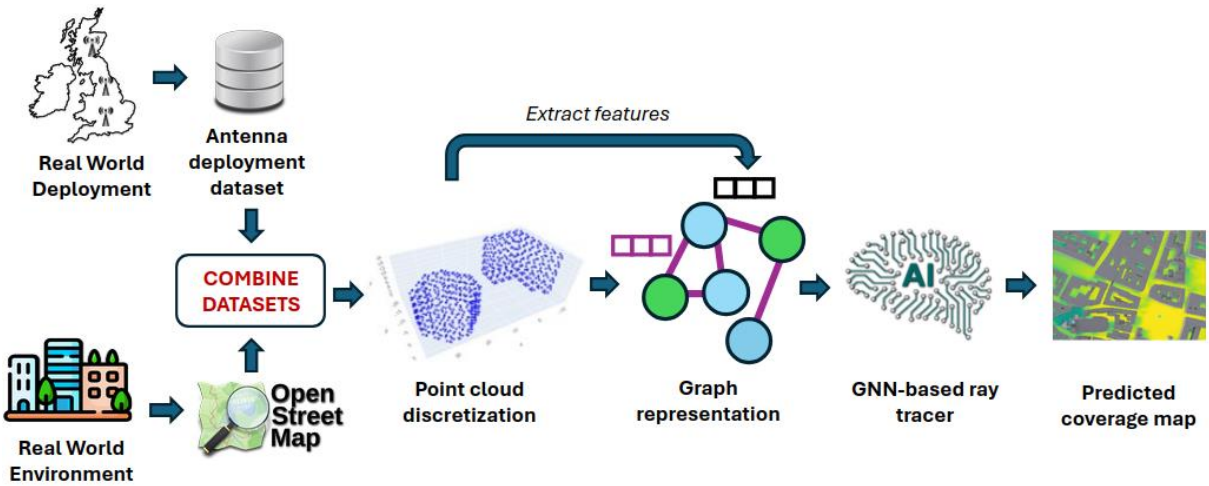

Figure 24: General overview of our software demonstrator

5.1.2 Understanding the entanglement between sensing and communication using stochastic geometry

The proposed software simulator is designed to evaluate the joint performance of JCAS compared to stand-alone Sensing or Communication. The software analyzes the interaction between communication and sensing operations carried out by BSs, using a joint waveform for both radar and communication functions. The key point is that the performance metrics are at *system level*, namely, they incorporate the fact that in such a large network, other BSs also communicate and sense and that this impacts the performance of any given BS communication or sensing process due to arose interference. The simulation environment models a large-scale network deployment where each BS serves multiple UEs while simultaneously tracking the parameters of interest of associated SOs. A Voronoi tessellation is used to define the coverage cell of each BS, defined from the input area and BS locations (see Figure 25).

The downlink communication between BSs and UEs is represented as a M/M/1 queue [5], capturing the stochastic nature of packet arrivals and service times. For object tracking, the framework implements a Kalman Filter (KF) estimator for each SO, with the temporal evolution of the SO's parameters modelled as a Gauss-Markov process. This process is parametrized according to SO-defined kinematic constants, enabling flexible simulation of various motion patterns. An example of captive motion, i.e., constrained within the BS respective cell is given in Figure 26.

Basic metrics within this setting are $SINR_{com}$ *and* $SINR_{rad}$. The simulator evaluates these basic metrics as well as derived metrics, which leverage the basic ones. These derived metrics are the stationary queue size of the communication link and the stationary covariance of the KF of the sensing estimation process.

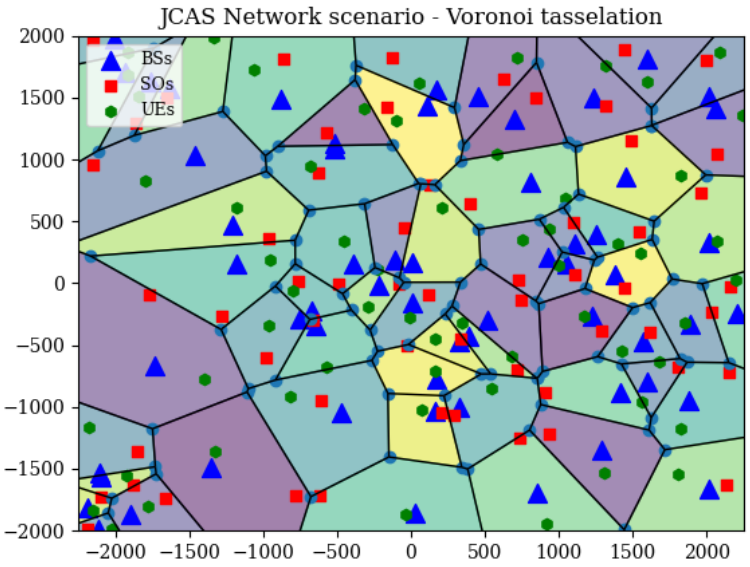

Figure 25: Network tessellation

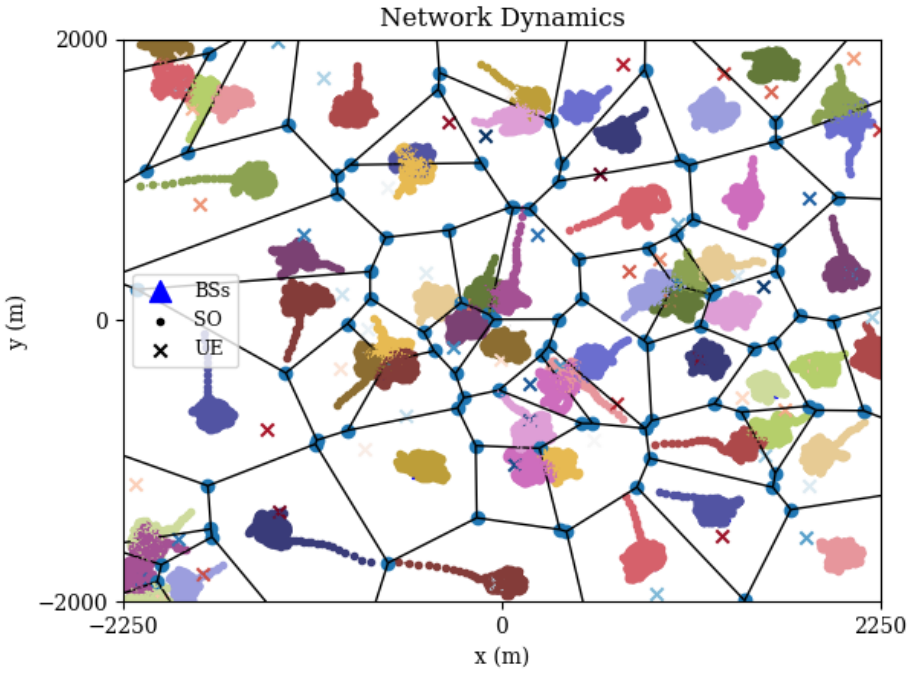

Figure 26: SO captive dynamics

5.2 Implementation Details

Here we give details about the software specifications.

5.2.1 Implementation details of TID

Once we have built the graph representation of the digital representation, we use Message Passing Neural Networks (MPNNs) [26] to implement our GNN-based model. This neural architecture processes all the graph-structured information and exploits the relationships between the different nodes in the graph. MPNNs are deep learning models specifically tailored to process graph-structured information. For example, unlike other popular neural network models (e.g., convolutional, recurrent, multi-layer perceptron), these models support graphs of variable number of nodes and edges and, more importantly, they are equivariant to node and edge permutations [27]. In the context of coverage prediction, this means that by representing the 3D scene as a graph, MPNN-based models can find symmetries or equivalent patterns (e.g., combinations of relative distances and orientations) between the scenes seen during training and new ones where the model is applied after training.

5.2.2 Implementation details of Inria

5.2.2.1 Input

The simulator is parametric. One can configure it by specifying the following parameters:

1. **The network size**: The network is represented as a 2-dimensional square torus. Its size is represented as the side of this torus. This torus is a compact representation of the Euclidean plane.
2. **The initial placement of BSs, UEs, and SOs**: These are determined either via a stochastic Point Process (PP) (see [5,6]), or by injecting real-world data. In the PPP case, these PPs are characterized by their intensity parameters. It is assumed that each cell has many UEs and SOs. A single (UE, SO) pair is selected at random in this set.
3. **The motion settings for UEs and SOs**: This includes special cases such as motionless behavior, constrained (captive) motion with respect to BS cells (e.g., Gauss-Markov model), or free movement across the network (e.g., along a straight line). Motion speed can also be tuned. When UEs and SOs move freely, they may change their associated BS over time. This results in handovers of downlink service and KF estimation, ensuring the preservation of queue states and estimation history.
4. **The radio environment**: Includes path loss, fading, bandwidth, power, thermal noise power, etc.
5. **The KFs and queue parameters**.
6. **Selection of operational mode**: JCAS, Sensing-only, or Communication-only.

5.2.2.2 Methodology

1. A Voronoi tessellation is used to define the coverage cell of each BS (see Figure 25), based on the input area and BS locations.
2. Interference from both communication and radar activity is modeled as additive shot-noise fields.
3. A time-based resource scheduling mechanism manages the operational mode (sensing or communication) and multiplexing.
4. Aggregated statistics are computed from the spatial average of the steady-state queue and the ℓ_2-norm of the KF error covariance distributions.
5. Marginal probability distributions are estimated using a Kernel Density Estimation (KDE) algorithm applied to the computed averages.

5.2.2.3 Output

The simulator provides results at both BS-level granularity (raw results) and aggregated network-level statistics (aggregated visualization). Additionally, it provides network snapshots in separate files.

5.2.2.4 Raw Results

For each BS and its associated UEs and SOs, the simulator delivers:

- KF estimates vs. true state (1D visualization) for all tracked dimensions.
- Evolution of absolute estimation errors over time.
- KF stability analysis.
- Evolution of the ℓ_2-norm of the KF error covariance over time.

- Shannon rate evolution.
- Queue workload dynamics.
- Packet queue dynamics.

5.2.2.5 *Aggregated Visualization*

1. A snapshot of the network topology.
2. A snapshot of UEs and SOs trajectories in the network.
3. Markdown and LaTeX-formatted tables of parameter settings.
4. **JCAS mode**: Marginal joint probability distributions of queue workload and KF error covariance ℓ_2-norm.
5. **Sensing mode**: Marginal spatial probability distribution of KF error covariance ℓ_2-norm.
6. **Communication mode**: Marginal spatial probability distribution of queue workload.

5.2.2.6 *Complementary Files*

- A time-based log file capturing network snapshots (BSs with associated UEs and SOs) at each discrete simulation step, along with handover history.

The software is currently hosted on a GitHub repository and is intended to be developed into a web-based application.

5.3 Test and Demonstration Scenarios

Here, we list the various scenarios in which the demonstrator will be tested.

5.3.1 Test plans of TID

For testing the performance of our software demonstrator, we are planning to leverage a real-world crowd-sourced measurements dataset from a mobile network operator in the UK. Initially, we are planning to select the received signal strength (RSS) measurements from two different cities in the UK, referred to as city A and B, and merge them with the antenna deployment dataset. We will select one antenna from city A and one from city B, creating two different digital environments with a set of empirical measurements. For each city, the samples will be split for training, testing and validation following a 70%, 20% and 10% ratio respectively. Once the ML model is trained, we will proceed to compute several evaluation metrics (i.e., mean absolute error, mean squared error, among others) for the predicted RSS values for the test set.

5.3.2 SG-JCAS testing by Inria

We intend to complement our empirical findings with an analytical framework, which is currently under development.

Regarding the evaluation scenarios, the following setups has been planned:

5.3.2.1 Basic Scenario

The baseline scenario examines varying degrees of mobility constraints imposed on UEs and SOs—namely, free, captive, and stationary states. By composing different motion patterns, we can simulate a diverse set of mobility conditions and study their impact on JCAS performance.

5.3.2.2 Enhanced Scenario: Impact of Reconfigurable Intelligent Surfaces (RIS)

This scenario involves introducing a set of RIS instances to evaluate their effect on communication performance. The aim is to compare system behavior with and without RIS deployment, focusing on the potential performance gains as well as associated trade-offs, such as spectrum efficiency and computational complexity.

5.3.2.3 Scenario Description

We consider a situation in which a SO obstructs the line-of-sight (LOS) downlink path between the BS and the UE. In response, the system leverages RIS to re-establish an alternative LOS path. The RIS units can be spatially distributed either deterministically or according to a PP. The objective is to quantify the trade-off between utilizing this secondary, RIS-assisted channel versus the blocked primary channel, in terms of Quality-of-Service (QoS) degradation. This analysis is essential for informing policy decisions that must consider the increased interference risks due to possible over-signaling. A schematic illustration of this scenario is provided in Figure 27.

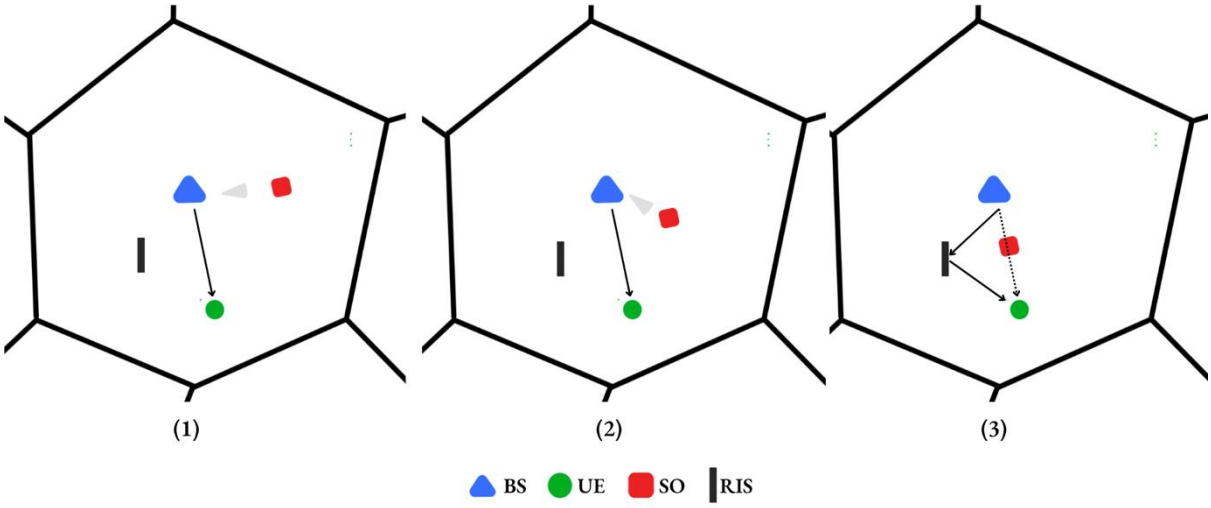

Figure 27: Use case sketch. The view proposes a discretized sequence of images where the SO ends up blocking the BS-UE LOS link, prompting the BS to leverage an RIS for re-establishing a LOS path

5.4 Ray Tracing Tool Development

5.4.1 Introduction

This chapter describes the ray tracer, which is been developed at UOulu. The development of the ray tracer tool has been started already before the INSTINCT project and the main parts have been developed in earlier EU-funded projects AI4Green and Ariadne. The work in INSTINCT concentrates on adding functionalities related to the ISAC concepts, namely the introduction of

target models and RIS units into the ray tracer. The tool is planned to be used in Task 4.4. of the INSTINCT project.

A common approach in ray tracers have been to use shooting-and-bouncing-ray method (SBR) [7] but this approach has problems with multiple diffractions. Therefore, a visibility analysis -based path search was selected for investigation in the projects mentioned, which resulted in the method described in [8].

In the following, an overview of the ray tracer is provided. We start by describing the approach used for providing geometrical input for ray tracing. A triangle mesh -based representation is used, which is complemented by a discretization-based modelling, which serves the needs of path computation. In addition, support for composing ray tracing scenes from multiple objects is discussed. Section 5.4.3 describes geometric path computation, which is done with the aid of GPU computation. Section 5.4.4 presents the approach for electromagnetic (EM) computation and related support. Finally, Section 5.4.6 discusses ideas related to application in the INSTINCT project.

5.4.2 Geometry input

The ray tracer uses triangle mesh -based geometry. A proprietary text file format (named TES) has been specified for this purpose, which includes grouping of triangles to planar walls and annotation of diffraction edges. In addition, named materials are associated with the triangles in the file format.

Restricted use of other file formats is provided by a proprietary tool (named gltf2tes), which is used to convert suitable GLTF files [9] to the TES format. For example, some building geometries available in OpenStreetMap [10] have been saved as GLTF with the aid of Blender software [11] and its OSM add-on. An example is shown in Figure 28.

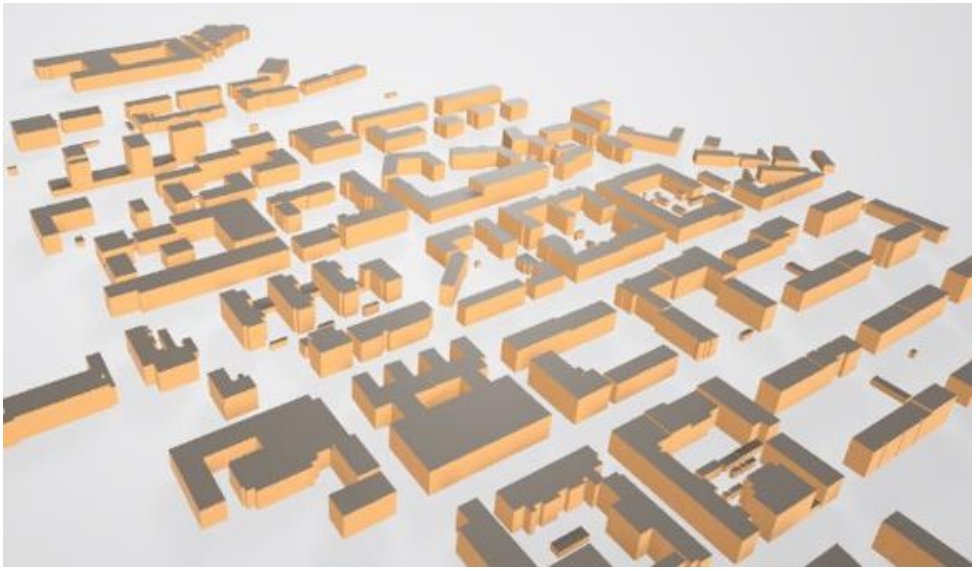

Figure 28: An example of building data downloaded from OpenStreetMap databases.

The ray tracer relies on discretization of the geometry, where each wall surface is represented by a set of tiles, and similarly diffraction edges are represented by a set of edge segments as

illustrated in Figure 29. Centroid points of tiles and edge segments are used in the analysis, which determines line-of-sight visibility for each pair of points. From the viewpoint of path computations, parameters controlling the tile and edge segment sizes are important.

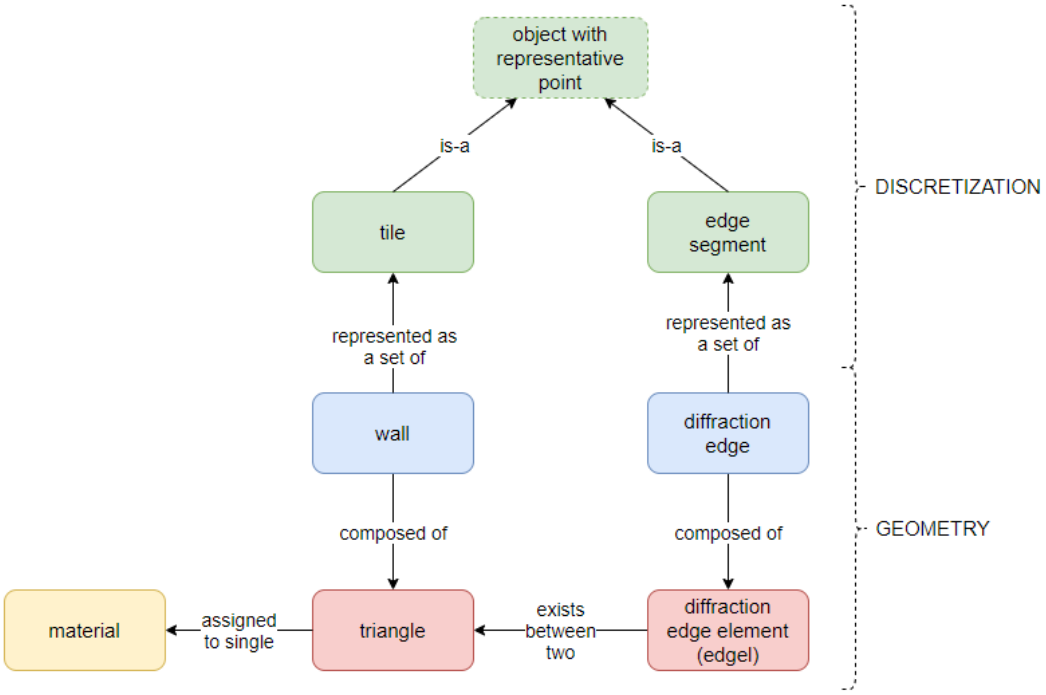

Figure 29: Elements of TES geometry and its relationship to discretization elements.

Moreover, transmitter/receiver antennas (and RIS devices too) are represented as points, and information about the points stored along with associated geometry discretization. An example is shown in . JSON files (referred as PTF) are used to store discretization information (tiles, edge segments, tx/rx points).

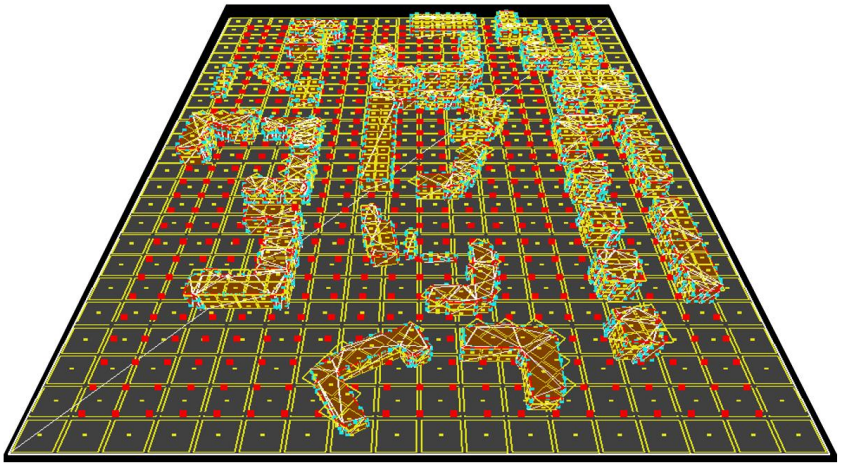

Figure 30: An example of discretization applied to a triangle mesh approximation of a region in Oulu. Tile-based representation can be seen from the ground area. The grid of red points shows RX points defined for the computation of a coverage map.

Components for the specification of ray tracing geometry are provided by TES and PTF files. Another JSON file format named GDI is used to define compositions to which ray tracing is applied. A GDI file provides a list of objects, which refer to TES and PTF files. Then, a description of the instantiation of those objects for one or more compositions is provided (Figure 30). An instantiation specified mappings of the objects to the world coordinate frame (WCF) using six parameters to define object position and orientation (rotation).

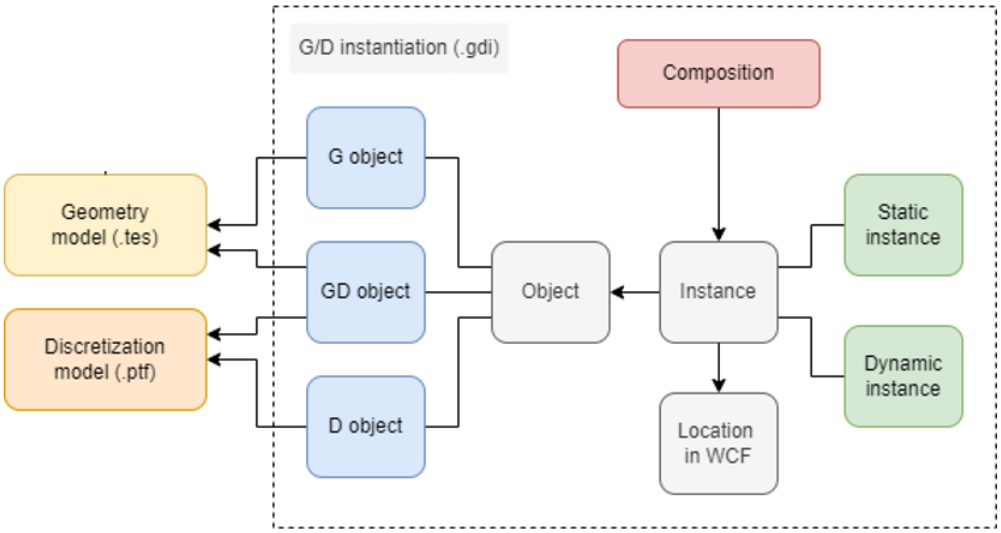

Figure 31: A GDI file specifies one or more compositions of the scene. An instance of an object can be static, which means that its location is fixed for all compositions, or dynamic, which means that its location varies over compositions.

5.4.3 Geometric path tracing

In geometric path computation, propagation paths between selected TX/RX point pairs are computed according to the laws of reflection, diffraction and refraction. The ray tracer supports path computation for chains of reflection and diffraction using a two-step approach [8]. In the first step, which is called route search, candidate geometric element sequences are determined using a two-level discretization-based visibility analysis. This is based on the ideas presented in [12]. In the second step, called path refinement, the intermediate points on the candidate paths are refined to obtain shortest-length optical paths, which are validated in the last step. Basically, this makes up a hybrid ray tracing method [7], where SBR ray tracing as the first step is substituted by an environment-driven approach described in [12].

Essential parameters in the path tracing include the maximum number of interactions allowed for the path. A specific limit is provided also for the number of diffractions. In earlier work, these values have been limited to the maximum of 4 interactions and 2 diffractions. To speed up computations, even lower values are typically preferred.

For simulation of diffuse scattering, tile-based discretization must be done for the geometry to establish scatter points. Then, to support EM computation, a two-step approach must be used to trace paths from TX to scatter points and from scatter points to RX. In this way, paths containing

a single scatter point can be supported for EM computation. Simulation of multiple scatter points on the path is not feasible[1].

Each path in the output of geometric path tracing is described by

- propagation delay (τ_i)
- angle of departure (AoD, $\Omega_{\mathrm{T},i} = (\theta_{\mathrm{T},i}, \varphi_{\mathrm{T},i})$)
- angle of arrival (AoA, $\Omega_{\mathrm{R},i} = (\theta_{\mathrm{R},i}, \varphi_{\mathrm{R},i})$)
- interaction points, each described by
  - interaction type (reflection/diffraction),
  - associated geometry element (wall/edge index)
  - coordinates of the point (x, y, z in WCF)

The output is stored using a proprietary JSON format.

5.4.4 EM computation

5.4.4.1 Principle

When a set of N geometric paths have been obtained to model the link between TX and RX antennas, the task of EM computation is to obtain channel impulse response (CIR) [13]

$$h(\tau) = \sum_{i=1}^{N} a_i \delta(\tau - \tau_i),$$

where $a_i \in \mathbb{C}$ are the path coefficients. Corresponding baseband CIR is [14]

$$h_b(\tau) = \sum_{i=1}^{N} a_i e^{-j\omega_c \tau_i} \delta(\tau - \tau_i).$$

As described in Sionna documentation [10], the path coefficients a_i are given by

$$a_i = \frac{\lambda}{4\pi} \mathbf{C}_{\mathrm{R}}^{H}(\Omega_{\mathrm{R},i}) \mathbf{T}_i \mathbf{C}_{\mathrm{T}}(\Omega_{\mathrm{T},i}),$$

where $\mathbf{T}_i$ is a 2x2 transfer matrix that encodes the effect of path on wave strength and polarization[2], and $\mathbf{C}_{\mathrm{T}}, \mathbf{C}_{\mathrm{R}}: \boldsymbol{\Omega} \to \mathbb{C}^2$ denote the antenna patterns of TX and RX antennas, respectively. Coefficients in the transfer matrices and antenna patterns are based on the use of local ray-based basis vectors. For a path with K interactions, the transfer matrix is computed as a product $\mathbf{T}_i = \mathbf{T}_{i,K} \mathbf{T}_{i,K-1} \cdots \mathbf{T}_{i,1} \mathbf{T}_{i,0}$, where $\mathbf{T}_{i,0}$ corresponds to the emission from the source point to the first interaction point (or destination point, if $K = 0$).

From this we can extract three possible tasks:

1. path modelling involving computation of $\mathbf{T}_i$,
2. computing electric field vectors at the receiver point $\mathbf{E}_{\mathrm{i}} = \mathbf{T}_i \mathbf{C}_{\mathrm{T}}(\Omega_{\mathrm{T},i})$, and
3. computing path coefficients a_i.

[1] Computational complexity of such implementation would be high. In some tools, scattering is allowed to be the last interaction on the path.
[2] The phase shift due to wave propagation not included.

For example, if we consider modelling of the wave propagation via RIS, we will perform path modelling for TX-RIS and RIS-RX links. Adding TX antenna pattern allows us to compute models for the N_I rays arriving to RIS ($\mathbf{E}_i^{\text{in}}$; $i = 1, \ldots, N_I$) and applying the RIS model provides departing ($\mathbf{E}_{i,j}^{\text{out}}$; $j = 1, \ldots, N_J$) for AoDs of N_J RIS-RX link paths. Then, path coefficients for the RX would be computed using

$$a_j = \frac{\lambda}{4\pi} \mathbf{C}_{\text{R}}^H(\Omega_{\text{R},j}) \mathbf{T}_j \mathbf{E}_j^{\text{out}},$$

where $\mathbf{E}_j^{\text{out}} = \sum_{i=1}^{N_I} \mathbf{E}_{i,j}^{out}$. For modelling of diffuse scattering, a similar two-link approach is used, that is, a scatter point S connects links TX-S and S-RX.

5.4.4.2 Path modelling

As mentioned in Section 5.4.1, a single named material is associated with each triangle in TES files. Materials associated with path interaction and associated face normals are fetched using interaction point info available in geometric path tracing output.

As usual, modelling of reflection is based on Fresnel equations [15]. Diffraction modelling is based on the Uniform Theory of Diffraction (UTD) [16], whose extensions are discussed in [17,18,19]. For diffuse scatter modelling, we use the approach described in is based on the approach outlined in [20,21]. Especially, we combine the reciprocal model from [21] with the option of using the Lambertian model.

EM modelling requires definition of material parameters, which are outlined in

Table 2. Permittivity and conductivity of some common materials is provided in [15].

Table 2: Material modelling parameters.

Parameter	**Notes**
Frequency	t.b.d.
Complex relative permittivity	Eq. (10) in [15]
Conductivity	t.b.d.
Effective roughness	Fixed or dependent on incident angle
Diffuse scatter lobe model	Controlled by two parameters
Diffuse scatter lobe shaping	Used in directive and double-lobe models [20,21]
Diffuse scatter cross-pol leakage	K_x; e.g., see scatter modelling EM primer of [14,16]

5.4.4.3 Antenna simulation

To provide support for simulation with various antenna patterns ($\mathbf{C}: \boldsymbol{\Omega} \rightarrow \mathbb{C}^2$; recall Section 5.4.4.1), the ray tracer includes a Matlab package, which specifies a common interface for accessing the pattern. Currently, in addition to simple models like dipole, support for some approaches like interpolation of slices and using patterns obtained via Satimo StarLab

measurements is available. An example of antenna pattern modelling is shown in Figure 32, where the horizontal and vertical slices (top images) have been interpolated to obtain a 3D antenna pattern (illustrated in two ways in bottom images).

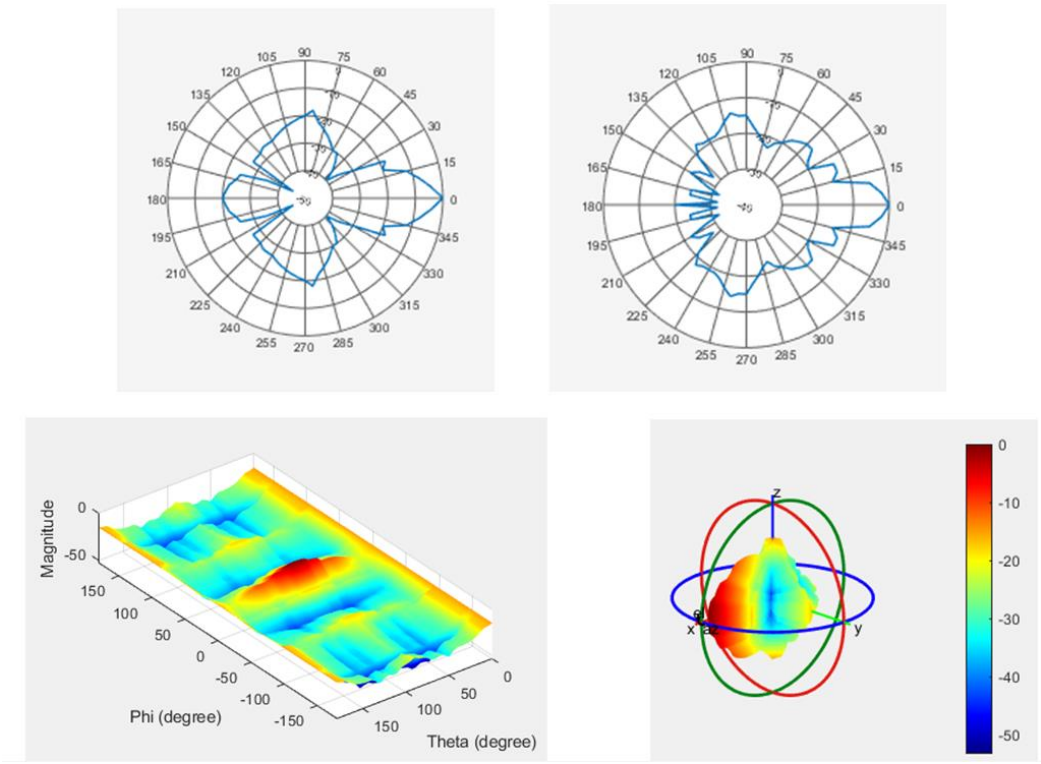

Figure 32: An example of antenna pattern modelling.

5.4.5 Implementation

The ray tracer has been implemented as a combination of binary executables and Matlab scripting. Executables perform the route search and path refinement. For these tasks, NVIDIA's OptiX ray tracing engine [22] is used for the task of determining point visibilities. In addition, path refinement has been implemented as accelerated GPU computations with CUDA [8].

Executables are called from Matlab scripts, which take care of preparing the required input files for ray tracing. After geometric paths have been obtained, subsequent EM calculations are done in Matlab. A use case may involve multiple launches of path computations. For example, simulating a case involving RIS requires modelling of TX-RIS link and RIS-RX link, simulation of RIS and simulation of antennas. Therefore, dedicated scripts must be implemented to organize computations. An example is shown in Figure 33.

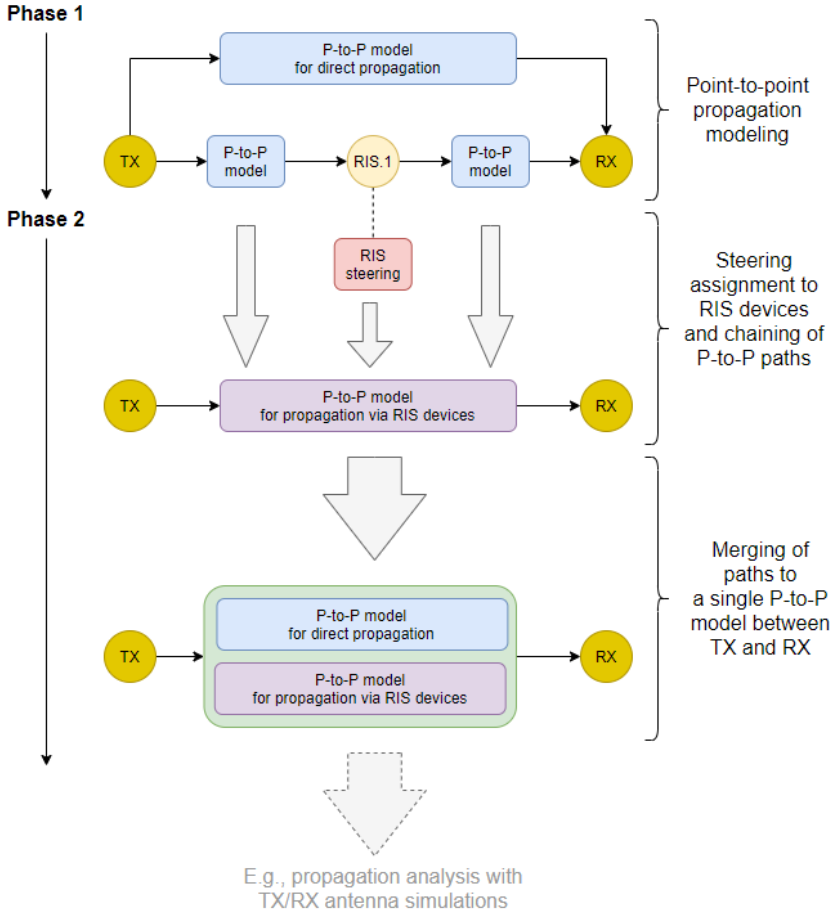

Figure 33: Multiple phases of computation in a RIS scenario. Implementation of ray tracing calculations for particular use case may involve organization to multiple steps, as shown in this figure.

Restrictions. Both computational complexity and memory consumption becomes easily high with ray tracing. When application of the tools is considered, we must pay attention to complexity of basic geometry, discretization granularity[3], and restriction for path interaction counts[4], for example.

5.4.6 On application to use cases of INSTINCT

In chapter 4 of [23], ISAC target modelling using a set of *geometrical scatter centers* is discussed. Considering the ray tracing tool, geometric computation of paths could be based on using scatter centers as TX/RX points in ray tracing, similarly to the approach used for RIS devices and diffuse scattering. Implementation of the simulation could involve

1. EM path computation from TX to RX (not via target).
2. EM path computation from TX to target points.
3. EM path computation from target points to RX.

[3] For examples, tiles of size 10x10 m2 are used in for urban scenarios.
[4] In experiments, we have restricted the number of interactions on the values less than or equal to 4. The diffraction count has been at most 2.

4. Simulation of target scattering to chain paths calculated in steps 2 and 3.

EM path computation in 1-3 involves geometric path computation and subsequent EM modelling. Then in step 4, having specified TX radiation, electric field vectors are obtained for each path entering the target points and, according to the target scattering model, they are mapped to the electric field vectors propagated over paths determined in step 3.

Simulation of a reconfigurable intelligent surface (RIS) can be managed in a similar way, i.e., propagating transmitted wave from TX to RIS, simulating RIS behavior, and finally propagating the RIS output to RX. Simulation was implemented in this manner in [24] where a phased array type RIS model was adopted to coverage studies (Figure 34). The model consists of multiple metallic plates, each which are small enough to act as diffuse scatterers, but which can jointly beamform a signal to a desired direction. The model has a physical basis with respect to scattering properties and gives a physically feasible pathloss law of the scattered signal as in e.g. [25]. The model mimics practical implementations, which consist of discretized sub-wavelength sized elements each having a constant phase shift. More information on the RIS model used and simulation done with this can be found in [24].

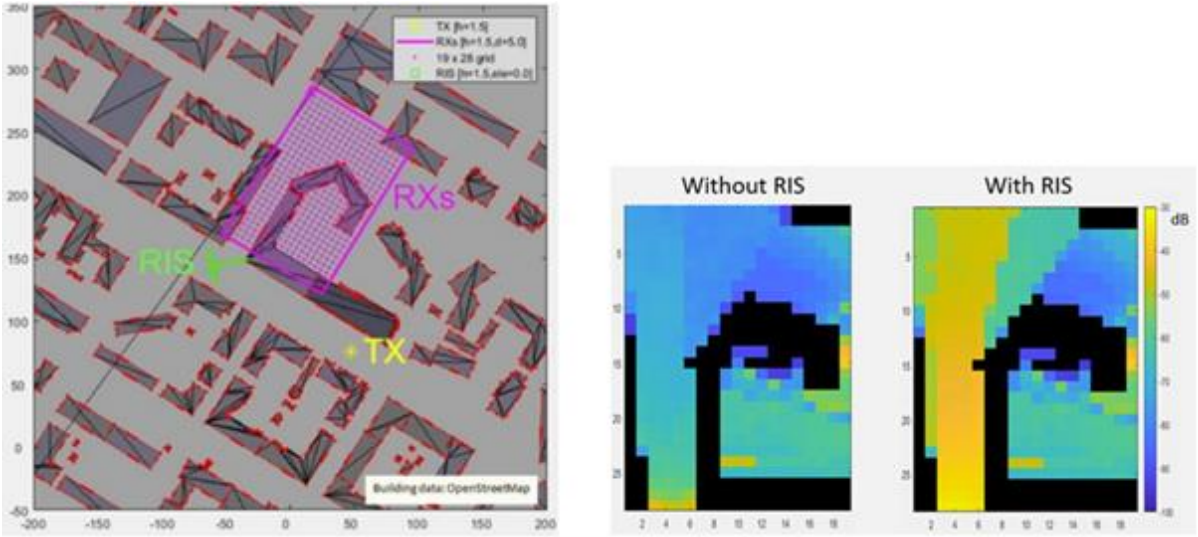

Figure 34: RIS coverage simulation in reference [20].

6 Conclusions

Deliverable D4.1 detailed the hardware and software specifications for three hardware demonstrators, alongside one software-based simulation platform. Additionally, it has provided preliminary guidelines for testing and measurement procedures intended for the upcoming evaluation campaigns.

In Section 2, two complementary hardware demos were presented leveraging the concepts of JCAS and RIS in FR1 (Sub6 GHz frequency range). The first demo presented by Insa-Lyon/INRIA offers a simple setup, in which the CorteXlab experiment room, divided into two inaccessible sections, uses a Greenerwave RIS to reflect sub-6GHz signals between these areas. Employing two USRPs — one transmitting pilot signals and the other receiving them, the setup validates significant changes in channel coefficients caused by optimized reflections from the RIS. The second demo presented by BI seamlessly evaluates communication and sensing algorithms on full duplex JCAS hardware by using BI's Hermes Py framework, where the same python code is used for evaluating results from software and hardware, gathering both sensing and communication KPIs.

In Section 3, the RIS-aided JCAS demonstration is presented, outlining two distinct scenarios: indoor and outdoor designed to assess the JCAS hardware provided by HHI, complemented by the FR2 RIS (so, focusing on FR2). Both scenarios will be executed on Bosch premises, showcasing monostatic sensing combined with communication and sensing enhanced by RIS. A cooperation between FR1 and FR2 demonstrations is under discussion, which may result in a short-time deployment of the FR2 setup also in the CorteXlab for a more detailed evaluation.

Additionally, although initially only two hardware demonstrations were planned for this work package, an additional distinct demonstrator emerged, warranting a separate showcase. Presented in Section 4, this innovative demonstration features a two-stage system that integrates 5G New Radio (5G-NR) with GNSS signals, aiming for high-precision positioning specifically suited for UAV-assisted search and rescue (S&R) missions.

In Section 5, we developed a software simulation platform to extend hardware demonstrations into more complex scenarios, utilizing Inria's core software. At TID, an AI-driven demonstrator is created for realistic coverage map prediction using graph neural networks trained with real-world antenna and OpenStreetMap data, reducing computational complexity. Inria contributed a stochastic geometry-based simulator to evaluate joint communication and sensing (JCAS) performance, incorporating realistic interference and mobility modeling. Finally, UOulu's ray tracing tool is enhanced by integrating target models and RIS, significantly improving electromagnetic propagation simulations for JCAS scenarios.

7 References

[1] Instinct Deliverable 1.1," INSTINCT use cases definition, requirements and system architecture".

[2] Instinct Deliverable 3.1," Initial report on orchestration and control of massive JCAS deployments".

[3] G. Stratidakis, S. Droulias, and A. Alexiou, "Near-Field Hierarchical Localization for Integrated Sensing and Communication," *IEEE Access,* vol. 13, pp. 17766-17776, 2025.

[4] S. Droulias, G. Stratidakis and A. Alexiou, "Near-Field Engineering in RIS-Aided Links: Beamfocusing Analytical Performance Assessment," *IEEE Access,* vol. 12, pp. 29536-29546, 2024.

[5] Błaszczyszyn, B., & Karray, M. K. (2019). Performance analysis of cellular networks with opportunistic scheduling using queueing theory and stochastic geometry. *IEEE Transactions on Wireless Communications*, 18(12), 5952–5966.

[6] F. Baccelli and B. Błaszczyszyn, *Stochastic Geometry and Wireless Networks: Volume I – Theory*, Foundations and Trends® in Networking, vol. 3, no. 3–4, pp. 249–449, 2009.

[7] Z. Yun, M.F. Iskander, "Ray Tracing for Radio Propagation Modeling: Principles and Applicatios," *IEEE Access,* vol. 3, pp. 1089 - 1100, 2015.

[8] N. Vaara, P. Sangi, J. Pyhtilä, J. Heikkilä, "A Refined Path Generation Pipeline for Radio Channel Propagation Modeling," in *17th European Conference on Antennas and Propagation (EuCAP)*, Florence, Italy, 2023.

[9] [Online]. Available: https://www.khronos.org/gltf/. [Accessed 30 3 2025].

[10] [Online]. Available: https://www.openstreetmap.org . [Accessed 30 3 2025].

[11] [Online]. Available: https://www.blender.org/. [Accessed 30 3 2025].

[12] J.S. Lu, et al., "A Discrete Environment-Driven GPU Based Ray Launching Algorithm," *IEEE Transactions on Antennas and Propagation,* vol. 67, pp. 1180 - 1192, 2019.

[13] D. Tse, Fundamentals of Wireless Communication, Cambridge University Press, 2005.

[14] "Sionna Ray Tracing (v1.0.1)," [Online]. Available: https://nvlabs.github.io/sionna/rt/. [Accessed 30 3 2025].

[15] "Rec. ITU-R P.2040-3 Effects of Building Materials and Structures on Radiowave Propagation above about 100 MHx. ITU Publication 08/2023".

[16] R. G. Kouyoumjian, "A Uniform Geometrical Theory of Diffraction for an Edge in a Perfectly Conducting Surface," *Proceedings of the IEEE*, vol. 62, no. 11, 1974.

[17] R. Luebbers, "Finite Conductivity Uniform GTD versus Knife Edge Diffraction in Prediction of Propagation Path Loss," *IEEE Transactions on Antennas and Propagation*, vol. 32, pp. 70 - 76, 1984.

[18] D.A. McNamara, C.W.I. Pistorius, J.A.G. Malherbe, Introduction to the Uniform Geometrical Theory of Diffraction, Artech House, 1990.

[19] "METIS Deliverable D1.4, "METIS Channel Models"," 2015.

[20] V. Degli-Esposti, F. Fuschini, E. M. Vitucci, G. Falciasecca, "Measurement and Modelling of Scattering from Buildings," *IEEE Transactions on Antennas and Propagation*, vol. 55, pp. 143 - 153, 2007.

[21] E.M. Vitucci, N. Cenni, V. Degli-Esposti, "A Reciprocal Heuristic Model for Diffuse Scattering from Walls and Surfaces," *IEEE Transactions on Antennas and Propagation*, vol. 71, p. 6072–6083, 2023.

[22] "NVIDIA OptiX Ray Tracing Engine," [Online]. Available: https://developer.nvidia.com/rtx/ray-tracing/optix. [Accessed 30 3 2025].

[23] "Next G Alliance Report: Channel Measurements and Modeling for Joint/Integrated Communication and Sensing, as well as 7-24 GHz Communication".

[24] J. Pyhtilä, J. Kokkoniemi, P. Sangi, N. Vaara, M. Juntti, "Ray Tracing Based Radio Channel Modelling Applied to RIS," in *26th International ITG Workshop on Smart Antennas and 13th Conference on Systems, Communications, and Coding (WSA&SCC)*, Braunschweig, Germany, 2023.

[25] Ö. Özdogan, E. Björnson, E. G. Larsson, "Intelligent Reflecting Surfaces: Physics, Propagation, and Pathloss Modeling," *IEEE Wireless Communications Letters*, vol. 9, pp. 581 - 585, 2020.

[26] Gilmer, J., Schoenholz, S. S., Riley, P. F., Vinyals, O., & Dahl, G. E. (2017, July). Neural message passing for quantum chemistry. In *International conference on machine learning* (pp. 1263-1272). PMLR.

[27] Satorras, V. G., Hoogeboom, E., & Welling, M. (2021, July). E (n) equivariant graph neural networks. In *International conference on machine learning* (pp. 9323-9332). PMLR.

[28] F. Campolo, A. Blaga, M. Rea, A. Lozano and X. Costa-Pérez, "5GNSS: Fusion of 5G-NR and GNSS Localization for Enhanced Positioning Accuracy and Reliability," in IEEE Transactions on Vehicular Technology, vol. 73, no. 9, pp. 13558-13568, Sept. 2024, doi: 10.1109/TVT.2024.3396991.

[29] A. Blaga, F. Campolo, M. Rea and X. Costa-Pérez, "3DSAR: A Single-Drone 3D Cellular Search and Rescue Solution Leveraging 5G- NR," GLOBECOM 2023 - 2023 IEEE Global Communications Conference, Kuala Lumpur, Malaysia, 2023, pp. 6183-6188, doi: 10.1109/GLOBECOM54140.2023.10437045.