Aalborg Universitet

Intelligent Sensing and Learning for Advanced MIMO Communication Systems

Vaca Rubio, Cristian Jesús

DOI (link to publication from Publisher):
10.54337/aau596083552

Publication date:
2023

Document Version
Publisher's PDF, also known as Version of record

Link to publication from Aalborg University

Citation for published version (APA):
Vaca Rubio, C. J. (2023). *Intelligent Sensing and Learning for Advanced MIMO Communication Systems*. Aalborg Universitetsforlag. https://doi.org/10.54337/aau596083552

General rights
Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal -

Take down policy
If you believe that this document breaches copyright please contact us at vbn@aub.aau.dk providing details, and we will remove access to the work immediately and investigate your claim.

Downloaded from vbn.aau.dk on: May 03, 2024

INTELLIGENT SENSING AND LEARNING FOR ADVANCED MIMO COMMUNICATION SYSTEMS

BY
CRISTIAN JESÚS VACA RUBIO

DISSERTATION SUBMITTED 2023

AALBORG UNIVERSITY
DENMARK

Intelligent Sensing and Learning for Advanced MIMO Communication Systems

Ph.D. Dissertation

Cristian Jesús Vaca Rubio

Aalborg University
Department of Electronic Systems
Fredrik Bajers Vej 7B
DK-9220 Aalborg

Dissertation submitted: August 2023

PhD supervisor:: Prof. Petar Popovski
Aalborg University

PhD Co-Supervisor: Prof. Zheng-Hua Tan
Aalborg University

PhD committee: Associate Professor Troels Bundgaard Sørensen (chair)
Aalborg University, Denmark

Professor Ana García Armada
Universidad Carlos III de Madrid, Spain

Associate Professor Ioannis Krikidis
University of Cyprus, Cyprus

PhD Series: Technical Faculty of IT and Design, Aalborg University

Department: Department of Electronic Systems

ISSN (online): 2446-1628
ISBN (online): 978-87-7573-650-8

Published by:
Aalborg University Press
Kroghstræde 3
DK – 9220 Aalborg Ø
Phone: +45 99407140
aauf@forlag.aau.dk
forlag.aau.dk

© Copyright: Cristian Jesús Vaca Rubio unless otherwise stated.

Printed in Denmark by Stibo Complete, 2023

Curriculum Vitae

Cristian J. Vaca-Rubio

He received the B.Sc. and M.Sc. degrees in telematics engineering and telematic and telecommunication networks from the University of Malaga, Spain, in 2018 and 2019, respectively. He is currently pursuing the Ph.D. degree with the Connectivity Section, Electronic Systems Department, Aalborg University, under a Marie Curie Fellowship as an Early Stage Researcher in the H2020 ITN WindMill Project, where he joined in September 2019. During his Ph.D., he was a visiting researcher at Mitsubishi Electric Research Laboratories (MERL), Cambridge, USA. From 2018 to July 2019, he was an Assistant Researcher with the Communication Engineering Department, University of Malaga in a joint project with DEKRA Testing and Certification S.A.U. His main research activities are in wireless communications and sensing, machine learning for signal processing, mobile networks, telematics, and applied statistics in video streaming. He received the IEEE International Conference on Acoustics, Speech, and Signal Processing top 3% paper award in 2023.

Abstract

Massive multiple-input multiple-output (MIMO) systems are a key technology for the fifth-generation (5G) wireless networks. Therefore, evolving towards the 6th generation (6G), MIMO and massive MIMO technology have to be upgraded to take advantage of the new groundbreaking technologies in the field of artificial intelligence (AI). The intersection between AI and wireless sensing is envisioned to be one of the key enablers of future 6G. With this background and motivation, this thesis investigates three topics: large intelligent surface (LIS), millimeter-wave (mmWave) Wi-Fi, and AI-enabled resource allocation and beam tracking, all within the context of MIMO systems.

First, this thesis explores the LIS technology for the integration of wireless communication and radio sensing. We establish that an LIS, acting as a radio image that describes the propagation environment, can provide a high-resolution rendering of the physical world. This enables the leverage of computer vision (CV) and machine learning (ML) techniques to develop precise sensing methods, underpinning the potential of LIS-based sensing in MIMO-based indoor industrial environments, among other applications discussed throughout the thesis.

Second, this thesis explores mmWave Wi-Fi for indoor wireless sensing, laying the groundwork for improved object localization. By proposing a novel dual-decoder neural dynamic learning framework, this work aims to address the unique challenges of mmWave Wi-Fi localization, including the intermittency of beam training measurements and the necessity to exploit underlying object dynamics. These advancements align with the 6G vision of providing accurate and efficient localization solutions in MIMO systems that can be directly integrated into the Wi-Fi standards.

Finally, this thesis explores the joint dynamic resource allocation and beam tracking for mmWave MIMO systems, a critical task for achieving ultra-reliable low-latency communication (URLLC). This part of the work leverages a proximal policy optimization (PPO) framework to dynamically allocate resources and track multi-user beams, taking into account the variable quality of wireless channels, thereby improving the overall system performance and reducing tracking overhead.

In summary, the main outcome of the thesis is the proposal of radio sensing AI-based methods to cover areas such as integrated sensing and communication, efficient localization methods, and dynamic resource allocation. This research provides an essential foundation for future advancements in MIMO-based 6G systems.

Resumé

Massive multiple-input multiple-output (MIMO) systemer er en nøgleteknologi til femte generation (5G) trådløse netværk. Derfor skal MIMO-teknologien opgraderes mod den sjette generation (6G) for at drage fordel af de nye banebrydende teknologier inden for artificial intelligence (AI). Krydsfeltet mellem AI og trådløs sensing forventes at være en af de centrale katalysatorer for fremtidige 6G.

Med denne baggrund og motivation undersøger denne afhandling tre emner: large intelligent surface (LIS), millimeter-wave (mmWave) Wi-Fi, og AI-aktiveret ressourceallokering og beam tracking, alt sammen inden for rammerne af MIMO-systemer.

Først udforsker denne afhandling LIS-teknologien til integration af trådløs kommunikation og radio sensing. Vi fastslår, at en LIS, der fungerer som et radiobillede, der beskriver udbredelsesmiljøet, kan give en højopløselig gengivelse af den fysiske verden. Dette muliggør anvendelsen af computer vision (CV) og machine learning (ML) teknikker til at udvikle præcise sensoriske metoder, hvilket understøtter potentialet i LIS-baseret sensing i MIMO-baserede indendørs industrielle miljøer, blandt andre anvendelser diskuteret i afhandlingen.

Dernæst udforsker denne afhandling mmWave Wi-Fi til indendørs trådløs sensing og lægger grundlaget for forbedret objektlokalisering. Ved at foreslå en ny dual-decoder neural dynamic learning framework sigter dette arbejde mod at tackle de unikke udfordringer ved mmWave Wi-Fi lokalisering, herunder intermittensen af beam training measurements og nødvendigheden af at udnytte underliggende object dynamics. Disse fremskridt stemmer overens med 6G-visionen om at tilbyde nøjagtige og effektive lokaliseringssystemer i MIMO-systemer, der kan integreres direkte i Wi-Fi-standarderne.

Endelig udforsker denne afhandling den fælles dynamiske ressourceallokering og beam tracking for mmWave MIMO-systemer, en kritisk opgave for at opnå ultra-reliable low-latency communication (URLLC). Denne del af arbejdet anvender en ramme for proximal policy optimization (PPO) til dynamisk at allokere ressourcer og spore multi-user beams, idet man tager hensyn til den variable kvalitet of trådløse kanaler, og derved forbedrer den samlede systemydelse og reducerer tracking overhead.

Sammenfattende er hovedresultatet af afhandlingen forslaget om radiosensoriske AI-baserede metoder til at dække områder som integreret sensing og kommunikation, ef-

fektive lokaliseringssystemer, og dynamisk ressourceallokering. Denne forskning giver et væsentligt fundament for fremtidige fremskridt i MIMO-baserede 6G-systemer.

Contents

I Introduction 1

Thesis Details

Thesis Title:	Intelligent Sensing and Learning for Advanced MIMO Communication Systems
Ph.D. Student:	Cristian J. Vaca-Rubio
Supervisors:	Prof. Petar Popovski, Aalborg University Prof. Zheng-Hua Tan, Aalborg University

The main body of this thesis consists of the following papers.

[A] Cristian J Vaca-Rubio, Pablo Ramirez-Espinosa, Robin Jess Williams, Kimmo Kansanen, Zheng-Hua Tan, Elisabeth de Carvalho, Petar Popovski, "A primer on large intelligent surface (LIS) for wireless sensing in an industrial setting," *International Conference on Cognitive Radio Oriented Wireless Networks*, pp. 126–138, 2020.

[B] Cristian J Vaca-Rubio, Pablo Ramirez-Espinosa, Kimmo Kansanen, Zheng-Hua Tan, Elisabeth De Carvalho, Petar Popovski, "Assessing wireless sensing potential with large intelligent surfaces," *IEEE Open Journal of the Communications Society*, vol.2, pp. 934–947, 2021.

[C] Cristian J Vaca-Rubio, Pablo Ramirez-Espinosa, Kimmo Kansanen, Zheng-Hua Tan, Elisabeth de Carvalho, "Radio sensing with large intelligent surface for 6G," *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 1-5, 2023.

[D] Cristian J Vaca-Rubio, Dariush Salami, Petar Popovski, Elisabeth de Carvalho, Zheng-Hua Tan, Stephan Sigg, "User localization using RF sensing: A performance comparison between LIS and mmWave radars," *IEEE European Signal Processing Conference (EUSIPCO)*, pp. 1916–1920, 2022.

[E] Cristian J Vaca-Rubio, Roberto Pereira, Xavier Mestre, David Gregoratti, Zheng-Hua Tan, Elisabeth de Carvalho, Petar Popovski, "Floor map reconstruction through radio sensing and learning by a large intelligent surface," *IEEE Workshop on Machine Learning for Signal Processing (MLSP)*, pp. 1–6, 2022.

[F] Herman L Santos, Victor Croisfelt, Cristian J Vaca-Rubio, Taufik Abrão, Petar Popovski, "OnRMap: An online radio mapping approach for large intelligent surfaces," *IEEE International Conference on Communications (ICC)*, 2023.

[G] Cristian J Vaca-Rubio, Pu Wang, Toshiaki Koike-Akino, Ye Wang, Petros Boufounos, Petar Popovski, "mmWave Wi-Fi trajectory estimation with continuous-time neural dynamic learning," *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 1-5, 2023.

[H] Cristian J Vaca-Rubio, Pu Wang, Toshiaki Koike-Akino, Ye Wang, Petros Boufounos, Petar Popovski, "Object trajectory estimation with continuous-time neural dynamic learning of millimeter-wave Wi-Fi," *IEEE Journal of Selected Topics in Signal Processing (JSTSP)*, 2023. **Submitted for publication**

[I] Cristian J Vaca-Rubio, Carles Navarro Manchón, Ramoni Adeogun, Petar Popovski, "Proximal policy optimization for integrated sensing and communication in mmWave systems," *IEEE Wireless Communication Letters (WCL)*, 2023. **Submitted for publication**

In addition to the main papers, the following publications have also been made.

[1] Anay Ajit Deshpande, Cristian J Vaca-Rubio, Salman Mohebi, Dariush Salami, Elisabeth De Carvalho, Petar Popovski, Stephan Sigg, Michele Zorzi, Andrea Zanella, "Energy-Efficient Design for RIS-assisted UAV communications in beyond-5G Networks," *IEEE Mediterranean Communication and Computer Networking Conference (MedComNet)*, pp. 158–165, 2022.

[2] Roberto Pereira, Anay Ajit Deshpande, Cristian J. Vaca-Rubio, Xavier Mestre, Andrea Zanella, David Gregoratti, Elisabeth de Carvalho, Petar Popovski, "User Clustering for Rate Splitting using Machine Learning," *IEEE European Signal Processing Conference (EUSIPCO)*, pp. 722-726, 2022.

[3] Herman L dos Santos, Cristian J Vaca-Rubio, Radosław Kotaba, Yi Song, Taufik Abrão, Petar Popovski, "EMF Exposure Mitigation in RIS-Assisted Multi-Beam Communications," *IEEE Global Communications Conference (GLOBECOM)*, 2023. **Submitted for publication**

[4] Pu Wang, Cristian J Vaca-Rubio, Toshiaki Koike-Akino, Ye Wang, Petros Boufounos, "System and method for sensing a state of a device with continuous-time dynamics," 2023. **US Patent Application**

This thesis has been submitted for assessment in partial fulfillment of the PhD degree. The thesis is based on the submitted or published scientific papers which are listed above. Parts of the papers are used directly or indirectly in the extended summary of the thesis. As part of the assessment, co-author statements have been made available to the assessment committee and are also available at the Faculty. The thesis is not in its present form acceptable for open publication but only in limited and closed circulation as copyright may not be ensured.

Acknowledgements

First and foremost, I would like to express my honest gratitude to my supervisors, Prof. Petar Popovski and Prof. Zheng-Hua Tan. Their unwavering support and expert guidance were instrumental throughout my Ph.D. journey. Additionally, I am thankful to Prof. Petar Popovski for welcoming me into his group, where I gained experiences that went beyond the realm of research.

I would also like to extend my heartfelt thanks to Dr. Pablo Ramírez Espinosa, whose support at various stages of my studies was invaluable, particularly during my adaptation to what was initially a new field. Moreover, my appreciation extends to Prof. Kimmo Kansanen and Prof. Elisabeth de Carvalho, who offered immense support during the early stages of my Ph.D. journey.

I was fortunate enough during my Ph.D. to visit Mitsubishi Electric Research Laboratories (MERL). There, I was introduced to a vibrant interdisciplinary environment where I had the chance to engage with renowned researchers in the field and develop new skills. I am grateful to the MERL group for their hospitality, the enlightening discussions, and the invaluable feedback that led to high-quality publication outputs, some of which were even recognized at ICASSP2023. I would like to extend special thanks to Dr. Pu (Perry) Wang, who supervised my visit and provided excellent training.

My heartfelt appreciation goes to Ana García Armada, Ioannis Krikidis, and Troels B. Sørensen for their willingness to serve on my Ph.D. defense committee. I am also grateful to all the members of the Connectivity group at Aalborg University for their company during these years.

I cannot forget to thank the friends I made in Denmark - Victor, Igor, Isabel, and Alessandro - who provided support throughout this process.

Finally, my deepest gratitude goes to my mother, father, and sister, whose constant support has been my backbone throughout these years. Their encouragement made this journey smoother and more bearable.

Cristian Jesús Vaca Rubio
Aalborg University, August 17, 2023

Part I

Introduction

Chapter 1

Introduction

1 Background and Motivation

The field of wireless communication is the one undergoing a major breakthrough. The enhancements in computing capabilities, the integration of artificial intelligence (AI), the exploitation of new frequency bands and the emergence of new cellular technologies are key enablers for this technological evolution. It once started by the development of the fifth generation (5G) [1] of cellular networks, and it is now at the beginning of a new phase with the emergence of the sixth generation (6G), which is promised to transcend the paradigms of wireless communication [2]. In this thesis, we study some of the cores of these advancements taking a look at massive multiple-input-multiple-output (mMIMO), large intelligent surfaces (LIS), AI, wireless sensing, joint dynamic beam tracking and resource allocation and millimeter-wave (mmWave) Wi-Fi sensing.

The 5G mobile network is the latest-implemented cellular technology, being the United States (US) the country with the most cities supporting this technology in 2023 [3]. This generation made a significant step towards a fully connected world. It was designed to support a vast amount of devices while offering enhanced broadband capabilities and extremely reliable, low-latency communications [4]. The main purpose of 5G is not only to lead to faster and more reliable connectivity, but also to provide the necessary infrastructure to pave the way for the next era of digital innovation [5], including the internet of things (IoT), autonomous vehicles, smart cities, and much more yet to discover [6]. One of the main advantages of 5G is its significant increase in data transmission, leading to speeds up to 100 times faster than the older fourth generation (4G) [7]. These speeds allow brand-new changes, such as real-time streaming in ultra-high-definition (UHD) and significantly faster download/upload speeds for all type of traffics. What is more, 5G goals also entail extremely low latency [7]. This means that 5G can support real-time control applications, such as remote surgeries in the healthcare sector, real-time control of industrial robots, and safer autonomous vehicles. Besides, another major characteristic of 5G is being able to handle many more devices simultaneously. This is necessary for the broad-scale implementation of IoT devices [8], ranging from industrial automation to smart home systems. Furthermore, 5G networks were designed to be more flexible and efficient enabling a wide range of new applications. This is in part due to the developments on mMIMO [7] technology, where providing base stations (BS) with a large number of antennas enabled further increase in spectral efficiency [9]. This was one of the biggest enablers of the three main features of 5G [10, 11]. Also, mMIMO is more energy efficient [12]. Because of the fact of using that many antennas, each of them transmits with a relatively low amount of energy. This leads to an enhacement in energy efficiency in terms of transmit power. However, the power required to drive the circuits is an important consideration. Besides, mMIMO contributes to latency reduction thanks to its ability to serve multiple users at the same time. This is a crucial feature for remote surgery, real-time gaming, etc, where a minimal delay can have a significant impact on the performance. In addition, mMIMO enhances the reliability of wireless networks, by enabling highly directive beamforming techniques, which reduces interference and provides directivity towards the targeted user. This leads to more reliable and robust transmissions, even in challenging environments.

As we move towards the 6G era, it is anticipated an unprecedented transformation of wireless networks. It will go beyond the bounds of being just designed as a communication platform, but to become smart/intelligent systems that possess cognitive abilities [13]. Although the exact specifications of 6G are still under development, it is anticipated to offer data rates over 100 Gbps, latency levels below one millisecond, and immense improvements in energy and spectral efficiency [13]. This forthcoming generation is aiming to redefine our digital prospects, leading to a world that is even more connected and smart.

Wireless sensing is envisioned to be one of the main enablers of 6G [14, 15]. It can be defined as the usage of wireless signals to perceive, identify and understand the environment. Fundamentally, it works by interpreting changes in the wireless signal's characteristics, i.e., signal strength, phase, frequency, etc [16]. These changes occur because of the interaction of the signals with objects, people, movement in the environment, or changes in environmental conditions. In theory, wireless sensing devices can detect these changes and use sophisticated algorithms to gain valuable insights of the physical world. These days, we live in an era in which wireless signals are ubiquitous, such as Wi-Fi, cellular or Bluetooth signals. This means that in many cases, wireless sensing can be implemented using existing wireless infrastructure, and integrate this functionality as an add-on seamlessly. This is one of the aspects we will address in the thesis, specifically for mmWave Wi-Fi.

On a related note, LIS is regarded as a natural extension of mMIMO that designates a large ideally continuous electromagnetic surface capable of transmitting and receiving radio waves [17, 18]. In practice, LIS is composed of a collection of closed-spaced tiny antenna elements, embedded in a planar surface, with signal processing capabilities. The potential of LIS in communications has received considerable attention [18, 19] in the research community, along with its applications in high directive beamforming [20], and positioning systems [21, 22]. However, in this thesis, we will leverage its radio image-sensing capabilities and provide insights into its applicability in future 6G systems.

Simultaneously, the field of mmWave Wi-Fi sensing is emerging, leading to promising advancements for indoor localization and object detection [23, 24]. By taking advantage of high-frequency mmWave signals, these systems can provide high-resolution sensing capabilities that will allow obtaining accurate environmental information and a detailed understanding of the environment dynamics. This opens a vast range of opportunities, from enhancing the accuracy of indoor navigation systems to developing state-of-the-art joint communication and sensing systems easily integrated into daily lives.

Alongside the promises of the 6G new era, AI plays a vital role in these advancements [25] that will take advantage of the huge amount of data generated in the networks. It is promised to act as the binding glue that will hold all these advancements together. By leveraging the usage of machine learning (ML) algorithms, AI can analyze complex patterns, turning raw data into insightful features. This will transform the new generation into intelligent and adaptive systems. AI can be applied to many different aspects of wireless communications, but in the context of this thesis, it is explored its capability to exploit signal processing for wireless sensing. This enables a further understanding of the physical world that can be later on used for new applications and communication performance enhancement. In summary, by following Huawei´s 6G vision [26] this thesis focuses on the intersection between *Native AI* and *Networked Sensing*. More specifically, this thesis covers AI-based sensing in three areas depicted in Figure 1.1.

As highlighted above, the intersection between wireless sensing and AI is one of the key enablers for future 6G MIMO systems. Furthermore, in this thesis, we aim to pave the way toward the first applications of beyond-massive MIMO sensing, through the study and analysis of the feasibility of radio image-based LIS for this task. We believe its well-demonstrated communication capabilities go beyond being a pure communication system, and we intend to show the first steps for developing its sensing capabilities as a potential device for integrated sensing and communication. Similarly, wireless sensing is not limited to massive MIMO scenarios and we also study its feasibility in MIMO commercial off-the-shelf devices based on mmWave Wi-Fi. This part of the research will show the potential of AI in developing sensing solutions easily integrable into already existing infrastructures and communication standards. We also explore the potential of AI for joint beam tracking and resource allocation in mmWave MIMO systems, enabling strategies to optimize both network performance and signaling overhead. Hence, we aim to devise the necessary AI-based signal processing algorithms to make the first steps into smart sensing systems. In short, we try to leverage the sensing capabilities of LIS, tackle the difficulties of mmWave Wi-Fi sensing, and provide solutions for efficient

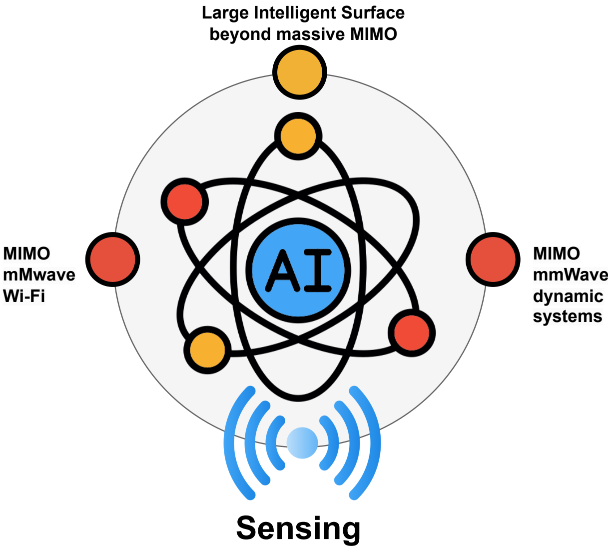


Fig. 1.1: Intersection of areas covered in this thesis.

resource allocation and beam tracking with our methods. A detailed description of the challenges and research questions is presented in the next chapter.

2 Structure of the thesis

The rest of the thesis is organized as follows: First, we formulate the problems and research questions this thesis is targeting to answer. Second, we follow by providing a review of the related literature for wireless sensing in different contexts as well as AI applications to wireless sensing. Next, we categorize the contribution of the thesis into different groups. For all the groups, we explain the specific contributions within the wireless sensing paradigm and present a summary of our papers. We conclude the thesis by discussing the main findings and outlining the directions for future research.

Chapter 2

Problem statement

This chapter identifies the advantages and challenges of LIS-based, mmWave Wi-Fi, and dynamic sensing systems. Next, we address each of these challenges with a research question. Then the proposed methodologies are introduced to perform the evaluation along with their comparison methods in the literature.

1 Challenges

Before going into the challenges, we give a clear definition of an LIS. In the literature, different terminologies can be used to define the same concept such as active intelligent surface (AIS), active metasurface, smart radiating surface, and digital large-scale antenna system (DLAS). In this thesis, we will use the term LIS. An LIS is a planar surface composed of many antenna elements closely spaced. Ideally, it is a continuous surface that can transmit and receive radio waves. Different to the majority of the works of the literature related to LIS, in this thesis, we do not focus on making a theoretical or asymptotic analysis but provide practical AI-based signal processing methods. What is more, we place a greater emphasis on assessing the sensing capabilities of LIS rather than focusing extensively on their communications, which have already been thoroughly studied in the existing literature. For instance, we do not evaluate its high directive beamforming capabilities, but instead, we aim to provide brand-new radio image-based sensing methods that can be key enablers for future applications. To make sure of its closeness to reality in modeling sensing behavior, for the LIS works we make use of ray tracing to ensure a reliable/realistic multipath propagation environment.

As mentioned before, in this thesis we focus on LIS-based sensing, mmWave Wi-Fi sensing, and dynamic joint resource allocation and beam tracking. The first focuses on beyond massive MIMO with LIS and the two remaining on mmWave MIMO systems in both experimental and theoretical setups. The main concerns over these three points can be grouped into the following:

1. **Feasibility of radio-image based sensing methods:**

   LIS image-based sensing systems can be susceptible to interference from other wireless devices operating in the same frequency band. Additionally, noise from the environment or system components can affect the quality and reliability of the sensed images. Advanced AI-empowered signal processing techniques are necessary to address these challenges.

2. **Intermittent sampling in mmWave Wi-Fi measurements:**

   Commercial mmWave MIMO Wi-Fi routers work under the necessity of a well-established beam sweeping and communication protocol for the 60 GHz frequency. Reusing ubiquitous already-deployed wireless communication signals can be beneficial for integrating sensing functionalities seamlessly. Within this context, we are interested in providing sensing-based localization methods. However, this presents a main challenge: the intermittent sampling of beam training measurements due to beam scanning overhead and contention over the wireless channel.

3. **Efficient resource allocation for integrated sensing and communications:**

   High-frequency mmWave MIMO systems need to be highly adaptive due to the fast varying channel conditions. We focus on 28 GHz. Usual protocols rely on fixed periodicity to allocate slots for beam tracking and communication operations, leading to highly inefficient and less reliable resource allocation policies.

2 Research questions (RQs)

RQ1: Can LIS provide a high-resolution image of the environment describing the physical world? How can such a method be developed? What aspects of LIS are beneficial for sensing performance? How accurate would a radio sensing method be for different applications?

RQ2: What are the consequences of intermittently sampled beam training measurements for indoor localization? Can this feature of mmWave Wi-Fi systems be learned to provide highly-accurate indoor localization solutions?

RQ3: Is there a way to improve mmWave network performance by dynamic resource allocation and beam tracking? In other words, can we learn a dynamic joint resource allocation and beam tracking policy to enhance the reliability and efficiency of mmWave MIMO systems?

Now, we investigate each of the questions and provide a short description of our approaches to tackle them.

As previously stated, LIS has been proven to be a powerful communication device in the literature. However, we focus on analyzing its sensing feasibility. We aim to develop radio sensing methods that take into account the propagation environment to describe the physical scenario. Nowadays, indoor scenarios are fully covered by communication signals (Wi-Fi APs, cell phones, laptops, etc). These signals travel all over the air impinging into obstacles, scatters, and people. These reflections will be processed to develop radio-image-based sensing methods from the LIS viewpoint. Concretely, we target to provide radio sensing methods that make use of already existing communications signals to provide an image of the environment. For this purpose, we analyze all the main features of LIS (antenna spacing, aperture size, amount of antennas) to assess its impact on radio sensing performance. What is more, we propose mainly two radio-image-based sensing methods that we denominate as (1) Holographic sensing, and (2) Matched Filtered sensing. The former relies just on received signal power at the LIS, while the latter relies on the variations of the signal phase all over the LIS. We then leverage their feasibility in some applications using the proposed approaches. We are providing a description of each of these methods in detail in Chapter 3.

Related to the second question, it is worth noting that the integration of sensing solutions into already established communication protocols is a challenging task. In this thesis, we aim to provide a method to further exploit mmWave Wi-Fi beam training measurements for localization purposes. These measurements are usually irregular and sparsed in time. This indeed adds difficulty to the problem, because using sequences-based AI solutions like recurrent neural networks (RNN) will not result in the best performance due to the assumption of regularly-sampled data. Also, using frame-based solutions (meaning 1-to-1 mapping between a beam training measurement and a localization coordinate) will not exploit the dynamics of beam measurements to assist the localization task. Therefore, there is a need for a smarter framework that can tackle the irregularity and sparsity of beam training measurement data. In this way, in this thesis, we propose what we denominate as a dual-decoder neural dynamic learning framework. More details are provided in Chapter 3.

Last but not least, fixed-slots allocation for beam tracking (sensing) and communications are not able to tackle the fast varying channels of mmWave systems. For example, if a channel varies slowly due to the stationarity of the user or it varies faster due to being a high-speed one (for instance a vehicle), fixed-slot allocations are not suitable and it might lead to high packet error rates (PER) or low frame slots distribution efficiency. Thus, providing a new policy design for dynamic beam tracking and resource allocation is a key enabler in future mmWave systems. Chapter 3 provides more details of our proposed method.

3 Methodology

Throughout this thesis, different methodologies have been employed to address the presented research questions. All the methods developed are implemented using computer simulations in the Python programming language.

For the LIS-based sensing, we relied on ray tracing software to perform the signal propagation environment characteristics. In this way, we can ensure a reliable behavior of the multipath components that later on we will use to perform sensing. Our performance assessment is then based on channels generated by the multipath contributions using the ray tracing software. The main sources of randomness in these simulations are the additive Gaussian noise and the positions of the active transmitters in the scenario, which act as the signal sources. As we are the first in the literature to tackle radio-image-based sensing with LIS, we leverage the performance of these methods by analyzing the impact of different parameters of LIS, as well as proposing new applications. Also, we compare the performance of this system in a localization task against mmWave radars.

For the mmWave Wi-Fi sensing we make use of experimental data that was collected during the internship at Mitsubishi Electric Research Laboratories (MERL). We created a mmWave Wi-Fi testbed. This testbed uses off-the-shelf 802.11ad devices to gather real-world mmWave Wi-Fi beam SNR data. For this, we used a pair of TP-Link AD7200 routers as an access point (AP) and a mobile user, which were housed on a stationary post and a mobile TurtleBot, respectively. The TurtleBot is fitted with a 2D LiDAR sensor and a wheel encoder, which helps to map the environment and place itself on a 2D floorplan with an accuracy of under 1 cm. We used these 2D localization results as the training labels and testing ground truth. The TP-Link AD7200 router used an analog phased array of 32 antenna elements to sequentially scan over 36 predefined directional beampatterns for each air time of a given responder. The data was collected by placing the pair of TP-Link AD7200 routers in a corner conference room. The AP router was kept stationary while the TurtleBot moved along a pre-set rectangular trajectory. Over two days, we gathered two separate data sessions that lasted several hours each. In order to compare the performance of our proposed method, we have implemented several benchmark techniques used in the Wi-Fi sensing literature. Concretely, we implement a support vector regressor (SVR), a fully connected neural network regressor (FCNNR), and two variations of RNN to handle irregularly-sampled data: RNN Δt and RNN Exponential Decay.

Finally, for the dynamic beam tracking and resource allocation we develop an environment simulated in Python. Here, the main sources of randomness are the velocity and direction of the moving users, the additive Gaussian noise, and the complex fading. In order to compare our proposed method, we implement a range of baselines based on variations of Time Division Multiple Access (TDMA) that relate to the current implementation of fixed-slots allocation in the 5G standard.

3.1 Main assumptions

In this subsection, we mention the main assumptions made throughout the thesis.

For the ray tracing simulations, we consider narrowband channels. We consider synchronous TDD cases where there is channel reciprocity in uplink and downlink. For the LIS-based system, we assume its aperture covers the entire dimension of a wall or a ceiling. Besides, we assume an RF chain is connected to every antenna element. Also, due to the dimensions and the proximity to the sensing targets, we assume a near-field propagation condition.

For the mmWave Wi-Fi experimental setup, we do not make any specific assumptions, as we are using real-world data. During the data collection phase, we ensure Line-of-Sight (LoS) between the APs.

Finally, for the dynamic sensing and communication mmWave MIMO system, we assume a geometrical LoS channel model and a narrowband channel. We also assume a codebook-based analog beamforming architecture with a single RF chain at the BS.

Chapter 3

Background and Contributions

In this chapter, we first provide a review of the relevant state-of-the-art (SoA) in relation to the specific areas covered in this thesis. We then categorize the areas of contribution of the thesis into 3 groups related to every of the research questions we try to address. Then we provide an explanation of the designed methods and their purpose to solve the presented research questions and go beyond the challenges. We conclude each group's discussion by providing brief overviews of their respective published papers.

1 State-of-the-art

Wireless sensing strategies based on radio signals have been thoroughly studied in the literature in different ways and applied to a wide range of applications. In the following subsections, we provide some review of the SoA grouped around the three RQs we were trying to address.

1.1 Radio map imaging

Radio map imaging can be described as the ability to use wireless signals to describe the physical environment as an image. We provide a summary of the most relevant literature below.

Initially, radio tomographic imaging (RTI) emerged as one of the earliest technologies capable of creating environmental radio maps. RTI utilizes wireless signals to pinpoint and track physical objects within a wireless sensor network (WSN), earning significant attention due to its capacity for passive detection of users—without requiring them to act as transmitters or receivers. [27]. Born from the principles of radar systems, RTI found widespread application in diverse fields like indoor factories, surveillance, and through-wall imaging [27, 28]. Essentially, RTI operates by measuring RF signals' attenuation between paired nodes in a WSN [29], which illustrates the absorption of RF waves at each location along their propagation path [30]. By analyzing these attenuations, RTI is able to construct radio maps that effectively detect the presence and location of objects within the WSN area [31].

To enhance the precision and robustness of RTI, numerous improvements have been implemented in the field. For example, sparse Bayesian learning (SBL) was integrated into the RTI process to augment reconstruction accuracy in scenarios featuring multipath fading [32]. Additionally, the deployment of RTI within factory environments became increasingly attractive due to the rigorous challenges presented by severe multipath environments [28].

In parallel, radar-like methodologies experienced considerable advancements, leading to the development of applications like gesture recognition [34], object classification [35], and indoor positioning. These applications exploit radar signals to extract invaluable information, offering reliable and robust results. Along the same line, radar-based simultaneous localization and mapping (SLAM) were developed. These systems combine radar with other sensors like cameras and LiDAR to achieve superior performance across all weather conditions [36]. High-resolution imaging utilizing virtual array processing was also demonstrated with the usage of mmWave MIMO radar incorporating non-uniform planar arrays for both transmitters and receivers [37].

Lastly, channel state information (CSI) radio map techniques [38] have gained recognition in the literature. A pioneering framework for channel charting-aided localization in mmWave networks has been introduced [39], which employs a convolutional autoencoder to estimate user localization based on multipath CSI. This framework, designed around unsupervised learning, enables the construction of a radio-geometry map, showcasing its applicability for beam prediction tasks [39].

1.2 Wi-Fi sensing

Wi-Fi sensing leverages the ubiquity of Wi-Fi signals to sense and interpret the physical and environmental changes. Researchers have paid attention to this field, as it means we can rely on already existing communication infrastructures for sensing applications. Wi-Fi sensing has been applied to a wide range of problems such as intrusion detection [41], activity recognition [42, 43], and vital signs or gesture recognition [44, 45]. However, we will focus on highlighting the milestones in localization tasks.

Wi-Fi for localization and tracking has been a topic of extensive research. Predominantly, Wi-Fi-based indoor localization methods are based on CSI from the physical layer or Received Signal Strength Indicator (RSSI) measurements from the Medium Access Control (MAC) layer for both fingerprinting and direct localization. Fingerprinting-based localization consists of a procedure having two phases: offline (training) and online (localization) phase. In the offline phase, a collection of measurements is collected to generate a database of wireless features at known locations of the environment, followed by an online (localization) phase where the location is estimated by comparing it to the previously gathered database. Its main advantage is the ability to handle complex environments. Conversely, direct localization methods typically use geometric techniques such as triangulation to determine the location of the target based on certain measurable parameters. These parameters may include time of arrival (ToA), angle of arrival (AoA), or RSSI. This leads to simple and straightforward implementations, however, its accuracy highly depends on the measurement quality, which can be easily affected.

The first steps toward Wi-Fi indoor localization relied on RSSI in a direct localization fashion [46–48]. Subsequently, classical ML solutions such as the k-nearest neighbor (kNN) and SVM were applied to RSSI fingerprinting measurements [49–52]. However, they were not really suitable and robust to the environment. More advanced ML solutions were applied, for instance, deep learning [53, 54] to exploit more complex and robust features for indoor localization. Furthermore, RNN-based RSSI sequence modeling was also taken into account [55, 56], proving more robust to environmental changes by capturing temporal patterns and fluctuations in the RSSI caused by the multipath effects.

As mentioned before, CSI fingerprinting is also a well-established approach. With CSI, richer information can be obtained about the environment. This allows us to train more advanced deep learning models to learn a mapping between CSI and location. For example, the authors in [57, 58] used convolutional neural networks (CNN) to perform localization. Also, some methods were directly used to feed CSI and regress the coordinate estimation [59, 60].

In line with current trends, there has been a significant effort to improve the capabilities of Wi-Fi signals. For instance, the introduction of IEEE 802.11bf standard aims to enable high-resolution mmWave Wi-Fi sensing [61]. This advancement opens up new possibilities for applications such as wireless LAN sensing and enhances the overall performance and functionality of Wi-Fi networks. Reusing mid-grained mmWave training measurements have been explored recently in the literature [62–67]. This consists in essentially using the signal-to-noise-ratio (SNR) values of the possible beam patterns during the beam training phase in mmWave Wi-Fi for indoor localization. This is of great interest due to its high applicability to current standards, as it is used in already-implemented communication protocols.

1.3 Resource allocation and sensing

Sensing can be seen as well as the beam alignment/tracking procedure in mmWave systems. Resource allocation and sensing are then critical for improving network performance and efficiency.

Beam tracking and alignment are key features of mmWave technology. The fact of using narrow beams requires a good alignment between transmitter and receiver, which is difficult due to the mobility of the user and the high-frequencies involved. The beam tracking goal is to adapt continuously the beam direction to ensure an optimal alignment. Some techniques rely on Kalman filters [68] and particle filters [69] to mention a few. Furthermore, reinforcement learning (RL) [70] methods have also been explored in the literature. For instance, the works [71, 72] solve a beam alignment problem using RL frameworks in different contexts.

On the other hand, the resource allocation goal is to optimize the distribution of resources among users to maximize overall performance. This has been extensively addressed in the literature, especially in the context of 5G. The techniques used range from classical methods like proportional fair scheduling [73], to sophisticated optimization methods such as such as convex optimization [74], game theory [75], and reinforcement learning [70]. Deep RL has also been applied to deal with the complexity and dynamic nature of the resource allocation problem [76–79].

The integration of beam tracking and resource allocation tasks is an exciting and challenging research direction. Joint optimization of these can significantly achieve an improvement in performance [80]. In this way, more efficient policies can be designed that take into account not only channel quality but also user data requirements. Some recent works have proposed deep learning solutions to solve this joint optimization problem [81, 82].

2 Grouping of the contribution areas

In order to group the contribution of the thesis, we start by discussing briefly the aim of the thesis. As mentioned in Chapter 2, the focus of the thesis is to answer 3 research questions. These research questions concern providing robust radio map sensing methods with LIS, tackling the challenges of intermittent sampling in beam training measurements for mmWave Wi-Fi localization, and tackling dynamic resource allocation and sensing in mmWave systems.

In this way, the objective is to come up with AI-based methods to tackle the aforementioned challenges.

Furthermore, we classify the contributions of the thesis into 3 groups, related to the type of problems we solved and the methods we exploit in each of them to answer the RQs.

Fig. 3.1 shows a schematic of the areas and the papers within each of the categories. The first group discusses the methods for radio map-based LIS sensing as well as evaluates different parameters of LIS for sensing performance, leverages some applications, and provides a comparison with mmWave radars. In this way, the first block focuses on RQ1. The second group focuses on RQ2 and provides a neural dynamic method to tackle intermittent sampling for localization using beam training measurements. Finally, the last group pays attention to the problem of joint resource allocation and sensing, trying to address RQ3.

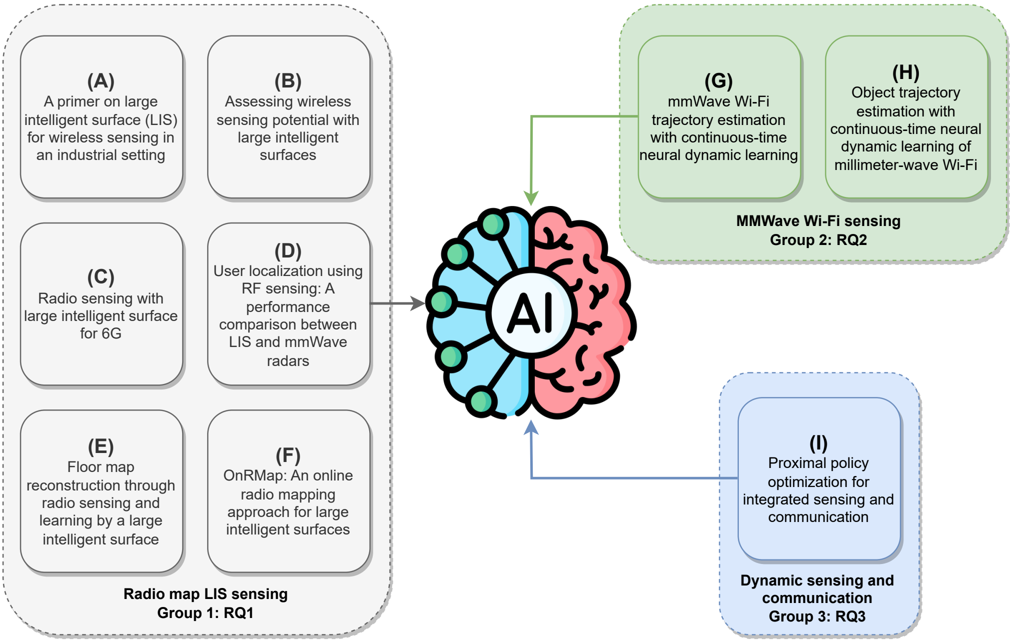


Fig. 3.1: Schematic representation of the contribution areas and the papers in each of the groups.

In the next sections, we study each of the groups in detail providing an explanation of the developed radio map methods, relating them to the SoA discussed in the previous section, and finally providing a summary for each paper and how it tries to address their corresponding RQ.

3 Group 1: Radio map LIS sensing

This group focuses on RQ1. Several radio map LIS methods are proposed to address the existing challenges. The main idea is to re-utilize the signals already occurring in the environment from emitter sources, and also take advantage of the reflections all over the people and scatters in the room to provide radio maps that describe the physical environment. We begin this part with a short description of the proposed radio map methods. Next, we introduce the papers in this group along with a short summary, where we highlight the role of AI using these radio maps.

3.1 Holographic radio map

For this method, we analyze for the first time in the literature the ability of LIS to create a radio map of the environment. Let us assume we have a transmitter entity in an indoor room that is moving along a trajectory.

In order to create a holographic radio map, we assume that a LIS is compounded by M antenna elements.

The complex baseband signal received at the LIS is then given by

$$\mathbf{y} = \mathbf{h}x + n, \tag{3.1}$$

with x the transmitted (sensing) symbol, $\mathbf{h} \in \mathbb{C}^{M\times 1}$ the channel vector to each antenna-element, and $\mathbf{n} \sim \mathcal{CN}_M(\mathbf{0}, \sigma^2\mathbf{I}_M)$ the noise vector. Moreover, we consider a static scenario where the channel $\mathbf{h}$ only depends on the user position, neglecting the impact of time variations.

To minimize deployment expenses while capturing significant variations, we focus on leveraging the received signal amplitude (or power) as a sensing metric. By adopting this approach, we can potentially achieve more cost-effective and streamlined system implementations, negating the need for intricate coherent detection techniques. In this way, the radio map will be given by $\|\mathbf{y}\|^2$.

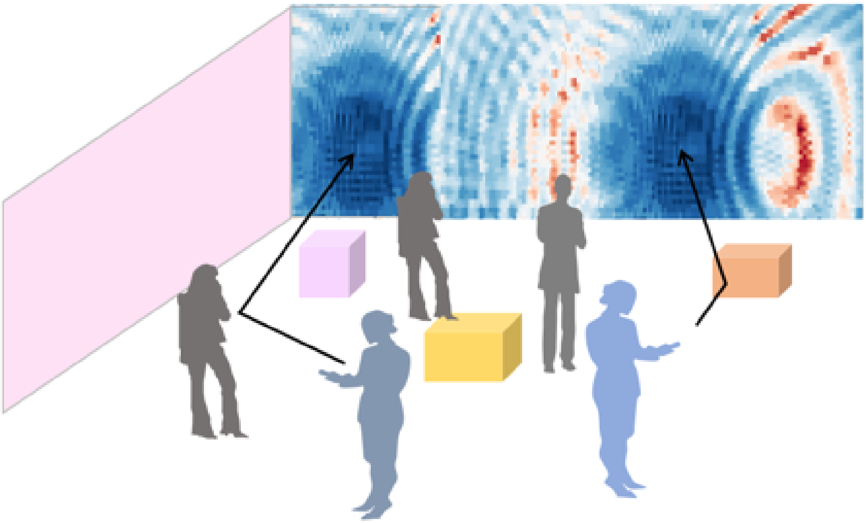


Fig. 3.2: Hologprahic sensing for an indoor environment.

Furthermore, this method works by measuring the superimposed multipath wireless signals in the environment at the LIS viewpoint in every of its antenna elements. The power of this received signal is exploited to generate a high-resolution image of the propagation environment. Fundamentally, the LIS uses envelope detectors to measure the CSI, not needing any kind of phase estimation. This method significantly simplifies the nature of multipath propagation by converting the information into an image. The holographic radio map method exhibits a dual advantage: i) the huge amount of elements that conform the LIS provided highly accurate sensing, and ii) it enables the application of CV and ML strategies to handle the resulting images.

To conclude, in comparison with the literature, this method eliminates the need of having a calibration phase and there is no requirement to model scatterers for them to be represented in the radio maps. This is a huge advantage to methods of radio map reconstruction based on signal power as RTI (Section 1.1). Figure 3.2 shows an exemplary scenario. The wireless signals are being propagated in the environment and they are reflecting from the scatterers and people. Then, the wall LIS measures the received signal power to compute the radio map. More details of these maps can be found in papers A, and B.

3.2 MF-based radio map

Assume an indoor setting where we have D_c active devices randomly located within a room carrying out their communication activities. For the sake of simplicity, we consider that all D_c devices transmit at the same frequency, which could be representative of Wi-Fi or cellular band transmissions.

To conduct the sensing, we place Large Intelligent Surface (LIS) made up of M antenna elements installed across the ceiling, with its physical aperture covering the entire area.

The combined complex baseband signal received at the LIS can be described as:

$$\mathbf{y} = \sum_{d=1}^{D_c} \mathbf{h_d} x_d + \mathbf{w}, \tag{3.2}$$

where x_d denotes the transmitted (sensing) symbol from device d, $\mathbf{h_d} \in \mathbb{C}^{M\times 1}$ signifies the channel vector from a particular position of device d to each antenna element, and $\mathbf{w} \sim \mathcal{CN}_M(\mathbf{0}, \sigma^2 \mathbf{I}_M)$ represents the noise vector. Please note that a narrowband transmission is being considered here, thereby evading frequency selectivity effects.

Because of the large physical aperture of the used LIS in comparison with the distance existing between the transmitters and the LIS, we assume spherical way propagation conditions. Due to this, the channel coefficient $h_{s,i}$ at the LIS i-th element is proportional to:

$$h_{s,i} \;\propto\; \frac{1}{d_i} e^{-j\frac{2\pi}{\lambda} d_i}, \tag{3.3}$$

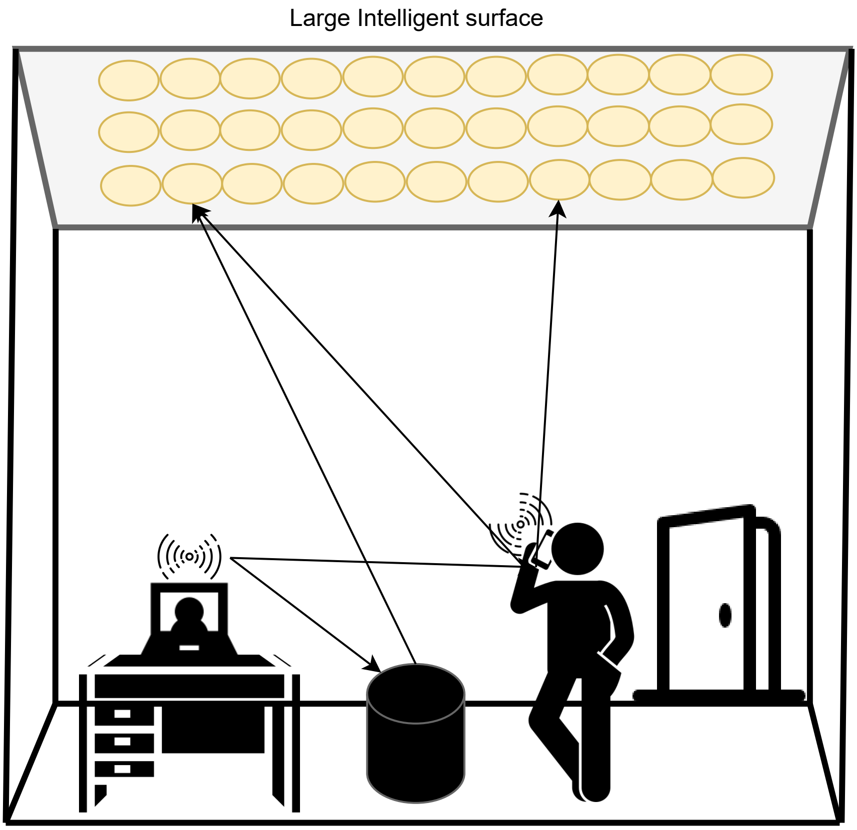


Fig. 3.3: MF sensing for an indoor environment.

where $d_i = \sqrt{(x_i - x_d)^2 + (y_i - y_d)^2 + (z_i - z_d)^2}$ denotes the distance between the active device d w.r.t the i-th antenna, and λ is the wavelegth. In this way, we can compute this operation along all the antenna elements from a designed virtual array, obtaining $\mathbf{h}_{MF}$. This gives us the expected pattern along the LIS with respect to the phase variations. Finally, describing the LIS in a vectorized notation, we can derive an MF such that:

$$\mathbf{y}_{MF} = \mathbf{h}_{MF} * \mathbf{y}, \tag{3.4}$$

where $*$ denotes the convolution operator. To obtain a radio map, we just need to compute the energy at the output of the MF procedure $|\mathbf{y}_{MF}| \in \mathbb{R}^{M\times 1}$. In this map, the active transmitters will be easily detected while the scatters and people where signals reflect from will act as virtual sources that will also be captured by the radio maps.

Figure 3.3 shows an exemplary scenario. The LIS deployed in the ceiling measures the received signal and performs the MF procedure to obtain a radio map of the environment describing the physical phenomena. More details of these maps can be found in papers C, D, E, and F.

3.3 Paper summaries

This group contains 6 papers:

Paper A: (*published*) Cristian J Vaca-Rubio, Pablo Ramirez-Espinosa, Robin Jess Williams, Kimmo Kansanen, Zheng-Hua Tan, Elisabeth de Carvalho, Petar Popovski, "A primer on large intelligent surface (LIS) for wireless sensing in an industrial setting," *International Conference on Cognitive Radio Oriented Wireless Networks*, pp. 126–138, 2020.

Paper B: (*published*) Cristian J Vaca-Rubio, Pablo Ramirez-Espinosa, Kimmo Kansanen, Zheng-Hua Tan, Elisabeth De Carvalho, Petar Popovski, "Assessing wireless sensing potential with large intelligent surfaces," *IEEE Open Journal of the Communications Society*, vol.2, pp. 934–947, 2021.

Paper C: (*published*) Cristian J Vaca-Rubio, Pablo Ramirez-Espinosa, Kimmo Kansanen, Zheng-Hua Tan, Elisabeth de Carvalho, "Radio sensing with large intelligent surface for 6G," *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 1-5, 2023.

Paper D: (*published*) Cristian J Vaca-Rubio, Dariush Salami, Petar Popovski, Elisabeth de Carvalho, Zheng-Hua Tan, Stephan Sigg, "User localization using RF sensing: A performance comparison between LIS and mmWave radars," *IEEE European Signal Processing Conference (EUSIPCO)*, pp. 1916-1920, 2022.

Paper E: (*published*) Cristian J Vaca-Rubio, Roberto Pereira, Xavier Mestre, David Gregoratti, Zheng-Hua Tan, Elisabeth de Carvalho, Petar Popovski, "Floor map reconstruction through radio sensing and learning by a large intelligent surface," *IEEE Workshop on Machine Learning for Signal Processing (MLSP)*, pp. 1-6, 2022.

Paper F: (*accepted*) Herman L Santos, Victor Croisfelt, Cristian J Vaca-Rubio, Taufik Abrão, Petar Popovski, "OnRMap: An online radio mapping approach for large intelligent surfaces," *IEEE International Conference on Communications (ICC)*, 2023.

Paper A is the first one that devises the method of holographic sensing. It is the first paper in the literature that uses LIS for radio mapping technology. In this work, we make the first steps toward LIS radio sensing and propose an industrial problem to leverage its applicability. We aim to detect the deviation of an industrial robot over a predefined route using a LIS radio map. The robot transmits a constant sensing signal. AI arises as a power solution to exploit radio sensing as an image, by exploiting computer vision methods and SVM for the detection of robot deviation over the trajectory. Concretely, from the CV literature, we reuse the pre-trained VGG19 network [83] as a backbone to extract meaningful features from the radio map that will be later on fed to train our SVM solution. Here, we are showing the advantage of obtaining the propagation environment as an image: we can exploit the advances in CV. We analyze the impact of different features of LIS on the sensing performance such as signal sampling, antenna spacing, and aperture. This comprehensive analysis underscores the transformative potential of LIS in reshaping our understanding and utilization of radio-sensing technology.

Paper B tries to assess the potential of LIS for wireless sensing in an industrial setting using a holographic radio map. We recover a similar scenario to paper A, and we try to detect the deviation of an industrial robot over a set of predefined routes. We derive a Generalized Likelihood Ratio Test (GLRT) to benchmark our AI solution. This test, works under the assumption of noise statistics, while the proposed AI method does not need any a priori assumptions. Again, we re-use the VGG19 as a feature extractor. We then train an unsupervised algorithm Local-Outlier-Factor (LOF) to perform the detection task. Besides, we propose an autoencoder (AE) for image-super resolution inspired again in the CV literature. This exploits the ability to understand the propagation environment as an image, augmenting the resolution of the radio maps from lower antennas to higher, or worse SNR conditions to better. Here, to leverage the sensing performance, we provide an extensive set of simulations by analyzing the impact of LIS parameters. We focus on analyzing aspects such as antenna spacing, aperture, number of antennas, as well as the generalization performance of the provided method.

Paper C is the first paper that proposes the MF-based radio map. In this work, we assume the LIS is deployed in the ceiling. Then, by taking advantage of arbitrary signals in the environment from active devices, we aim to detect and localize passive humans (i.e., they are not carrying active transmitters) by using the aforementioned maps that describe the environment. Here we make use of AI to perform the processing of the radio maps that will be used for human detection and localization. First, k-means is used to extract meaningful information from the map and separate the background of the reflections (scatters and humans). Next after some image processing procedures that will be discussed deeply in Paper C, we will end up adopting another CV technique called component labeling (CL) that will allow us to detect and position the passive humans in the scenario. In this paper, we leverage the effect of the number of active devices in the room in the sensing performance in terms of passive humans localization error and percentage of detection for 10 randomly located passive humans in the scenario, while we show as well where the system breaks: we analyze how much distance is necessary between humans to be correctly detected.

Paper D aims to demonstrate the versatile potential of LIS as a dual-function device, capable of both sensing and communication, as compared to a purely sensing device such as the millimeter-wave (mmWave) radar. This analysis is built upon the Matched Filter (MF)-based radio maps introduced in Paper C. The study presents a comparative performance evaluation between LIS and mmWave radar for the task of localizing a single passive human. Although the results reveal that mmWave radar yields superior performance, it is shown that the LIS system is capable of delivering competitive localization accuracy. This becomes particularly significant given the unique advantage of the LIS system's dual functionality. Unlike radars, which are limited to sensing operations, LIS can concurrently handle sensing and communication tasks, thereby highlighting its integrated operation within a singular device. This illustrates the promising potential and versatility of LIS in real-world applications, countering the limitations of devices like radar systems that are solely designed for sensing.

Paper E leverages the potential of an MF-based radio map with LIS for floor map reconstruction. In this paper, we intend to show another application beyond localization. With our method, we are able to use AI in the form of generative adversarial networks (GANs) to translate our radio maps into floor plans. The proposed method is leveraged against others in the literature such as U-Net and least squares linear reconstruction in terms of accuracy of floor map reconstruction.

Paper F aims to provide an improvement of the MF-based radio map method presented in paper C. We will provide online processing of the radio maps using AI with robust principal component analysis (RPCA) to process these maps. We then compare the detection and localization performance against our previous proposed method.

In summary, all the papers of this group aim to tackle the challenges of RQ1. We provide two radio maps method based on LIS to describe the environment and leverage its applicability in a range of examples. Besides, we also compare its accuracy against pure sensing systems such as mmWave radars. With this research, we aim to pave the way towards the usage of LIS not only for communication but sensing, enabling an integrated device that, exploited by AI, can be a key enabler in future 6G MIMO systems.

4 MMWave Wi-Fi sensing

This group focuses on RQ 2. A method based on neural ordinary differential equations (NODE) [84] is proposed to address the existing challenges. The main idea is to take advantage of object dynamics for mmWave localization and model the evolution of the hidden states in a continuous fashion. We begin this part with a short overview of the beam training localization problem and follow by explaining the NODE theory to further understand its advantages in this task. Next, we provide a summary of the papers of this group to close this section.

4.1 Beam training protocol for mmWave Wi-Fi

The IEEE 802.11ad/ay standard is an improvement of 802.11ad for WLANs and provides increased bandwidth and data rates by employing MIMO up to 60 GHz.

One of the most critical parts of such high frequencies is the beam training process. Due to the high path-loss proper beam training must be conveyed in order to take

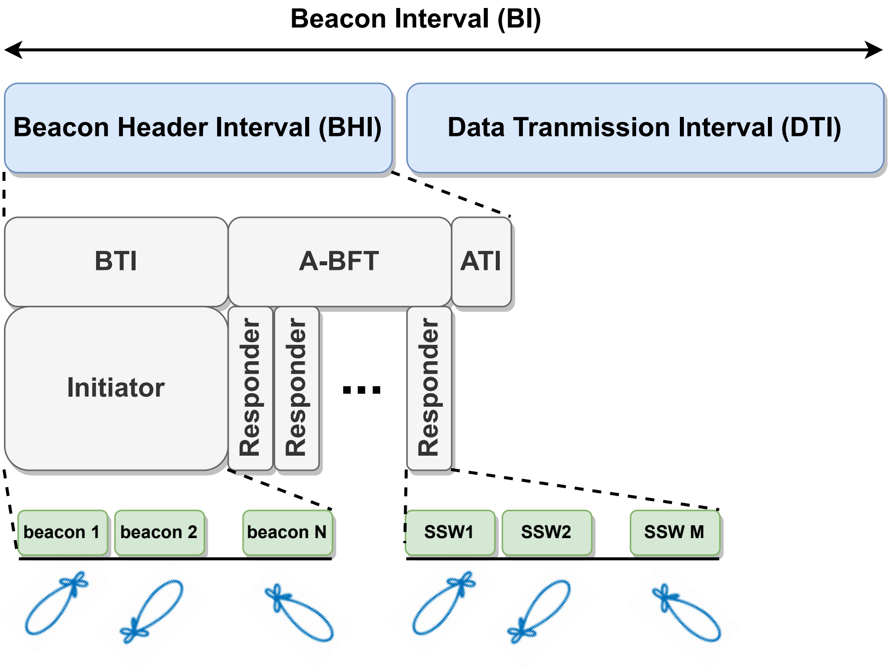


Fig. 3.4: Beam training protocol for IEEE 802.11ad/ay.

advantage of this technology. This is mandatory in the 60 GHz band, where signals do not reach far and are easily absorbed/blocked by obstacles.

The beam training procedure involves the AP and the receiver (client) working together to determine the best direction for the beam steerings of both devices. To do that, the exchange of specific training frames is performed, defined by the standard. Figure 3.4 shows an illustration of this procedure. In the 802.11ad/ay protocols, the time is divided into fixed repeated intervals known as beacon intervals (BI). Its length is defined by the network and can range from 100 ms to 800 ms. Within the BI, two subframes occur: the beacon header interval (BHI) and the data transmission interval (DTI). Besides during the BHI two tasks are also performed: beacon transmission interval (BTI) and association beamforming training (A-BFT).

During the BTI, the AP sends out beacon frames in all directions to announce its presence in the network and be detectable by the clients. These beacon frames are sent over a predefined set of directions using beamforming. Then, the clients that receive any of the beacon frames can use it to identify the presence of the AP. In a situation where multiple users are simultaneously calculating their own received beam SNRs corresponding to each of the transmitted beam patterns, a quasi-omnidirectional receiving beam pattern is employed. This process supposes a significant overhead. Consequently, it is desirable to limit the number of directional beam patterns within a beacon, leading to a condition of sparsely sampled beam measurement. We here spot one of the challenges of using these measurements for localization: *sparsity*.

After the BTI, the A-BFT takes place. At this moment, the clients that have identified the AP can request an association with it. They can start a beam training procedure by sending a short sequence of sector sweep (SSW) frames to the AP. Multiple responders randomly select a slot that could lead to contention over the wireless channel. As a result, this can lead to beam SNR measurements to be *irregular* in time, which is another of the challenges of using these features for localization.

Finally, the DTI is used for the exchange of data frames. These data frames are sent using the beamforming directions that were determined during the BHI. We will not use anything of these frames for localization.

4.2 Neural Ordinary Differential Equations

As mentioned throughout the thesis, using beam training measurements is a challenging task due to the intermittency of its beam training measurements. Here, we will provide some basis to further understand the framework proposed in papers G and H to tackle RQ2.

Neural ODEs basics

First of all, we try to give an explanation of the continuous nature of Neural ODEs and why it suits irregularly sampled data learning, as our target beam training measurements. The hidden state of a Neural ODE evolves according to the following ordinary differential equation:

$$\frac{d\mathbf{h}(t)}{dt} = f(\mathbf{h}(t), t; \theta), \tag{3.5}$$

where $\mathbf{h}(t)$ represents the hidden state at any time t, f is a function parameterized by θ (a neural network) which determines the rate of change of the hidden state, and t indicates any point in continuous time. This is a significant change w.r.t traditional deep learning models, which involve discrete transformations of hidden states corresponding to the number of layers.

To solve this differential equation, we usually employ numerical ODE solvers. The solver starts at the initial time t_0 with the initial state $\mathbf{h}(t_0) = \mathbf{h}_0$ (usually set to the input data), and it integrates the ODE to find the final state $\mathbf{h}(t_1)$ at some later time t_1, which is considered as the output of the model.

In this way, solving the ODE provides us with a continuous path of the hidden states $\mathbf{h}(t)$ for $t_0 \leq t \leq t_1$. We can query the state at any point in time within this interval, giving us the ability to construct models with continuous depth. This enables a big range of possibilities, as we are not restricted to the sampling of the data.

In the context of irregularly sampled data, the timestamps of the data points are spread unevenly over time. Standard models, such as RNNs, struggle with this kind of data because they are designed to handle sequences with fixed intervals. On the other hand, the flexibility provided by the continuous-time framework of Neural ODEs allows the model to handle data at any arbitrary time point, making it a natural fit for irregularly sampled data. We can use this principle to handle the measurements beam SNR data.

ODE-RNN framework

We now show the benefits of both RNNs and Neural ODEs in a single model, resulting in what is known as an ODE-RNN.

An ODE-RNN combines the strengths of RNNs and Neural ODEs. The model architecture includes a recurrent unit that updates the hidden state whenever a new data point arrives and an ODE solver that evolves this hidden state in between observations.

This hybrid architecture allows the model to deal with irregularly sampled data more effectively. The RNN component ensures that the model can handle sequences and can update the state with new information when it becomes available. Meanwhile, the Neural ODE component ensures that the model can accurately and efficiently interpolate the hidden state between observations, regardless of the gap between timestamps.

In mathematical terms, whenever a new data point $x(t_i)$ arrives at time t_i, the RNN updates the hidden state as

$$\mathbf{h}(t_i) = RNN(\mathbf{h}(t_{i-1}), x(t_i); \theta_{RNN}), \tag{3.6}$$

where θ_{RNN} are the parameters of the RNN, and t_{i-1} and t_i are two consecutive timestamps. Then, in between observations, the hidden state evolves according to the Neural ODE:

$$\frac{d\mathbf{h}(t)}{dt} = f(\mathbf{h}(t), t; \theta_{ODE}), \tag{3.7}$$

where θ_{ODE} are the parameters of the Neural ODE.

Latent Neural ODEs

To further understand our papers G and H, we here provide an overview of a latent neural ODE [85] framework. Inspired by latent-variable models based on Autoencoders [86], it can be proposed a latent-variable time series model, where the generative model is defined by an ODE. The latent state z_0 would then determine the entire trajectory:

$$z_0 \sim p(z_0) \tag{3.8}$$

$$z_0, z_1, ..., z_N = ODESolve(f(z_0, (t_0, t_1, ..., t_n); \theta_{ODE}) \tag{3.9}$$

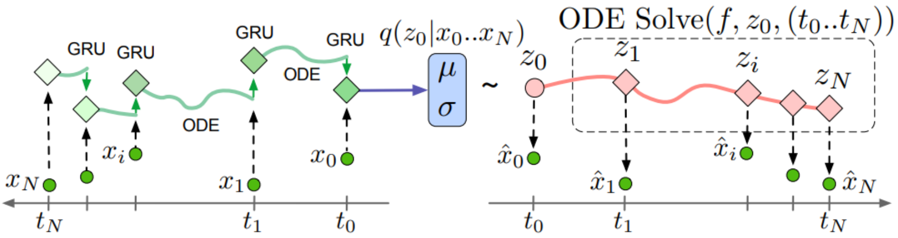


Fig. 3.5: The latent ODE model with an ODE-RNN encoder [85].

$$\text{each} \qquad x_i \sim p(x_i|z_i) \qquad i = 0, 1, ..., N, \tag{3.10}$$

where N represents the number of time steps in a sequence.

By following a variational autoencoder framework, it is required to estimate the approximate posterior $q(z_0|\{x_i, t_i\}_{i=0}^N)$. In this way, a variable length-sequence is encoded into a fixed dimensional embedding that is later on decoded into another variable length-sequence.

An ODE-RNN network can be used as the encoder for a latent ODE model, resulting then in a fully ODE-based sequence-to-sequence approach. Usually, the mean and the standard deviation of the approximate posterior $q(z_0|\{x_i, t_i\}_{i=0}^N)$ are a function of the final hidden state of an ODE-RNN encoder:

$$q(z_0|\{x_i, t_i\}_{i=0}^N) = \mathcal{N}(\mu_{z_0}, \sigma_{z_0}) \quad \text{where} \quad \mu_{z_0}, \sigma_{z_0} = g(ODE-RNN(\{x_i, t_i\}_{i=0}^N; \phi_{ODE}), \tag{3.11}$$

where g is a simple multilayer perceptron (MLP) translating the final hidden state of the ODE-RNN encoder into the mean and the variance of z_0. Then this model is jointly training the encoder and the decoder by maximizing the evidence lower bound (ELBO):

$$\mathcal{L}_{\text{ELBO}} = \mathbb{E}_{q(z_0|\{x_i,t_i\}_{i=0}^N)}[\log p(x_0, ..., x_N)] - \text{KL}[q(z_0|\{x_i, t_i\}_{i=0}^N)||p(z_0)], \tag{3.12}$$

where KL denotes the Kullback–Leibler divergence term. The first terms correspond to the reconstruction loss of the input sequence data and the second term acts as a regularizer of the latent space. We can see the idea of this model seems suitable to tackle irregularly-sampled data. More details of our proposed framework, which follows a latent neural ODE principle, will be provided in their respective sections related to the papers of these groups. Figure 3.5 illustrates the framework.

4.3 Paper summaries

There are 2 papers in this category:

Paper G: (*published*) Cristian J Vaca-Rubio, Pu Wang, Toshiaki Koike-Akino, Ye

Wang, Petros Boufounos, Petar Popovski, “mmWave Wi-Fi trajectory estimation with continuous-time neural dynamic learning,” *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 1-5, 2023.
Paper H: (*submitted*) Cristian J Vaca-Rubio, Pu Wang, Toshiaki Koike-Akino, Ye Wang, Petros Boufounos, Petar Popovski, “Object trajectory estimation with continuous-time neural dynamic learning of millimeter-wave Wi-Fi,” *IEEE Journal of Selected Topics in Signal Processing (JSTSP)*, 2023.

Paper G proposes our first framework entitled dual decoder neural dynamic learning (DDND). We aim to tackle the problem of intermittent beam training measurements by providing a framework that handles the evolution of the beam SNRs data in a continuous fashion for indoor localization over a trajectory. Besides, we want to exploit the object dynamics of the moving target and we aim to learn this behavior with a neural ODE in the latent space. In this paper, we provide the first steps to solve this problem comparing them to frame-based methods in the literature as well as sequence regression models based on RNN, which are not suitable to handle the irregular time samples. We show exceptional improvement in performance against all the baseline methods.

Paper H further extends the dual decoder neural dynamic framework. We derive instead a loss function from the original ELBO of VAE to perform the localization task. Besides, we provide again a comparison against a range of baseline methods outperforming them significantly in a trajectory estimation task. We also provide an extensive set of simulations to perform an ablation study of the proposed framework and show its immense flexibility in supervision during training that improves significantly the performance. Besides, we show its generalization capabilities testing in different days of data collection sessions.

In short, we provide a flexible and dynamic framework that can be easily integrated into the communication standards of mmWave MIMO Wi-Fi, achieving the integration of sensing and communication.

5 Dynamic sensing and communication

This group focuses on RQ3. We provide an overview of the 5G fixed slot vision for allocating resources for data transmission and beam sweeping/tracking (sensing) in cellular networks. We then provide a high-level idea description of our proposed method based on proximal policy optimization, an RL framework. To finish, we close this section with a summary of the paper within this group.

5.1 Beam tracking and communications periodicity in 5G

In 5G networks that operate in mmWave bands, the necessity of beam tracking (sensing) and beamforming are essential to ensure a reliable connection between the user equipment (UE) and the BS. Due to these high frequencies, these connections can be disrupted by physical objects or sudden changes in the environment.

Beam tracking is a mechanism to try to keep the quality of the beam to adapt the best possible connection according to the changes in the environment. This means, it entails continuously adjusting the direction of the formed used beam to follow the UE and compensate for these changes.

With this in mind, the periodicity of beam tracking is a critical aspect. There is a trade-off between ensuring a reliable connection (which requires high beam tracking frequency) and reducing unnecessary signaling to perform more data transmission (which requires less frequent beam tracking).

The periodicity is defined by the 3GPP standards. Typically, the standard defines some fixed periodicity according to different scenarios and use cases. Once a periodicity is set it is kept for the whole time. It might change if the network detects the situation changes, but this is not a dynamic solution but changing between a set of fixed periodicity cases.

Furthermore, we aim to provide a smart AI-based solution that does dynamic allocation of both: resources for communication and sensing (beam tracking).

5.2 Proximal policy optimization for dynamic resource allocation and sensing

As mentioned before and trying to answer RQ3, we aim to provide a dynamic solution for sensing and communications.

On the SoA we highlighted how RL solutions were useful to solve the problems of beam alignment/tracking and resource allocation when framing the requirements and tradeoff as an optimization problem. Following this line of thought, in this thesis, we propose a solution based on PPO. In paper I a more in-depth description of this framework will be provided, while here we will provide a high-level overview of the problem to solve.

We assume time-slotted resources and that we can serve one user at a time. Our PPO algorithm can just take one action: sensing (beam tracking for a user), scheduling downlink traffic, or scheduling uplink traffic. In this way, this can be modeled as a markov decision process (MDP) illustrated in Figure 3.6.

Furthermore, our simplistic representation of the scenario tries to serve multiple users in a completely dynamic fashion, i.e., no fixed periodicity for beam tracking operations but adaptive. In short, Figure 3.7 shows an exemplary representation of a simplified frame structure. The slots are allocated either for sensing, uplink or downlink for a specific user.

To give a better understanding of the high-level functioning of the PPO procedure, we can take a look at Figure 3.8. Our method is implemented as an Actor-Critic that interacts with the environment gathering a set of experience tuples that are stored in

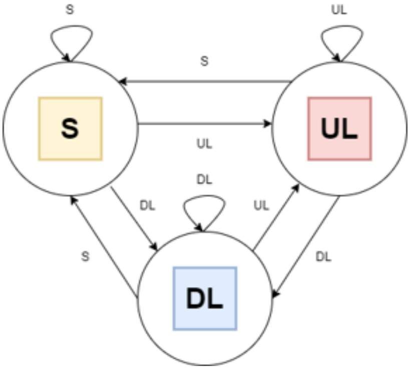


Fig. 3.6: MDP dynamic sensing and communications.

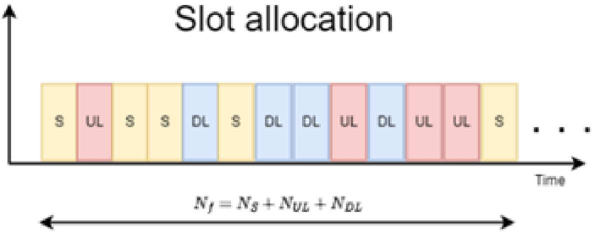


Fig. 3.7: MDP dynamic sensing and communications.

the memory. After that, both networks are trained to maximize the rewards. A more detailed explanation is given in paper I.

5.3 Paper summaries

This group has only one paper:
Paper I: (*submitted*) Cristian J Vaca-Rubio, Carles Navarro Manchón, Ramoni Adeogun, Petar Popovski, "Proximal policy optimization for integrated sensing and communication in mmWave systems," *IEEE Wireless Communication Letters (WCL)*, 2023.

Paper I proposes a method based on PPO to perform dynamic allocation slots for both sensing and communications with the goal of minimizing the total average packet error rate (PER). We frame the problem in a vehicular network context in which reliability and slot allocation efficiency are key aspects to ensure a safe operation. Our

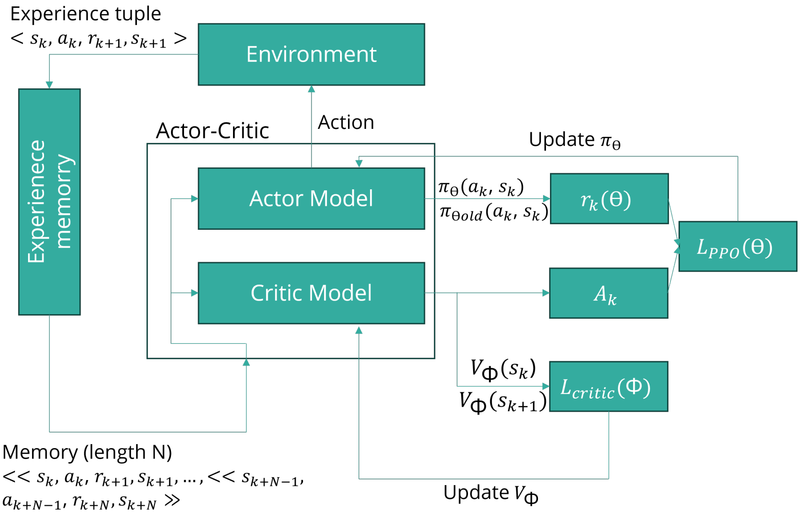


Fig. 3.8: MDP dynamic sensing and communications.

approach is compared against a range of baselines that mimics 5G fixed periodicity structure for different values. The results show a PER reduction with respect to the baselines as well as a significant beam tracking reduction and consequent slot allocation efficiency while exhibiting great generalization performance.

Chapter 4

Concluding remarks

This chapter is meant to wrap up the main contributions of the thesis by revising the key benefits of the methods and ideas employed and introduced throughout this work. Besides, a discussion about different future applications and research directions is provided.

1 Conclusions

In this section, we start with a general review of the findings of the thesis. Next, we revise the research questions asked at the beginning to show our studies answer and/or pave the initial ways towards solving them.

This thesis focuses on mainly providing AI-empowered methods for MIMO and beyond massive MIMO systems in a sensing context. It shows that in terms of LIS sensing, we pay the way and shed some light on the ability of beyond massive MIMO systems to perform sensing through radio map reconstruction. Besides, the thesis deviates from the idea that sequence-based solutions for mmWave indoor localization based on RNN are useful for new mmWave Wi-Fi standards integrated sensing and communications. Finally, the thesis also disagrees with the fixed periodicity of 5G beam tracking standards. In this way, for these two claims, this thesis studies and provides some methods to combat the challenges.

With the goal of answering RQ1, the proposed radio map methods using LIS use signal processing techniques to create radio maps of the environment as images. This thesis is the first in the literature to make the first steps into this approach and provides some insights about how to improve radio mapping, through extensive validation and simulation of LIS parameters and their impact on performance. It also makes use of CV and AI methods to take advantage of these radio images. This work also sheds some light on possible applications enabled by these methods and leverages its performance against pure sensing systems such as mmWave radars. In short, in response to RQ1, this thesis provides brand-new methods, analyzes parameters to understand its potential, leverages some applications, and compares against other sensing systems.

With respect to RQ2, our studies first analyze the current mmWave Wi-Fi standards and then make use of the already-established communication protocols in favor of integrating sensing into it seamlessly. Works in the literature were focused on tackling this indoor localization problem by developing either frame-based or sequenced-based RNN solutions to perform indoor localization based on beam training measurements. However, this thesis is looking into a more fundamental way of modeling machine learning solutions to address the issue of irregularity and sparsity measurements for Wi-Fi data. After analyzing all these aspects, the thesis aims to utilize this information to provide a continuous-time solution that overcomes such challenges. Besides, the results of this work, are experimentally evaluated in a real testbed, which guarantees its direct applicability to the real world. As a related note of these promising results, a patent was applied for the proposed method in this thesis.

In response to RQ3, our method can provide dynamic scheduling and communication and sensing to be more reliable, and efficient in mmWave MIMO systems. One of the main problems of fixed periodicity for beam tracking is its lack of immediate response to the changes in the network as well as excessive signaling. In this thesis, we provide a more adaptable and efficient solution.

Up to now, we have mentioned about all the advantages of the methods provided in this thesis. Now here we mention some disadvantages of these methods that can be seen as motivation to pursue further work.

One of the main problems of the LIS sensing methods is the fact that the arrays are huge in aperture, and the amount of antenna elements, and assuming one RF chain per antenna makes them difficult to manufacture right now. Besides, some physical phenomena degradation as mutual coupling should be taken into account. One way to compensate for these problems is composing the LIS by small sub-patches of arrays that are more feasible. Related to the RF chain and hybrid architecture could be also analyzed.

For the mmWave Wi-Fi localization framework provided the main disadvantage would be its heavy computational expense during training. However, this could be addressed by leveraging simpler architectures that still capture the dynamics of the motion in the latent space and provide good performance.

For the dynamic sensing and communication method scenarios with a higher number of users should be leveraged to assess its scalability performance. A way to tackle this drawback is to go into solutions that exploit multi-agent reinforcement learning.

2 Applications

In this section, we try to give some application examples of the methods developed throughout the thesis for current and future MIMO 6G systems. The first obvious application would be leveraging the joint communication and sensing capabilities of LIS by extracting information from radio environmental maps. Besides, 3D scene reconstruction could also be an interesting application for LIS-based systems. With regards to mmWave MIMO Wi-Fi, proper integration into the communication standard is the most direct application of the method, enabling new advantages that could be even used to enhance the beam training procedure assisted by localization prediction. Finally, for the dynamic sensing and communication a direct application would be in the 5G uses cases, trying to unify a method that consistently and smartly adapts to the network conditions.

3 Future work directions

There are many ways to extend the work done in the thesis:

With respect to LIS, advanced modeling and simulation can be developed. Including more physical phenomena like scattering, polarization and mutual coupling would offer a deeper understanding of LIS' behavior. Research into quantum materials as an alternative for excessive RF chains might also open new horizons that can be exploited to further develop radio map sensing techniques. Further research on interoperability and standardization might also be addressed in the future. Research on how LIS can be integrated into existing and emerging standards, ensuring compatibility accross different devices and technologies. Finally, providing and designing a protocol for integrated sensing and communication control message exchange is of major interest to fully exploit its integrated sensing and communication capabilities.

For mmWave Wi-Fi it can be extended to multimodal sensing applications. The applications can be extended beyond to human detection, tracking and environmental sensing. Besides it can be used as an integration with other sensing modalities. The collaborative use of mmWave with other sensors as optical, thermal or acoustic can lead to an enhance in reliability and accuracy. What is more the localization features explored in this thesis could be further expanded to impact on communication performance, leading to improve traffic routing, congestion management and so on. Another interesting direction could be exploiting the framework to leverage its performance over different rooms and essentially new scenarios not seen during training.

For dynamic sensing and communication, a possible and interesting research direction is the development of a learned protocol design. Developing an AI solution that can learn the interaction between BSs and UEs to coordinate all these scheduling procedures and associated signaling could be a really interesting topic that could go beyond current by-hand-developed protocols. Also, cross-layer design can be studied across multiple layers of the networking stack.

References

[1] M. Shafi, A. F. Molisch, P. J. Smith, T. Haustein, P. Zhu, P. De Silva, F. Tufvesson, A. Benjebbour, and G. Wunder, "5G: A tutorial overview of standards, trials, challenges, deployment, and practice," *IEEE Journal on Selected Areas in Communications*, vol. 35, no. 6, pp. 1201-1221, 2017.

[2] S. Dang, O. Amin, B. Shihada, and M.-S. Alouini, "What should 6G be?," *Nature Electronics*, vol. 3, no. 1, pp. 20-29, 2020.

[3] “Number of cities in which 5G is available 2023 by country.” `https://www.statista.com/statistics/1215456/5g-cities-by-country/`, accessed: 2023-07-05.

[4] H. Tullberg, P. Popovski, Z. Li, M. A. Uusitalo, A. Hoglund, O. Bulakci, M. Fallgren, and J. F. Monserrat, "The METIS 5G system concept: Meeting the 5G requirements," *IEEE Communications Magazine*, vol. 54, no. 12, pp. 132-139, 2016.

[5] O. O. Erunkulu, A. M. Zungeru, C. K. Lebekwe, M. Mosalaosi, and J. M. Chuma, "5G mobile communication applications: A survey and comparison of use cases," *IEEE Access*, vol. 9, pp. 97251-97295, 2021.

[6] H. Ullah, N. G. Nair, A. Moore, C. Nugent, P. Muschamp, and M. Cuevas, "5G communication: An overview of vehicle-to-everything, drones, and healthcare use-cases," *IEEE Access*, vol. 7, pp. 37251-37268, 2019.

[7] J. G. Andrews, S. Buzzi, W. Choi, S. V. Hanly, A. Lozano, A. C. K. Soong, and J. C. Zhang, "What will 5G be?," *IEEE Journal on Selected Areas in Communications*, vol. 32, no. 6, pp. 1065-1082, 2014.

[8] S. Li, L. D. Xu, and S. Zhao, "5G Internet of Things: A survey," *Journal of Industrial Information Integration*, vol. 10, pp. 1-9, 2018.

[9] E. G. Larsson, O. Edfors, F. Tufvesson, and T. L. Marzetta, "Massive MIMO for next generation wireless systems," *IEEE Communications Magazine*, vol. 52, no. 2, pp. 186-195, 2014.

[10] A.-S. Bana, G. Xu, E. De Carvalho, and P. Popovski, “Ultra reliable low latency communications in massive multi-antenna systems,” in *2018 52nd Asilomar Conference on Signals, Systems, and Computers.* IEEE, 2018, pp. 188–192.

[11] A.-S. Bana, E. De Carvalho, B. Soret, T. Abrao, J. C. Marinello, E. G. Larsson, and P. Popovski, “Massive mimo for internet of things (iot) connectivity,” *Physical Communication*, vol. 37, p. 100859, 2019.

[12] D. López-Pérez, A. De Domenico, N. Piovesan, G. Xinli, H. Bao, S. Qitao, and M. Debbah, "A survey on 5G radio access network energy efficiency: Massive MIMO, lean carrier design, sleep modes, and machine learning," *IEEE Communications Surveys & Tutorials*, vol. 24, no. 1, pp. 653-697, 2022.

[13] C.-X. Wang, X. You, X. Gao, X. Zhu, Z. Li, C. Zhang, H. Wang, Y. Huang, Y. Chen, H. Haas, and others, "On the road to 6G: Visions, requirements, key technologies and testbeds," *IEEE Communications Surveys & Tutorials*, 2023.

[14] F. Liu, Y. Cui, C. Masouros, J. Xu, T. X. Han, Y. C. Eldar, and S. Buzzi, "Integrated sensing and communications: Toward dual-functional wireless networks for 6G and beyond," *IEEE Journal on Selected Areas in Communications*, vol. 40, no. 6, pp. 1728-1767, 2022.

[15] A. Bourdoux, A. N. Barreto, B. van Liempd, C. de Lima, D. Dardari, D. Belot, E.-S. Lohan, G. Seco-Granados, H. Sarieddeen, H. Wymeersch, et al., "6G white paper on localization and sensing," *arXiv preprint arXiv:2006.01779*, 2020.

[16] C. Li, Z. Cao, and Y. Liu, "Deep AI enabled ubiquitous wireless sensing: A survey," *ACM Computing Surveys (CSUR)*, vol. 54, no. 2, pp. 1-35, 2021.

[17] C. Huang, S. Hu, G. C. Alexandropoulos, A. Zappone, C. Yuen, R. Zhang, M. Di Renzo, and M. Debbah, "Holographic MIMO surfaces for 6G wireless networks: Opportunities, challenges, and trends," *IEEE Wireless Communications*, vol. 27, no. 5, pp. 118-125, 2020.

[18] S. Hu, F. Rusek, and O. Edfors, "Beyond Massive MIMO: The Potential of Data Transmission With Large Intelligent Surfaces," *IEEE Transactions on Signal Processing*, vol. 66, no. 10, pp. 2746-2758, 2018.

[19] D. Dardari, "Communicating With Large Intelligent Surfaces: Fundamental Limits and Models," *IEEE Journal on Selected Areas in Communications*, vol. 38, no. 11, pp. 2526-2537, 2020. doi: 10.1109/JSAC.2020.3007036

[20] R. J. Williams, P. Ramírez-Espinosa, E. De Carvalho, and T. L. Marzetta, "Multiuser MIMO with large intelligent surfaces: Communication model and transmit design," in *ICC 2021-IEEE International Conference on Communications*, pp. 1-6, 2021.

[21] S. Hu, F. Rusek, and O. Edfors, "Beyond Massive MIMO: The Potential of Positioning With Large Intelligent Surfaces," *IEEE Transactions on Signal Processing*, vol. 66, no. 7, pp. 1761-1774, 2018.

[22] S. Hu, F. Rusek, and O. Edfors, "Cramér-Rao lower bounds for positioning with large intelligent surfaces," in *2017 IEEE 86th Vehicular Technology Conference (VTC-Fall)*, pp. 1-6, 2017.

[23] J. Yu, P. Wang, T. Koike-Akino, Y. Wang, P. V. Orlik, and H. Sun, "Human pose and seat occupancy classification with commercial MMWave WiFi," in *2020 IEEE Globecom Workshops (GC Wkshps)*, pp. 1-6, 2020, IEEE.

[24] T. Koike-Akino, P. Wang, M. Pajovic, H. Sun, and P. V. Orlik, "Fingerprinting-based indoor localization with commercial MMWave WiFi: A deep learning approach," *IEEE Access*, vol. 8, pp. 84879-84892, 2020.

[25] Y. Siriwardhana, P. Porambage, M. Liyanage, and M. Ylianttila, "AI and 6G security: Opportunities and challenges," in *2021 Joint European Conference on Networks and Communications & 6G Summit (EuCNC/6G Summit)*, pp. 616-621, 2021, IEEE.

[26] “6G: The Next Horizon.” `https://www.huawei.com/en/huaweitech/future-technologies/6g-the-next-horizon`, accessed: 2023-07-05.

[27] C. R. Anderson, R. K. Martin, T. O. Walker, and R. W. Thomas, "Radio tomography for roadside surveillance," *IEEE Journal of Selected Topics in Signal Processing*, vol. 8, no. 1, pp. 66-79, 2013.

[28] T. Matsuda, Y. Nishikawa, E. Takahashi, T. Onishi, and T. Takeuchi, "Binary Radio Tomographic Imaging in Factory Environments Based on LOS/NLOS Identification," *IEEE Access*, vol. 11, pp. 22418-22429, 2023.

[29] C. Vergara, R. K. Martin, P. J. Collins, and J. R. Lievsay, "Multi-sensor data fusion between radio tomographic imaging and noise radar," *IET Radar, Sonar & Navigation*, vol. 14, no. 2, pp. 187-193, 2020.

[30] D. Lee, D. Berberidis, and G. B. Giannakis, "Adaptive Bayesian radio tomography," *IEEE Transactions on Signal Processing*, vol. 67, no. 8, pp. 1964-1977, 2019.

[31] Q. Guo, Y. Li, X. Liang, J. Dong, and R. Cheng, "Through-the-wall image reconstruction via reweighted total variation and prior information in radio tomographic imaging," *IEEE Access*, vol. 8, pp. 40057-40066, 2020.

[32] Z. Wang, X. Guo, and G. Wang, "Exploring the Laplace prior in radio tomographic imaging with sparse Bayesian learning towards the robustness to multipath fading," *Sensors*, vol. 19, no. 23, p. 5126, 2019.

[33] S. Denis, R. Berkvens, and M. Weyn, "A survey on detection, tracking and identification in radio frequency-based device-free localization," *Sensors*, vol. 19

[34] F. Khan, S. K. Leem, and S. H. Cho, "In-air continuous writing using UWB impulse radar sensors," *IEEE Access*, vol. 8, pp. 99302-99311, 2020.

[35] M. Tatarchenko and K. Rambach, "Histogram-based Deep Learning for Automotive Radar," in *2023 IEEE Radar Conference (RadarConf23)*, pp. 1-6, 2023, IEEE.

[36] Z. Hong, Y. Petillot, A. Wallace, and S. Wang, "Radar SLAM: A robust SLAM system for all weather conditions," *arXiv preprint arXiv:2104.05347*, 2021.

[37] T. Kato, H. Yamada, and H. Mori, "Experimental study on 3D imaging using millimeter-wave non-uniform 2D-MIMO radar," *IEICE Communications Express*, vol. 11, no. 12, pp. 760-765, 2022.

[38] S. Bi, J. Lyu, Z. Ding, and R. Zhang, "Engineering radio maps for wireless resource management," *IEEE Wireless Communications*, vol. 26, no. 2, pp. 133-141, 2019.

[39] Q. Zhang and W. Saad, "Semi-supervised learning for channel charting-aided IoT localization in millimeter wave networks," in *2021 IEEE Global Communications Conference (GLOBECOM)*, pp. 1-6, 2021, IEEE.

[40] T. Ponnada, P. Kazemi, H. Al-Tous, Y.-C. Liang, and O. Tirkkonen, "Best beam prediction in non-standalone mm wave systems," in *2021 Joint European Conference on Networks and Communications & 6G Summit (EuCNC/6G Summit)*, pp. 532-537, 2021, IEEE.

[41] Z. Tian, Y. Li, M. Zhou, and Z. Li, "WiFi-based adaptive indoor passive intrusion detection," in *2018 IEEE 23rd International Conference on Digital Signal Processing (DSP)*, pp. 1-5, 2018, IEEE.

[42] W. Wang, A. X. Liu, M. Shahzad, K. Ling, and S. Lu, "Understanding and modeling of WiFi signal based human activity recognition," in *Proceedings of the 21st Annual International Conference on Mobile Computing and Networking*, pp. 65-76, 2015.

[43] W. Wang, A. X. Liu, M. Shahzad, K. Ling, and S. Lu, "Device-free human activity recognition using commercial WiFi devices," *IEEE Journal on Selected Areas in Communications*, vol. 35, no. 5, pp. 1118-1131, 2017.

[44] J. Liu, Y. Wang, Y. Chen, J. Yang, X. Chen, and J. Cheng, "Tracking vital signs during sleep leveraging off-the-shelf WiFi," in *Proceedings of the 16th ACM International Symposium on Mobile Ad Hoc Networking and Computing*, pp. 267-276, 2015.

[45] H. Abdelnasser, M. Youssef, and K. A. Harras, "Wigest: A ubiquitous WiFi-based gesture recognition system," in *2015 IEEE Conference on Computer Communications (INFOCOM)*, pp. 1472-1480, 2015, IEEE.

[46] X. Li, "RSS-based location estimation with unknown pathloss model," *IEEE Transactions on Wireless Communications*, vol. 5, no. 12, pp. 3626-3633, 2006.

[47] S. Mazuelas, A. Bahillo, R. M. Lorenzo, P. Fernandez, F. A. Lago, E. Garcia, J. Blas, and E. J. Abril, "Robust indoor positioning provided by real-time RSSI values in unmodified WLAN networks," *IEEE Journal of Selected Topics in Signal Processing*, vol. 3, no. 5, pp. 821-831, 2009.

[48] C. Liu, D. Fang, Z. Yang, H. Jiang, X. Chen, W. Wang, T. Xing, and L. Cai, "RSS distribution-based passive localization and its application in sensor networks," *IEEE Transactions on Wireless Communications*, vol. 15, no. 4, pp. 2883-2895, 2015.

[49] A. Chriki, H. Touati, and H. Snoussi, "SVM-based indoor localization in wireless sensor networks," in *2017 13th International Wireless Communications and Mobile Computing Conference (IWCMC)*, pp. 1144-1149, 2017, IEEE.

[50] N. Singh, S. Choe, and R. Punmiya, "Machine learning based indoor localization using Wi-Fi RSSI fingerprints: An overview," *IEEE Access*, vol. 9, pp. 127150-127174, 2021.

[51] D. Li, B. Zhang, and C. Li, "A feature-scaling-based k-nearest neighbor algorithm for indoor positioning systems," *IEEE Internet of Things Journal*, vol. 3, no. 4, pp. 590-597, 2015.

[52] S. He and S.-H. G. Chan, "Wi-Fi fingerprint-based indoor positioning: Recent advances and comparisons," *IEEE Communications Surveys & Tutorials*, vol. 18, no. 1, pp. 466-490, 2015.

[53] Z. Turgut, S. Üstebay, G. Zeynep Gürkaş Aydın, and A. Sertbaş, "Deep learning in indoor localization using WiFi," in *International Telecommunications Conference: Proceedings of the ITelCon 2017, Istanbul*, pp. 101-110, 2019, Springer.

[54] F. Mahdavi, H. Zayyani, and R. Rajabi, "RSS localization using an optimized fusion of two deep neural networks," *IEEE Sensors Letters*, vol. 5, no. 12, pp. 1-4, 2021.

[55] M. T. Hoang, B. Yuen, X. Dong, T. Lu, R. Westendorp, and K. Reddy, "Recurrent neural networks for accurate RSSI indoor localization," *IEEE Internet of Things Journal*, vol. 6, no. 6, pp. 10639-10651, 2019.

[56] H.-Y. Hsieh, S. W. Prakosa, and J.-S. Leu, "Towards the implementation of recurrent neural network schemes for WiFi fingerprint-based indoor positioning," in *2018 IEEE 88th Vehicular Technology Conference (VTC-Fall)*, pp. 1-5, 2018, IEEE.

[57] H. Chen, Y. Zhang, W. Li, X. Tao, and P. Zhang, "ConFi: Convolutional neural networks based indoor Wi-Fi localization using channel state information," *IEEE Access*, vol. 5, pp. 18066-18074, 2017.

[58] R. Zhou, M. Hao, X. Lu, M. Tang, and Y. Fu, "Device-free localization based on CSI fingerprints and deep neural networks," in *2018 15th Annual IEEE International Conference on Sensing, Communication, and Networking (SECON)*, pp. 1-9, 2018, IEEE.

[59] C. Xiang, Z. Zhang, S. Zhang, S. Xu, S. Cao, and V. Lau, "Robust sub-meter level indoor localization - a logistic regression approach," in *ICC 2019-2019 IEEE International Conference on Communications (ICC)*, pp. 1-6, 2019, IEEE.

[60] C. Xiang, S. Zhang, S. Xu, X. Chen, S. Cao, G. C. Alexandropoulos, and V. K. N. Lau, "Robust sub-meter level indoor localization with a single WiFi access point—Regression versus classification," *IEEE Access*, vol. 7, pp. 146309-146321, 2019.

[61] IEEE, "IEEE Draft Standard for Information Technology - Telecommunications and Information Exchange Between Systems Local and Metropolitan Area Networks - Specific Requirements Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications Amendment 2: Enhancements for wireless LAN sensing, IEEE p802.11bf/d0.5, 2022."

[62] G. Bielsa, J. Palacios, A. Loch, D. Steinmetzer, P. Casari, and J. Widmer, "Indoor localization using commercial off-the-shelf 60 GHz access points," in *IEEE INFOCOM*, 2018, pp. 2384-2392.

[63] M. Pajovic, P. Wang, T. Koike-Akino, H. Sun, and P. V. Orlik, "Fingerprinting-based indoor localization with commercial MMWave WiFi–Part I: RSS and Beam Indices," in *GLOBECOM*, Dec. 2019.

[64] P. Wang, M. Pajovic, T. Koike-Akino, H. Sun, and P. Orlik, "Fingerprinting-based indoor localization with commercial mmwave WiFi–Part II: Spatial beam SNRs," in *GLOBECOM*, Dec 2019.

[65] P. Wang, T. Koike-Akino, and P. V. Orlik, "Fingerprinting-based indoor localization with commercial mmwave WiFi: A deep learning approach," *IEEE Access*, vol. 8, pp. 84879-84892, 2020.

[66] T. Koike-Akino, P. Wang, and P. V. Orlik, "Fingerprinting-based indoor localization with commercial mmwave WiFi: NLOS propagation," in *GLOBECOM*, December 2020.

[67] A. Blanco, P. J. Mateo, F. Gringoli, and J. Widmer, "Augmenting mmwave localization accuracy through sub-6 GHz on off-the-shelf devices," in *Proceedings of the 20th Annual International Conference on Mobile Systems, Applications and Services*, 2022, pp. 477-490.

[68] Va, Vutha, Choi, Junil, Shimizu, Takayuki, Bansal, Gaurav, and Heath, Robert W. "Inverse multipath fingerprinting for millimeter wave V2I beam alignment." IEEE Transactions on Vehicular Technology, vol. 67, no. 5, pp. 4042-4058, 2017.

[69] Ali, Anum, González-Prelcic, Nuria, and Heath, Robert W. "Millimeter wave beam-selection using out-of-band spatial information." IEEE Transactions on Wireless Communications, vol. 17, no. 2, pp. 1038-1052, 2017.

[70] Sutton, R. S., and Barto, A. G. (2018). Reinforcement learning: An introduction. MIT Press.

[71] Susarla, Praneeth, Deng, Yansha, Juntti, Markku, and Sílven, Olli. "Hierarchial-DQN Position-Aided Beamforming for Uplink mmWave Cellular-Connected UAVs." In GLOBECOM 2022-2022 IEEE Global Communications Conference, pp. 1308-1313, 2022.

[72] Raj, Vishnu, Nayak, Nancy, and Kalyani, Sheetal. "Deep reinforcement learning based blind mmwave MIMO beam alignment." IEEE Transactions on Wireless Communications, vol. 21, no. 10, pp. 8772-8785, 2022.

[73] Tse, D., and Viswanath, P. (2005). Fundamentals of wireless communication. Cambridge University Press.

[74] Boyd, S., and Vandenberghe, L. (2004). Convex optimization. Cambridge University Press.

[75] Saad, Walid, Han, Zhu, Debbah, Mérouane, Hjorungnes, Are, and Basar, Tamer. "Coalitional game theory for communication networks." IEEE Signal Processing Magazine, vol. 26, no. 5, pp. 77-97, 2009.

[76] Sun, Haoran, Chen, Xiangyi, Shi, Qingjiang, Hong, Mingyi, Fu, Xiao, and Sidiropoulos, Nicholas D. "Learning to optimize: Training deep neural networks for interference management." IEEE Transactions on Signal Processing, vol. 66, no. 20, pp. 5438-5453, 2018.

[77] Andrews, Jeffrey G., Bai, Tianyang, Kulkarni, Mandar N., Alkhateeb, Ahmed, Gupta, Abhishek K., and Heath, Robert W. "Modeling and analyzing millimeter wave cellular systems." IEEE Transactions on Communications, vol. 65, no. 1, pp. 403-430, 2016.

[78] Tefera, Mulugeta Kassaw, Zhang, Shengbing, and Jin, Zengwang. "Deep Reinforcement Learning-Assisted Optimization for Resource Allocation in Downlink OFDMA Cooperative Systems." Entropy, vol. 25, no. 3, p. 413, 2023, MDPI.

[79] Alam, Saniul, Islam, Sadia, Khandaker, Muhammad RA, Khan, Risala T, Tariq, Faisal, and Toding, Apriana. "Deep Q-Learning Based Resource Allocation in Interference Systems with Outage Constraint." arXiv preprint arXiv:2203.02791, 2022.

[80] Zhang, Haijun, Huang, Site, Jiang, Chunxiao, Long, Keping, Leung, Victor CM, and Poor, H Vincent. "Energy efficient user association and power allocation in millimeter-wave-based ultra dense networks with energy harvesting base stations." IEEE Journal on Selected Areas in Communications, vol. 35, no. 9, pp. 1936-1947, 2017.

[81] Zappone, Alessio, Di Renzo, Marco, Debbah, Merouane, Lam, Thanh Tu, and Qian, Xuewen. "Model-aided wireless artificial intelligence: Embedding expert knowledge in deep neural networks for wireless system optimization." IEEE Vehicular Technology Magazine, vol. 14, no. 3, pp. 60-69, 2019.

[82] Alkhateeb, Ahmed, Alex, Sam, Varkey, Paul, Li, Ying, Qu, Qi, and Tujkovic, Djordje. "Deep learning coordinated beamforming for highly-mobile millimeter wave systems." IEEE Access, vol. 6, pp. 37328-37348, 2018.

[83] Simonyan, Karen and Zisserman, Andrew. "Very deep convolutional networks for large-scale image recognition." arXiv preprint arXiv:1409.1556, 2014.

[84] Chen, Ricky TQ, Rubanova, Yulia, Bettencourt, Jesse, and Duvenaud, David K. "Neural ordinary differential equations." Advances in neural information processing systems, vol. 31, 2018.

[85] Rubanova, Yulia and Chen, Ricky TQ and Duvenaud, David K. "Latent ordinary differential equations for irregularly-sampled time series." Advances in neural information processing systems, vol. 32, 2019.

[86] Kingma, Diederik P and Welling, Max. "An introduction to variational autoencoders." Foundations and Trends® in Machine Learning, vol. 12, no. 4, pp. 307-392, 2019.

Part II

Papers

Paper A

A primer on large intelligent surface (LIS) for wireless sensing in an industrial setting

Cristian J Vaca-Rubio, Pablo Ramirez-Espinosa, Robin Jess Williams, Kimmo Kansanen, Zheng-Hua Tan, Elisabeth de Carvalho, Petar Popovski

The paper has been published in the
International Conference on Cognitive Radio Oriented Wireless Networks, pp.126-138, 2020.

© 2020 Springer
The layout has been revised.

Abstract

One of the beyond-5G developments that is often highlighted is the integration of wireless communication and radio sensing. This paper addresses the potential of communication-sensing integration of Large Intelligent Surfaces (LIS) in an exemplary Industry 4.0 scenario. Besides the potential for high throughput and efficient multiplexing of wireless links, an LIS can offer a high-resolution rendering of the propagation environment. This is because, in an indoor setting, it can be placed in proximity to the sensed phenomena, while the high resolution is offered by densely spaced tiny antennas deployed over a large area. By treating an LIS as a radio image of the environment, we develop sensing techniques that leverage the usage of computer vision combined with machine learning. We test these methods for a scenario where we need to detect whether an industrial robot deviates from a predefined route. The results show that the LIS-based sensing offers high precision and has a high application potential in indoor industrial environments.

1 Introduction

Massive Multiple-input Multiple-output (MIMO) is a fundamental technology in the 5th generation of wireless networks (5G), with the addition of a large number of antennas per base station as its key feature [1]. Looking towards post-5G, researchers are defining a new generation of base stations that are equipped with an even larger number of antennas, giving raise to the concept of LIS. Formally, an LIS designates a large continuous electromagnetic surface able to transmit and receive radio waves [2], which can be easily integrated into the propagation environment, e.g., placed on walls. In practice, an LIS is composed of a collection of closely spaced tiny antenna elements. Whilst the performance of LIS in communications has received considerably attention recently [2–5], the potential of these devices could go beyond communications applications, e.g., environment sensing. Indeed, such large surfaces contain many antennas that can be used as sensors of the environment based on the Channel State Information (CSI).

Sensing strategies based on electromagnetic signals have been thoroughly addressed in the literature in different ways, and applied to a wide range of applications. For instance, in [6], a real-time fall detection system is proposed through the analysis of the communication signals produced by active users, whilst the authors in [7] use Doppler shifts for gesture recognition. Radar-like sensing solutions are also available for user tracking [8] and real-time breath monitoring [9], as well as sensing methods based on radio tomographic images [10, 11]. Interestingly, whilst some of these techniques resort solely on the amplitude (equivalently, power) of the receive signals [8, 11], in those cases where sensing small scale variations is needed, the full CSI (i.e., amplitude and phase of the impinging signals) is required [9, 10].

On a related note, Machine Learning (ML) based approaches are gaining popularity in the context of massive MIMO systems, providing suitable solutions to optimization problems [12–15]. Due to the even larger dimensions of the system in extra-large arrays, deep learning may play a key role in exploiting complex patterns of information dependency between the transmitted signals.

The popularization of LIS as a natural next step from massive MIMO gives rise to larger arrays and more degrees of freedom, providing huge amounts of data which can feed ML algorithms. Hence, deep learning arises as a potential solution to exploit the performance of LIS.

In this work, we aim to pave the way to the combined use of both deep learning algorithms and the aforementioned large surfaces, exploring, for first time in the literature, the potential of such a joint solution to sense the propagation environment. Specifically, the contribution of this work is twofold:

- We propose an image-based sensing technique based on the received signal power at each antenna element of an LIS. These power samples are processed to generate a high resolution image of the propagation environment that can be used to feed computer vision algorithms to sense large-scale events.
- A computer vision algorithm, based on transfer learning and support vector machine (SVM), is defined to process the radio images generated by the LIS in order to detect anomalies over a predefined robot route.

The performance of the proposed solution is tested in an indoor industrial scenario, where the impact of the array aperture, sampling period and the inter-antenna distance is thoroughly evaluated. We show that both larger apertures and smaller separations between the LIS elements render higher resolution images, improving the performance of the system.

2 Problem formulation

We consider an industrial scenario where a robot is following a fixed route, and assume that, due to arbitrary reasons, it might deviate from the predefined route and follow an alternative (undesired) trajectory. Hence, our goal is, based on the sensing signal transmitted by the target device, being able to detect whether the robot is following the correct route or not.

In order to perform the anomalous route detection, we assume that an LIS (i.e., a large array of M closely spaced antennas), is placed in the scenario. Therefore, the sensing problem reduces to determine, from the received signal at each of the LIS elements, if the transmission has been made from a point at the desired route, denoted by $\mathbf{p}_c \in \mathbb{R}^3$, or from an anomalous one, denoted by $\mathbf{p}_a \in \mathbb{R}^3$. For the sake of simplicity in a real system implementation, and because we are interested in sensing large scale variations, we resort to the received signal amplitude (equivalently, power). This assumption may lead to simpler system implementations, avoiding the necessity of performing coherent detection.

A classical approach for the aforementioned problem would be performing a hypothesis test based on the received power signal vector. To that end, consider the received complex signal from either $\mathbf{p}_c$ or $\mathbf{p}_a$ to be

$$\mathbf{y}_k = \mathbf{h}_k x + \mathbf{n}_k, \quad k = \{c, a\}, \tag{A.1}$$

with x the transmitted (sensing) symbol, $\mathbf{h}_k \in \mathbb{C}^{M\times 1}$ the channel vector from each point and $\mathbf{n}_k \sim \mathcal{CN}_M(\mathbf{0}, \sigma^2\mathbf{I})$ the noise vector. Assume, without loss of generality, that $x = 1$. Hence, the received power vector is given by

$$\mathbf{w}_k = \left(\|y_{1,k}\|^2, \ldots, \|y_{M,k}\|^2\right)^T, \tag{A.2}$$

where $y_{i,k}$ for $i = 1, \ldots, M$ are the elements of $\mathbf{y}_k$. The hypothesis test is therefore formulated as

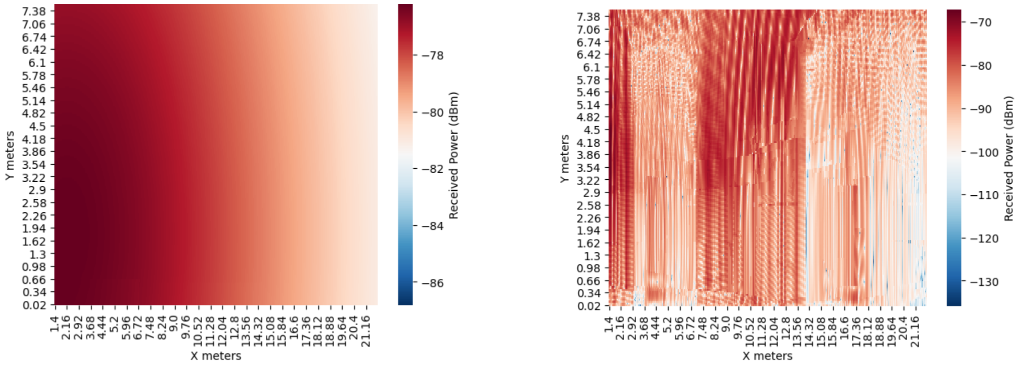
$$\frac{f_{\mathbf{w}_c}(\mathbf{w}|\mathbf{p}_c)}{f_{\mathbf{w}_a}(\mathbf{w}|\mathbf{p}_a)} \underset{\mathbf{p}_a}{\overset{\mathbf{p}_c}{\gtrless}} \frac{P_a}{P_c}, \tag{A.3}$$

where $f_{\mathbf{w}_k}(\cdot)$ for $k = \{c, a\}$ is the joint probability function of the received signal from each point, $\mathbf{w}$ is the observation vector, and P_a and P_c denote the probability of receiving a signal from $\mathbf{p}_a$ and $\mathbf{p}_c$, respectively. To obtain an optimal estimator, we would need to characterize the joint distribution of the received vector over all the possible anomolaous points, which implies knowing all the possible states of the channels for each path. Also, even in the most simple case, i.e., assuming a pure Line of Sight (LoS) propagation, we would still be unable to distinguish if the two points are in different trajectories or at distinct positions of the same route. Moreover, the a priori probabilities P_a and P_c are needed, which is a non-trivial task.

In a realistic environment, the complexity of the propagation paths is considerable, and the theoretical analysis becomes cumbersome and site-dependent. Hence, in order to gain insight into how the propagation paths between different positions translate into differences in the received signals, we have to resort on machine learning algorithms. This, together with the use of LIS, can provide the necessary information about the propagation environment in order to perform the anomalous route detection.

3 Holographic sensing

A hologram is a recorded interference pattern as a result of constructive and destructive combinations of the superimposed light-wavefronts, i.e., a photographic recording of a light field [16]. In a wireless context, an LIS could be described as a structure which uses electromagnetic signals impinging in a determined scatterer in order to obtain a profile of the environment. That is, we can use the signal power received at each of the multiple



(a) LoS, noiseless. **(b)** Real scenario, noiseless.

Fig. A.1: Holographic images for LOS and Industry scenarios.

elements of the LIS to obtain a high resolution image of the propagation environment. Using this approach, the complexity of the multipath propagation is reduced to using information represented as an image. This provides a twofold benefit: *i)* the massive number of elements that composed the LIS leads to an accurate environment sensing (i.e. high resolution image), and *ii)* it allows the use of computer vision algorithms and image processing techniques to deal with the resulting images.

As an illustrative example, Fig. A.1 shows the holographic images obtained from different propagation environments (x and y correspond to the physical dimension of the LIS). Specifically, Figs. A.1a correspond to a LoS propagation (no scatterers), whilst Fig. A.1b is obtained from an industrial scenario with a rich scattering. Note that, in the case in which different scatterers are placed, their position and shapes are captured by the LIS and represented in the image. To the best of the authors' knowledge, this is the first time that imaged-based sensing is proposed in the literature.

4 Machine learning for holographic sensing

4.1 Model description

We here propose the use of a machine learning model to perform the anomalous route classification task, based on the holographic images obtained at the LIS. In our considered problem, the training data is obtained by sampling the received power at certain temporal instants while the target device is moving along the route. In order to reduce both training time and scanning periods, we resort on transfer learning [17]. Thus, a small dataset can be used, improving the flexibility of the system in real deployments. Among the available strategies for this matter, we will use feature representation.

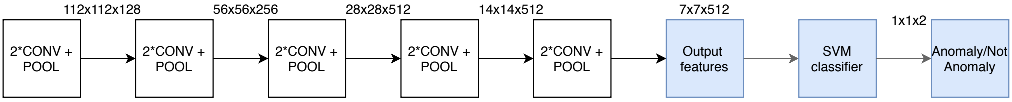


Fig. A.2: Proposed model. White and blue blocks refer to VGG19 re-used original architecture and to the additional blocks for our task, respectively.

One of the main requirements for transfer learning is the presence of models that perform well on already defined tasks. These models are usually shared in the form of a large number of parameters/weights the model achieved while being trained to a stable state [18]. The famous deep learning Python library, Keras [19], provides an easy way to reuse some of these popular models. We propose the use of a SVM binary classifier, which has been proved to perform correctly when using a large number of features [20]. In our case, we choose the VGG19 architecture [21].

The model is detailed in Fig. A.2. In order to perform the feature extraction, we remove the last fully connected layer (FC) that performs the classification for the purpose of VGG19 and modify it for our specific classification task (anomaly/not anomaly in robot's route). We note that the architecture has been frozen for our case, i.e., the weights and biases in VGG19 are fixed and re-used to generate the features to feed the SVM classifier while the regularization parameter C is tuned to prevent overfitting along the training process.

4.2 Dataset format

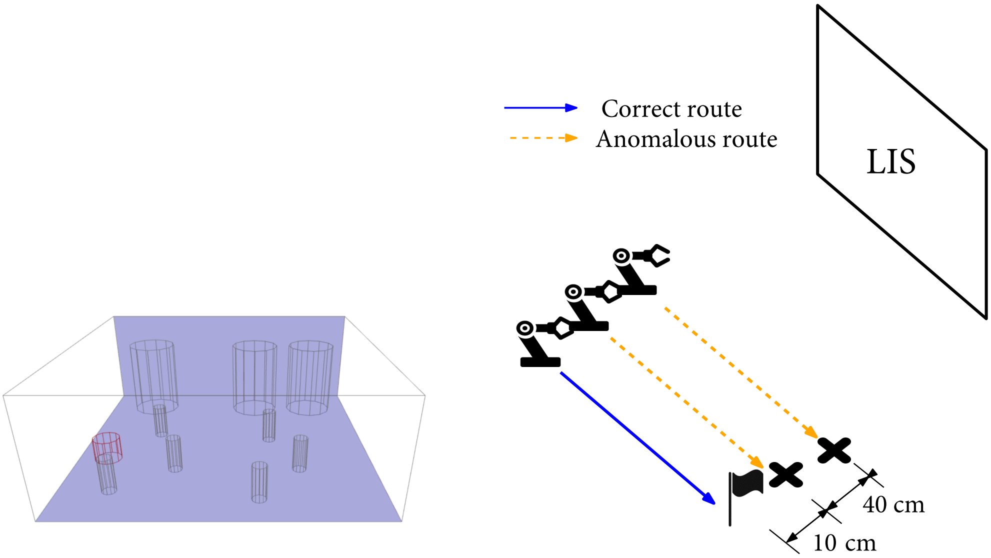
The dataset is obtained by sampling the received signal power at each element of the LIS while the robot moves along the trajectories. Formally, we can define the trajectories as the set of points in the space $\mathbf{P}_t \in \mathbb{R}^{N_p \times 3}$ being N_p the total number of points in the route. Let assume the system is able to obtain N_s samples at each channel coherence interval $\forall\ \mathbf{p}_j \in \mathbf{P}_t$, being $\mathbf{p}_j$ for $j = 1, \ldots, N_p$ an arbitrary point of the route. Hence, the dataset is conformed by $T = N_p \times N_s$ samples (monocromatic holographic image snapshots of received power). Each sample is a gray-scale image which is obtained by mapping the received power into the range of [0, 255]. To that end, we apply min-max feature scaling, in which the value of each pixel $m_{i,j}$ for $i = 1, \ldots, M$ and $j = 1, \ldots, N_p$ is obtained as

$$m_{i,j} = \left\lceil m_{\text{MIN}} + \frac{(w_{i,j} - w_{\text{MIN},j})(m_{\text{MAX}} - m_{\text{MIN}})}{w_{\text{MAX},j} - w_{\text{MIN},j}} \right\rceil, \tag{A.4}$$

where $w_{i,j}$ are the elements of $\mathbf{w}_j$ in (A.2), i.e. $w_{i,j} = \|h_{i,j} + n_{i,j}\|^2$, $m_{\text{MAX}} = 255$ and $m_{\text{MIN}} = 0$, and

$$w_{\text{MAX},j} = \max_{\{i=1,\ldots,M\}} \mathbf{w}_{i,j}, \quad w_{\text{MIN},j} = \min_{\{i=1,\ldots,M\}} \mathbf{w}_{i,j} \tag{A.5}$$

are the maximum and minimum received power value from a point $\mathbf{p}_j$ along the surface.



(a) Use case scenario.

(b) Correct robot route (blue) vs anomalous routes (orange).

Fig. A.3: Simulated scenario.

The input structure supported by VGG19 is a RGB image of $n_c = 3$ channels. Due to our monocromatic measurements, our original gray-scale input structure is a one-channel image. To solve this problem, we expand the values by copying them into a $n_c = 3$ channels input structure.

Once the feature extraction is performed, the output is $n_c = 512$ channels of size $n_w = 7$ and $n_h = 7$ pixels. Since SVM works with vectors, the data is reshaped into an input feature vector formed by $7 \times 7 \times 512 = 25088$ features, meaning our dataset is $\{x^{(i)}, y^{(i)}\}_{i=1}^{T}$, where $x^{(i)}$ is the i-th n-dimensional training input features vector (being $n = 25088$), $x_j^{(i)}$ is the value of the j-th feature, and $y^{(i)}$ is the corresponding desired output label vector.

5 Model validation

In order to validate the proposed method, we carried out an extensive set of simulations to analyze the performance of the system. To properly obtain the received power values, we use a ray tracing software, therefore capturing the effects of the multipath propagation in a reliable way. Specifically, we consider ALTAIR FEKO WINPROP [22].

5.1 Simulated scenario

The baseline set-up is described in Fig. A.3a, a small size industrial scenario of size 484 m^2. We address the detection of the deviation of the target robot (highlighted in red color) when following a fixed route parallel to the bottom wall, in which the LIS is deployed. The distance between the LIS and the desired trajectory is 13.9 m. For the anomalous routes, a separation of 50/10 cm have been simulated to analyze the performance of the system when $\Delta d >> \lambda$ and $\Delta d \approx \lambda$ respectively, as detailed in Fig. A.3b.

Table A.1: Parameters

Frequency (GHz)	Tx Power (dBm)	Nray paths	Antenna type	Antenna Spacing (cm)	Propagation model
3.5	20	20	Omni	$\frac{\lambda}{2}/\lambda/2\lambda$	Free Space

For these routes, we simulate in the ray tracing software N_p points, which corresponds to different positions of the robot in both the correct and anomalous routes. Then, N_s holographic image snapshots of the measurements are taken at every $\mathbf{p}_j$, $j = 1, \ldots, N_p$. The most relevant parameters used for simulation are summarized in Table A.1.

In our simulations, we set $N_p = 367$ and $N_s = 10$, thus the dataset is composed of $T = N_p \times N_s = 3670$ radio propagation snapshots containing images of both anomalous and non-anomalous situations, as described in Section 4.2. The dataset is split into a 80% training set and 20% for the test set. During the training phase, the obtained optimum regularization value is $C = 0.001$, which was identified by using a 5-fold cross-validation strategy [23].

5.2 Received power and noise modeling

The complex electric field arriving at the i-th antenna element at sample time t, $\widetilde{E}_i(t)$, can be regarded as the superposition of each path, i.e.[1],

$$\widetilde{E}_i(t) = \sum_{n=1}^{N_r} \widetilde{E}_{i,n}(t) = \sum_{n=1}^{N_r} E_{i,n}(t) e^{j\phi_{i,n}(t)}, \tag{A.6}$$

where N_r is the number of paths and $\widetilde{E}_{i,n}(t)$ is the complex electric field at i-th antenna from n-th path, with amplitude $E_{i,n}(t)$ and phase $\phi_{i,n}(t)$. From (A.6), and assuming

---

[1] Note that the electric field also depends on the point $\mathbf{p}_j$. However, for the sake of clarity, we drop the subindex j throughout the following subsections.

isotropic antennas, the complex signal at the output of the i-th element is therefore given by

$$y_i(t) = \sqrt{\frac{\lambda^2 Z_i}{4\pi Z_0}} \widetilde{E}_i(t) + n_i(t), \tag{A.7}$$

with λ the wavelength, $Z_0 = 120\pi$ the free space impedance, Z_i the antenna impedance, and $n_i(t)$ is complex Gaussian noise with zero mean and variance σ^2. Note that (A.7) is exactly the same model than (A.1); the only difference is that we are explicitly denoting the dependence on the sampling instant t. For simplicity, we consider $Z_i = 1 \, \forall \, i$. Thus, the power $w_i(t) = \|y_i(t)\|^2$ is used at each temporal instant t to generate the holographic image, as pointed out before. Finally, in order to test the system performance under distinct noise conditions, the average Signal-to-Noise Ratio (SNR) over the whole route, $\overline{\gamma}$, is defined as[2]

$$\overline{\gamma} \triangleq \frac{\lambda^2}{4\pi Z_0 M T \sigma^2} \sum_{t=1}^{T} \sum_{i=1}^{M} |\widetilde{E}_i(t)|^2, \tag{A.8}$$

where M denotes the number of antenna elements in the LIS.

5.3 Noise averaging strategy

Noise is critical in image classification performance [24]. Normally, in the image processing literature, noise removal techniques assume additive noise in the images [25], which is not the case in our system.

Referring to (A.1) and (A.7), since we are considering only received powers, the signal at the output of the i-th antenna detector is given by

$$w_i = \left\| \sqrt{\frac{\lambda^2 Z_i}{4\pi Z_0}} \widetilde{E}_i + n_i \right\|^2, \tag{A.9}$$

where we have dropped the dependence on t. Also, let assume the system is able to obtain S extra samples at each channel coherence interval $\forall \, \mathbf{p}_j \in \mathbf{P}$. That is, at each point $\mathbf{p}_j$, the system is able to get $N'_s = N_s \times S$ samples. Since the algorithm only expects N_s samples from each point, we can use the extra samples to reduce the noise variance at each pixel. To that end, the value of each pixel $m_{i,j}$ is not computed using directly $w_{i,j}$ as in (A.4) but instead

$$w'_{i,j} = \frac{1}{S} \sum_{s=1}^{S} w_{i,j,s}, \tag{A.10}$$

---

[2] This is equivalent to average over all the points $\mathbf{p}_j$ of the trajectory $\mathbf{P}$.

where $w_{i,j,s}$ denote the received signal power at each extra sample $s = 1, \ldots, S$. Note that, if $S \to \infty$, then

$$w'_{i,j}\big|_{S\to\infty} = \mathbb{E}[w_{i,j}|h_{i,j}] = \|h_{i,j}\|^2 + \sigma^2, \tag{A.11}$$

meaning that the noise variance at the resulting image has vanished, i.e., the received power at each antenna (conditioned on the channel) is no longer a random variable. Observe that the image preserves the pattern with the only addition of an additive constant factor σ^2. This effect is only possible if the system would be able to obtain a very large number S of samples within each channel coherence interval.

5.4 Performance metrics

To evaluate the prediction effectiveness of our proposed method, we resort on common performance metrics that are widely used in the related literature. Concretely, we are focusing on the F1-Score which is a metric based on the Precision and Recall metrics [26] and is described as:

- Positive F1-Score (PF_1) and Negative F1-Score(NF_1) as the harmonic mean of precision and recall:

$$PF_1 = 2 \cdot \frac{\text{PP} \cdot \text{RP}}{\text{PP} + \text{RP}}, \qquad NF_1 = 2 \cdot \frac{\text{PN} \cdot \text{RN}}{\text{PN} + \text{RN}}. \tag{A.12}$$

Where PP and RP stand for Precision and Recall of the positive class (anomaly) while PN and RN stand for Precision and Recall of the negative class (not anomalous situation).

6 Numerical results and Discussion

Generally, in the considered industrial setup, it would be more desirable to avoid undetected anomalies (which may indicate some error in the robot or some external issue in the predefined trajectory) than obtaining a false positive. Hence, all the figures in this section shows the algorithm performance in terms of the PF_1 metric.

6.1 Impact of sampling and noise averaging

To evaluate the impact of both sampling and noise averaging, we consider an LIS compounded by $M = 128 \times 128$ antennas and a spacing $\Delta s = \lambda/2$ for the $\Delta d = 50$ cm anomalous route.

For our particular case, $N'_s \in \{1000, 500, 100\}$. Then $\forall\, \mathbf{p}_j$ we use $S = \frac{N'_s}{N_s}$ samples for obtaining N_s S-averaged samples for training the algorithm, being still $T = N_p \times N_s =$

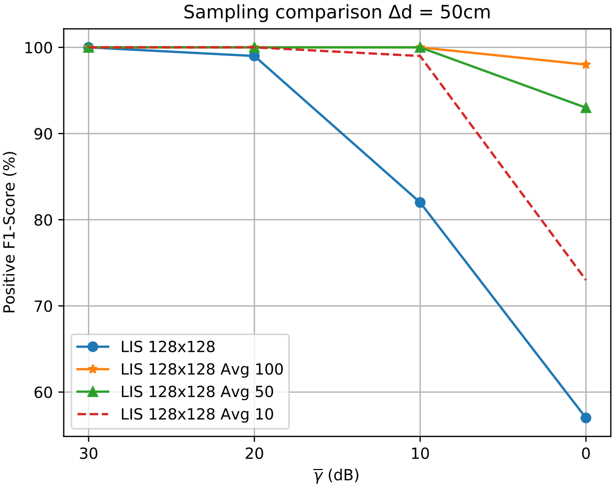


Fig. A.4: PF_1 score averaged noise vs non-averaged.

3670. Note that the number of samples N_s' would depend on the sampling frequency and the second order characterization of the channel, i.e., the channel coherence time and its autocorrelation function.

Figure A.4 shows the performance of the system when using non-averaged samples and averaged ones respectively. The blue line represents the system when non-averaged data is being used. When the noise contribution is non-negligible in the interval $\overline{\gamma} \in [10 \text{ dB}, 0 \text{ dB}]$, the detection performance presents a significant drop. Thanks to the averaging, results are significantly improved, even in the critical interval. As expected, when noise level is higher, more samples are needed to preserve the pattern by averaging, being $N_s' = 1000$ the one which yields a better performance. For the following discussions, this sampling strategy will be used, meaning we are using $S = 100$ extra samples.

6.2 Impact of antenna spacing

To evaluate the impact of inter-antenna distance, we fix the aperture to 5.44×5.44 m, we assess the performance in both $\Delta d = 50/10$ cm, and we analyze different spacings with respect to the wavelength ($\lambda/2$, λ and 2λ).

The performance results for the distinct configurations are depicted in Fig. A.5. As observed, the spacing of 2λ — which is far from the concept of LIS — is presenting really inaccurate results showing that the spatial resolution is not enough. We can

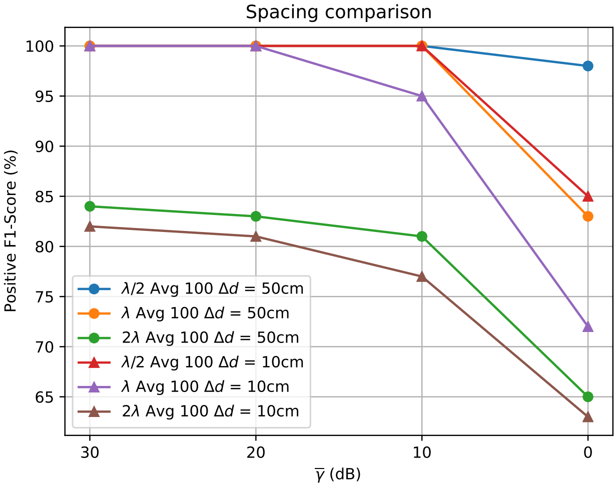


Fig. A.5: PF_1 score antenna spacing

conclude that the quick variations along the surface provide important information to the classifier performance. Besides, this information becomes more important the lower the distance between the routes is. The performance drop due to the closer distances among the routes is related to the pattern classification. The closer the routes are, the more similar the pattern is making more challenging to perform the detection. However, reducing the antenna spacing even more can improve the performance when routes are even closer. What is more, the effect of antenna densification for a given aperture is highlighted and it can be seen that the lowest spacing leads to the best results.

6.3 LIS aperture comparisons

In this case LIS with different apertures have been evaluated. The spacing is fixed to $\lambda/2$.

Looking at Fig. A.6, the aperture plays a vital role in the sensing performance. Increasing the number of antennas leads to a higher resolution image, being able to capture the large-scale events occurring in the environment more accurately. Note the usage of incoherent detectors is yielding to a good performance when the aperture is large enough. The key feature for this phenomena is the LIS pattern spatial consistency, i.e., the ability of representing the environment as a continuous measurement image.

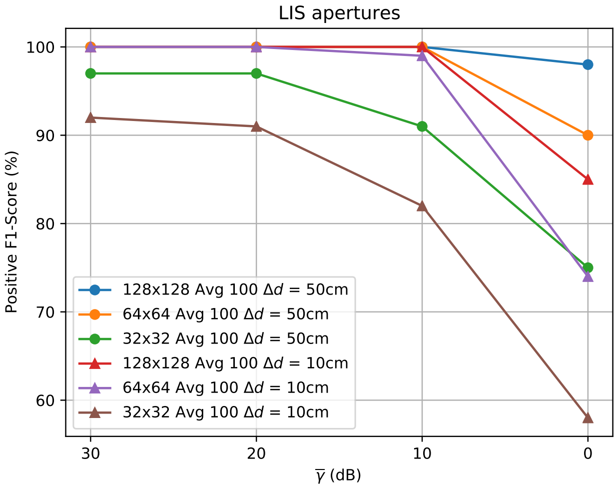


Fig. A.6: Different LISs apertures comparison

7 Conclusions

We have shown the potential of LIS for sensing the environment, being able to provide high resolution radio images of the propagation environment that can be processed by existing and versatile solutions in the context of computer vision algorithms. This sensing technique, which we consider appropriate to refer to as holographic sensing, arises as a robust solution to capture the large scale events of a target scenario, with the inherent advantage that the received signal phase does not need to be estimated. The combined usage of both LIS and machine learning algorithms may be potentially used in the context of cognitive radio and multiuser massive MIMO as a support technology to enhance the performance of these systems.

8 Acknowledgements

This project has received funding from the European Union's Horizon 2020 research and innovation programme under the Marie Sklodowska-Curie grant agreement No 813999.

References

[1] J. G. Andrews et al., “What will 5G be?” *IEEE J. Sel. Areas Commun.*, vol. 32, no. 6, pp. 1065-1082, 2014, *IEEE.*

[2] S. Hu, F. Rusek, and O. Edfors, “Beyond Massive MIMO: The Potential of Data Transmission With Large Intelligent Surfaces,” *IEEE Trans. Signal Process.*, vol. 66, no. 10, pp. 2746-2758, 2018.

[3] E. Basar, “Transmission through large intelligent surfaces: A new frontier in wireless communications,” in *EuCNC: Eur. Conf. Netw. Commun.*, pp. 112-117, 2019, *IEEE.*

[4] D. Dardari, “Communicating with Large Intelligent Surfaces: Fundamental Limits and Models,” *arXiv:1912.01719 [cs.IT]*, 2019.

[5] E. Björnson and L. Sanguinetti, “Power Scaling Laws and Near-Field Behaviors of Massive MIMO and Intelligent Reflecting Surfaces,” *arXiv:2002.04960 [cs.IT]*, 2020.

[6] H. Wang et al., “RT-Fall: A real-time and contactless fall detection system with commodity WiFi devices,” *IEEE Trans. Mobile Comput.*, vol. 16, no. 2, pp. 511–526, 2016.

[7] Q. Pu et al., “Whole-home gesture recognition using wireless signals,” in *Proc. 19th Annual Inter. Conf. Mobile Comput. & Netw.*, pp. 27–38, 2013.

[8] Y. Zhao et al., “Radio tomographic imaging and tracking of stationary and moving people via kernel distance,” in *2013 ACM/IEEE Inter. Conf. Inf. Process. Sensor Networks (IPSN)*, pp. 229–240, 2013.

[9] F. Adib, Z. Kabelac, H. Mao, D. Katabi, and R. C. Miller, “Real-time breath monitoring using wireless signals,” *Proc. 20th Annual Inter. Conf. Mobile Comput. Netw.*, pp. 261-262, 2014.

[10] M. Zhao, T. Li, M. Abu Alsheikh, Y. Tian, H. Zhao, A. Torralba, and D. Katabi, “Through-wall human pose estimation using radio signals,” *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, pp. 7356-7365, 2018.

[11] J. Wilson and N. Patwari, “Radio tomographic imaging with wireless networks,” *IEEE Trans. Mobile Comput.*, vol. 9, no. 5, pp. 621-632, 2010.

[12] J. Joung, “Machine learning-based antenna selection in wireless communications,” *IEEE Commun. Lett.*, vol. 20, no. 11, pp. 2241-2244, 2016.

[13] O. T. Demir and E. Bjornson, “Channel Estimation in Massive MIMO Under Hardware Non-Linearities: Bayesian Methods Versus Deep Learning,” *IEEE O. J. Commun. Soc.*, vol. 1, pp. 109-124, 2020.

[14] X. Ma and Z. Gao, “Data-Driven Deep Learning to Design Pilot and Channel Estimator for Massive MIMO,” *IEEE Trans. Veh. Technol.*, vol. 69, no. 5, pp. 5677-5682, 2020.

[15] H. Huang, J. Yang, H. Huang, Y. Song, and G. Gui, “Deep Learning for Super-Resolution Channel Estimation and DOA Estimation Based Massive MIMO System,” *IEEE Trans. Veh. Technol.*, vol. 67, no. 9, pp. 8549-8560, 2018.

[16] RRA Syms, “Practical Volume Holography Clarendon,” *Oxford*, vol. 19902, p. 125, 1990.

[17] S. J. Pan and Q. Yang, “A survey on transfer learning,” *IEEE Trans. Knowl. Data Eng.*, vol. 22, no. 10, pp. 1345-1359, 2009.

[18] D. Sarkar et al., “Hands-On Transfer Learning with Python: Implement advanced deep learning and neural network models using TensorFlow and Keras,” *Packt Publishing Ltd*, 2018.

[19] F. Chollet et al., *Keras*, 2015. Available online: `https://keras.io`.

[20] C. M. Bishop, *Pattern Recognition and Machine Learning*, Springer, 2006.

[21] K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” *arXiv preprint arXiv:1409.1556*, 2014.

[22] *Winprop, Altair Engineering, Inc*, Available online: `https://www.altairhyperworks.com/winprop`.

[23] D. Anguita et al., “K-Fold Cross Validation for Error Rate Estimate in Support Vector Machines,” in *DMIN*, pp. 291–297, 2009.

[24] P. Roy et al., “Effects of degradations on deep neural network architectures,” *arXiv preprint arXiv:1807.10108*, 2018.

[25] T. B. Moeslund, “Introduction to video and image processing: Building real systems and applications,” *Springer Science & Business Media*, 2012.

[26] D. M. Powers, “Recall & Precision versus The Bookmaker,” in *International Conference on Cognitive Science*, 2003.

Paper B

Assessing wireless sensing potential with large intelligent surfaces

Cristian J Vaca-Rubio, Pablo Ramirez-Espinosa, Kimmo Kansanen,
Zheng-Hua Tan, Elisabeth De Carvalho, Petar Popovski

The paper has been published in the
IEEE Open Journal of the Communications Society, vol.2, pp. 934-947, 2021.

© 2021 IEEE

The layout has been revised.

Abstract

Sensing capability is one of the most highlighted new feature of future 6G wireless networks. This paper addresses the sensing potential of Large Intelligent Surfaces (LIS) in an exemplary Industry 4.0 scenario. Besides the attention received by LIS in terms of communication aspects, it can offer a high-resolution rendering of the propagation environment. This is because, in an indoor setting, it can be placed in proximity to the sensed phenomena, while the high resolution is offered by densely spaced tiny antennas deployed over a large area. By treating an LIS as a radio image of the environment relying on the received signal power, we develop techniques to sense the environment, by leveraging the tools of image processing and machine learning. Once a radio image is obtained, a Denoising Autoencoder (DAE) network can be used for constructing a super-resolution image leading to sensing advantages not available in traditional sensing systems. Also, we derive a statistical test based on the Generalized Likelihood Ratio (GLRT) as a benchmark for the machine learning solution. We test these methods for a scenario where we need to detect whether an industrial robot deviates from a predefined route. The results show that the LIS-based sensing offers high precision and has a high application potential in indoor industrial environments[1]*.*

Keywords— Computer vision, Industry 4.0, large intelligent surfaces, machine learning, sensing

1 Introduction

MassiveMIMO is one of the essential technologies in the 5G [2]. Compared with traditional multiuser MIMO systems, the base station of a massive MIMO system is equipped with a large number of antennas, aiming to further increase spectral efficiency [3]. Looking towards the 6th generation of wireless networks (6G), there are some significant breakthroughs on the design of reprogramable metamaterials [4], giving raise to new concepts such as *holographic MIMO surfaces (HMIMO)* [5], *LIS* [6] and *reconfigurable intelligent surface (RIS)* [7]. While HMIMO and LIS originally refer to the use of continuous radiating surfaces where the received electromagnetic field is recorded and ultimately reconstructed [8], in practice an LIS is envisioned and regarded as a collection of closely spaced tiny antenna elements. On the other hand, RIS are composed by small passive reflectors embedded in a surface, allowing to arbitrarily modify the phase of the impinging electromagnetic waves and thus enabling a smart control of the propagation environment [9–11].

The performance analysis of LIS and RIS assisted systems has attracted considerable attention in the recent years, and many works have studied the applicability of these technologies. For instance, the use of RIS to control the signals propagation has been analyzed in the context of communications through the so-called passive beamforming [7, 12–14], in location and positioning systems [15–17], and in physical layer security [18, 19]. The combination of Deep Learning (DL) and RIS elements efficient reconfiguration has also been studied in [20]. In

---

[1] A preliminary version of this work has been accepted to 15-th EAI International Conference on Cognitive Radio Oriented Wireless Networks (CROWNCOM 2020) [1]

turn, LIS are considered as a natural extension of massive MIMO, and their potential for data transmission [6, 21] and positioning [22, 23] has been also addressed.

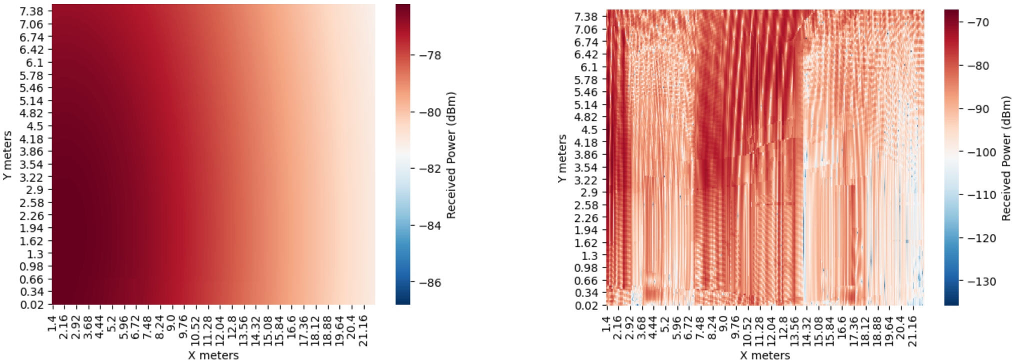
However, the potential of LIS could go beyond communications applications. Indeed, such large surfaces contain many antennas that can be used as sensors of the environment based on the CSI.

Sensing strategies based on electromagnetic signals have been thoroughly addressed in the literature in different ways, and applied to a wide range of applications. For instance, in [24], a real-time fall detection system is proposed through the analysis of the communication signals produced by active users, whilst the authors in [25] use Doppler shifts for gesture recognition. Radar-like sensing solutions are also available for user tracking [26] and real-time breath monitoring [27], as well as sensing methods based on Radio Tomographic Image (RTI) [28, 29]. Interestingly, whilst some of these techniques resort solely on the amplitude (equivalently, power) of the receive signals [26, 29], in the cases where sensing small scale variations is needed, the full CSI (i.e., amplitude and phase of the impinging signals) is required [27, 28].Moreover, in terms of power-based radio maps generation, some indoor positioning strategies leverage the use of ML solutions [30] based on the Received Signal Strength (RSS) of several beforehand known anchors for localization purposes.

On a related note, ML based approaches are gaining popularity in the context of massive MIMO, mainly due to the inherent complexity of this type of systems and their sensitivity to hardware impairments and channel estimation errors. Hence, DL techniques arise as a promising solution to deal with massive MIMO, and several works have shown the advantages of ML solutions in channel estimation and precoding [13, 31–36]. Due to the even larger dimensions of the system in extra-large arrays, DL may play a key role in exploiting complex patterns of information dependency between the transmitted signals. Also, the popularization of LIS as a natural next step from massive MIMO gives rise to larger arrays and more degrees of freedom, providing huge amount of data which can feed ML algorithms.

Despite all the available works dealing with beyond massive MIMO and sensing, both topics have been addressed rather separately. This has motivated the present work, where the objective is to assess the potential of the combined use of DL algorithms and large surfaces for the purpose of sensing the propagation environment. To that end, the received signal along with the LIS is treated as a radio image of the propagation environment, giving raise to the use of image processing techniques to improve the performance of sensing systems beyond purely radio-based approaches. Also, we analyze the pros and cons of this image sensing proposal, comparing it to alternative solutions based on classical post-processing of the received radio signal. Specifically, the contributions of this work are summarized as follows:

- We propose an image-based sensing technique based on the received signal power at each antenna element of an LIS. These power samples are processed to generate a high resolution image of the propagation environment that can be used to feed ML algorithms to sense large-scale events. The usage of received signal power would lead to simple deployments, since there is no need of coherent receivers.
- A ML algorithm, based on transfer learning and local outlier factor (LOF), is defined to process the radio images generated by the LIS in order to detect anomalies over a predefined robot route.
- We show the advantage of representing the radio propagation environment as an im-



(a) LoS, noiseless. **(b)** Real scenario, noiseless.

Fig. B.1: Radio images for LoS and Industry scenarios.

age, allowing us to use a denoising autoencoder network (DAE) for augmenting image resolution and significantly increasing the performance of the system.

- We derive a statistical test, based on the classical generalized likelihood ratio test (GLRT), to carry out the same sensing task, and perform a comparison with the ML solution in terms of generality, performance and further potential applications.

To evaluate the capabilities for sensing of LIS, we consider a simple problem of route anomaly detection in an indoor industrial scenario. Hence, we analyze the feasibility of this proposal to determine whether a robot has deviated from its predefined route, and compared it with the here derived statistical solution.

The reminder of the paper is organized as follows. Section 2 introduces the concept of sensing based on radio images. Then, Section 3 presents the problem of robot deviation detection in an industrial setup. The classical solution based on hypothesis testing is derived and characterized in Section 4 and the proposed ML algorithm is detailed in Section 5. With the ML solution presented, the model validation is carried out in Section 6, whilst simulated results are discussed in Section 7. Finally, some conclusions are drawn in Section 8.

2 Radio image-based LIS sensing

In a wireless context, a LIS could be described as a structure which uses electromagnetic signals impinging in a determined scatterer in order to obtain a profile of the environment. That is, we can use the resulting signal of the superposition of all the involved paths that imping into every of the antenna elements conforming the surface. Then, the power of the resulting superimposed signal is used to obtain a high resolution image of the propagation environment. Note the LIS elements are using the CSI information as envelope detectors, as no phase estimation is needed but the recieved signal power. Using this approach, the complexity of the multipath propagation is reduced to using information represented as an image. This provides a twofold benefit: *i)* the massive number of elements that compose the LIS leads to an accurate environment sensing (i.e. high resolution image), and *ii)* it allows the use of computer vision algorithms and ML techniques to deal with the resulting images.

As an illustrative example, an LIS is deployed in a wall along a 22×8 m physical aperture, containing antenna elements separated $\frac{\lambda}{2}$ cm while an arbitrary robot is transmitting a sensing signal. Fig. B.1 shows the LIS radio images obtained from different propagation environments under this setup. Specifically, Fig. B.1a corresponds to an LoS propagation (no scatterers), whilst Fig. B.1b is obtained from an industrial scenario with a rich scattering. Note that, in the case in which different scatterers are placed, their position and shapes are captured by the LIS and represented in the image. Beyond that, LIS-based imaging does not need of a previous calibration period as well as no scatterers need to be modelled to be captured in the radio image, contrary to other wireless image reconstruction techniques that rely on the received signal power, such as RTI [28, 29]. To the best of the authors' knowledge, this is the first time that LIS image based wireless sensing is proposed in the literature.

3 System Model and Problem Formulation

We stick to a simple baseline problem in order to analyze, for first time in the literature, the sensing potential of LIS. To that end, let consider an industrial scenario where a robot is supposed to follow a fixed route. Assume that, due to arbitrary reasons, it might temporarily deviate from the predefined route and follow an alternative (undesired) trajectory. The goal is to be able to detect whether the robot is following the correct route or not, based on the sensing signal transmitted by the target device.

In order to perform the detection, we assume that a LIS, consisting of M antenna elements, is placed along a wall. The sensing problem reduces to determine, from the received signal at each of the M LIS elements, whether the transmission has been made from a point at the desired route or from anomalous ones. Formally, if we define the correct trajectory as the set of points in space $\mathbf{P}_c \in \mathbb{R}^{N_p \times 3} = (\mathbf{p}_1, \ldots, \mathbf{p}_{N_p})$, and the received complex signal from an arbitrary point, $\mathbf{p}_k \in \mathbb{R}^{1\times 3}$, as $\mathbf{y}_k \in \mathbf{C}^{M \times 1}$, then the problem reduces to estimating whether $\mathbf{p}_k \in \mathbf{P}_c$ based on $\mathbf{y}_k$. Note that this formulation can be generalized to any anomalous detection based on radio-waves in an arbitrary scenario.

The complex baseband signal received at the LIS from point $\mathbf{p}$ (the subindex is dropped for the sake of clarity in the notation) is given by

$$\mathbf{y} = \mathbf{h}x + \mathbf{n}, \tag{B.1}$$

with x the transmitted (sensing) symbol, $\mathbf{h} \in \mathbb{C}^{M\times 1}$ the channel vector from point $\mathbf{p}$ to each antenna-element, and $\mathbf{n} \sim \mathcal{CN}_M(\mathbf{0}, \sigma^2 \mathbf{I}_M)$ the noise vector. Moreover, we consider a static scenario where the the channel $\mathbf{h}$ only depends on the user position, neglecting the impact of time variations.

In order to reduce deployment costs, and because we are interested in sensing large scale variations, we consider the received signal amplitude (equivalently, power). This assumption may lead to cheaper and simpler system implementations, avoiding the necessity of performing coherent detection.

4 Statistical Approach: Likelihood Ratio Test

4.1 Decision rule

Let us consider that the system is able to obtain several samples from each point $\mathbf{p}_k$ belonging to the correct route $\mathbf{P}_c$ during a training phase. Then, once the system is trained, the problem can be tackled from a statistical viewpoint by performing a generalized hypothesis test, as shown throughout this section.

To start with, let us assume that the value of σ^2 in (B.1) is perfectly known and, without loss of generality, that $x = 1$. Since we are considering only received powers, the signal at the output of the i-th antenna detector is given by

$$w_i = \|y_i\|^2 = \|h_i + n_i\|^2, \tag{B.2}$$

where y_i, h_i and n_i for $i = 1, \dots, M$ are the elements of $\mathbf{y}$, $\mathbf{h}$ and $\mathbf{n}$, respectively. When conditioned on h_i, w_i follows a Rician distribution (in power) [37], and due to the statistical independence of the noise samples, the joint conditioned probability density function (PDF) of the vector $\mathbf{w} = (w_1, \dots, w_M)^T$ is given by the product of the individual PDFs.

Consider also that, during the training phase, N_t samples of $\mathbf{w}_0$, namely $\mathbf{w}_{0,j}$ for $j = 1, \dots, N_t$, are obtained from a correct (trained) point $\mathbf{p}_0$. The samples $\mathbf{w}_{0,j}$ are then jointly Rician distributed with vector parameter[2] $\mathbf{g}_0 = (\|h_{0,1}\|^2, \dots, \|h_{0,M}\|^2)^T$. Then, from $\mathbf{w}_{0,j}$, the system obtains an estimation $\widehat{\mathbf{g}}_0 = (\widehat{g}_{0,1}, \dots, \widehat{g}_{0,M})^T$ of $\mathbf{g}_0$.

Once trained (evaluation phase), the LIS receives another set of samples $\mathbf{w}_k$ for $k = 1, \dots, N_v$ from an arbitrary point $\mathbf{p}$. Therefore, the objective is, based on $\mathbf{w}_{0,j}$ and $\mathbf{w}_k$, to determine whether $\mathbf{p} = \mathbf{p}_0$ or not. To that end, we formulate a binary hypothesis test as

$$\begin{cases} H_0 : & \widehat{\mathbf{g}} = \widehat{\mathbf{g}}_0 \\ H_1 : & \widehat{\mathbf{g}} \neq \widehat{\mathbf{g}}_0 \end{cases}, \tag{B.3}$$

where $\widehat{\mathbf{g}} = (\widehat{g}_i, \dots, \widehat{g}_M)^T$ is the channel vector estimated from $\mathbf{w}_k$. The test is hence formulated based on the GLRT, but replacing the knowledge of the null hypothesis by its estimated counterpart, i.e.,

$$\log\Lambda = \sum_{k=1}^{N_v}\sum_{i=1}^{M} \log I_0\left(\frac{2\widehat{g}_{0,i}\sqrt{w_{i,k}}}{\sigma^2}\right) + \sum_{k=1}^{N_v}\sum_{i=1}^{M} \frac{\widehat{g}_i - \widehat{g}_{0,i}}{\sigma^2} - \sum_{k=1}^{N_v}\sum_{i=1}^{M} \log I_0\left(\frac{2\widehat{g}_i\sqrt{w_{i,k}}}{\sigma^2}\right) \underset{H_1}{\overset{H_0}{\gtrless}} \eta, \tag{B.4}$$

where $w_{i,k}$ denote the i-th entry of $\mathbf{w}_k$. Replacing the true value of $\mathbf{g}_0$ by its estimation introduces a non-negligible error in the test that has to be considered in the threshold design, as we will see in the following subsections.

---

[2] Note that, due to the circular symmetry of the noise, the distribution of $\mathbf{w}$ does not depend on the complex channel h_i but on its squared modulus $g_i = \|h_i\|^2$.

4.2 Estimator for $\mathbf{g}$

In conventional likelihood ratio tests, the estimation of the involved parameters is carried out through maximum likelihood. However, since in our problem the distribution of the received power signal $\mathbf{w}_k \ \forall \ k$ is a multivariate Rician, the maximum likelihood estimation implies solving a system of M non-linear equations [38]. This may lead to a considerable computational effort taking into account the large number of antennas (M) that characterizes the LIS. To circumvent this issue, we proposed a suboptimal — albeit unbiased — estimator based on moments.

Since $\mathbb{E}[\mathbf{n}\mathbf{n}^H] = \mathbf{I}_M$, the estimation of the channel at each antenna element can be solved separately. Then, we can estimate g_i in both the training and evaluation phases as

$$\widehat{g}_{0,i} = \frac{1}{N_t}\sum_{j=1}^{N_t} w_{0,i,j} - \sigma^2, \qquad \widehat{g}_i = \frac{1}{N_v}\sum_{k=1}^{N_v} w_{i,k} - \sigma^2, \tag{B.5}$$

where $w_{0,i,j}$ are the elements of $\mathbf{w}_{0,j}$. It is easily proved that the estimators in (B.5) are unbiased with normally distributed error for relatively large number of samples.

4.3 Threshold design

Although the asymptotic properties of $\log\Lambda$ have been well studied in the literature (see, e.g., [39]), these general results are not valid in our case because *i)* we are replacing the true value of $\mathbf{g}_0$ by its estimation, and *ii)* we are using moment-based estimators instead of the optimal one. A more general result, which is the starting point of our derivation, is that the the limiting distribution of $-2\log\Lambda$, under the null hypothesis, is given by [40, eq. (4.3)]

$$-2\log\Lambda|_{H_0} \xrightarrow{p} (\widehat{\mathbf{g}}_0 - \widehat{\mathbf{g}})^T N_v \mathbf{J}(\widehat{\mathbf{g}}_0 - \widehat{\mathbf{g}}), \tag{B.6}$$

where we have replaced $\mathbf{g}_0$ by $\widehat{\mathbf{g}}_0$. In (B.6), $\xrightarrow{p}$ stands for convergence in probability and $\mathbf{J} \in \mathbb{R}^{M\times M}$ is the Fisher information matrix of $\mathbf{w}_k$ with respect to $\mathbf{g}_0$ [41]. In our case, $\mathbf{J}$ is a diagonal matrix whose i-th element is given by

$$J_i(g_i) = \frac{e^{-g_i/\sigma^2}}{\sigma^6 g_i}\int_0^\infty w_i e^{-w_i/\sigma^2}\frac{I_1^2\left(\frac{2}{\sigma^2}g_i\sqrt{w_i}\right)}{I_0\left(\frac{2}{\sigma^2}g_i\sqrt{w_i}\right)}dw_i - \frac{1}{\sigma^4}. \tag{B.7}$$

Eq. (B.6) can be rewritten as

$$-2\log\Lambda|_{H_0} \xrightarrow{p} (\boldsymbol{\epsilon}_0 - \boldsymbol{\epsilon})^T N_v \mathbf{J}(\boldsymbol{\epsilon}_0 - \boldsymbol{\epsilon}), \tag{B.8}$$

where $\boldsymbol{\epsilon}_0 = (\epsilon_{0,1},\dots,\epsilon_{0,M})^T$ and $\boldsymbol{\epsilon} = (\epsilon_1,\dots,\epsilon_M)^T$ are the error vectors of estimators in (B.5). Note that both error vectors are Gaussian distributed, but they vary at very different time scales. The true channel $\mathbf{g}_0$ is estimated during the training phase, and thus the error $\boldsymbol{\epsilon}_0$, albeit random, remains constant during the whole evaluation phase until the system is retrained. In turn, each time the system evaluates a point, $\boldsymbol{\epsilon}$ takes a different (random) realization. With that in mind, we propose choosing η based on a worst case design, i.e., we consider an estimation error $\boldsymbol{\epsilon}_0$ that overestimates the true error at $1-\alpha_0$ percent of the time. That is,

$$\epsilon'_{0,i} = F^{-1}_{\epsilon_{0,i}}(1-\alpha_0/2), \tag{B.9}$$

where $F_{\epsilon_{0,i}}$ stands for the cumulative distribution function (CDF) of a Gaussian variable with zero mean and variance $\frac{\sigma^2}{N_t}(\sigma^2 + 2\widehat{g}_{0,i})$. Note that we have replaced the true channel value by its estimation in the calculation of the aforementioned percentile.

Finally, conditioned on $\epsilon'_{0,i}$, the distribution of the test for large number of samples is given by

$$-2\log\,\Lambda|_{H_0,\epsilon'_0} \xrightarrow{p} D = \sum_{i=1}^{M} N_v J_i(g_i)(\epsilon_i - \epsilon'_{0,i})^2, \tag{B.10}$$

which corresponds to a non-central Gaussian quadratic form. Therefore, given a predefined false alarm probability α, the test finally reads as

$$-2\log\Lambda \underset{H_0}{\overset{H_1}{\gtrless}} -2\eta = F_D^{-1}(1-\alpha|\widehat{\mathbf{g}}_0), \tag{B.11}$$

where F_D is the CDF of D, which can be obtained by Monte-Carlo simulations or by using some of the approximations given in the literature for Gaussian quadratic forms (see, e.g., [42–44]). Note that, in (B.11), we have again replaced the true channel values by their estimations. In our proofs, this seems to have a negligible impact on the threshold distribution unless the number of samples is very low (in which case the asymptotic analysis here presented does no longer hold). A summary of the proposed statistical test is provided in Algorithm 1, where the here presented pointwise comparison is performed along the whole route $\mathbf{P}_c$.

Algorithm 1: Statistical test for sensing

```
Training phase:
for each $\mathbf{p} \in \mathbf{P}_c$ do
    I. Estimate $\widehat{\mathbf{g}}_0$ using (B.5)
    II. Compute $\epsilon'_{0,i}$ for $i = 1, \dots, M$ from (B.9) for a confidence value $\alpha_0$
    III. Compute $J(\widehat{g}_{0,i})$ for $i = 1, \dots, M$ from (B.7)
    IV. Compute $-2\eta$ using (B.11) for a confidence value $\alpha$
end
Evaluation phase: Received $\mathbf{w}_j$ for $j = 1, \dots, N_v$, do
for each $\mathbf{p} \in \mathbf{P}_c$ do
    I. Estimate $\widehat{\mathbf{g}}$ using (B.5)
    II. Compute $-2\log\Lambda$ using (B.4)
    III. Reject $H_0$ if $-2\log\Lambda > -2\eta$
end
The point does not belong to $\mathbf{P}_c$ if $H_0$ is rejected $\forall$ $\mathbf{p}$
```

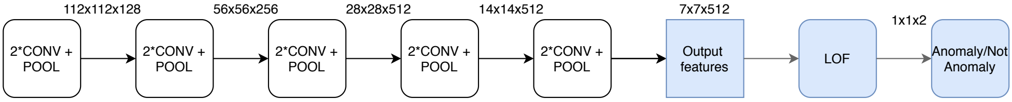


Fig. B.2: Proposed model. White and blue blocks refer to VGG19 re-used original architecture and to the additional blocks for our task, respectively.

5 Machine learning for radio image-based LIS sensing

In the previous section, we have presented a statistical method to sense the environment based on the received power signal at the different antenna elements of the LIS, and hence detecting route deviations from a predefined correct trajectory. This approach exploits the large number of antennas in the LIS in the same way as in massive MIMO systems. However, the high spatial density of antennas and the large array aperture of LIS can be exploited in an alternative way. The basis of this novel technique is using the power of the received signal across the surface as a radio image of the environment, as stated in Sec. 2.

5.1 Model description

We introduce a ML model to perform the anomalous route classification task based on the radio-based images obtained at the LIS. The main advantage of this proposal, as we will see, is that it is independent on the data distribution, and no assumptions are needed to its implementation. This is in contrast with section 4 where we considered the noise is Gaussian distributed with noise variance known for the sake of simplicity in the analytical derivations. In reality, these assumptions may not hold.

In our considered problem, the training data is obtained by sampling the received power at certain temporal instants while the target device is moving along the correct route. In order to reduce both training time and scanning periods, which may be heavy tasks for large trajectories, we resort on transfer learning [45]. Because of this matter, the risk of overfitting due to our constraint of short scanning periods is quite significant, being transfer learning also a proper way to tackle it. This allows using a small dataset and therefore improving the flexibility of the system in real world deployments. In our case, we choose the VGG19 architecture [46]. Due to our specific use case, and the training data constraints, we propose the use of an unsupervised ML algorithm named as LOF which identifies the outliers presents in a dataset (i.e., the anomalous positions of the target robot) [47].

The proposed model is detailed in Fig. B.2, where, in order to perform the feature extraction, we remove the last FC that performs the classification for the purpose of VGG19 and modify it for our specific classification task (anomaly/not anomaly in robot's route).

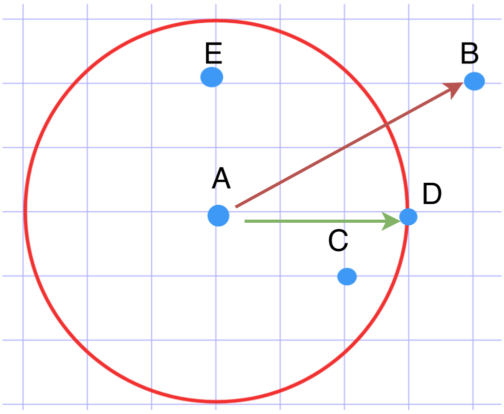


Fig. B.3: Illustration of Reachability Distance with $K = 3$. Manhattan distance used for illustration purposes.

5.2 Local Outlier Factor

LOF is an unsupervised ML algorithm that relies on the density of data points in the distribution as a key factor to detect outliers (i.e., anomalous events). In the context of anomaly detection, LOF was proposed in [47] as a solution to find anomalous data points by measuring the local deviation of a given point with respect to its neighbors.

LOF is based on the concept of local density, where the region that compounds a *locality* is determined by its K nearest neighbors. By comparing the local density of a point to the local densities of its neighbors, one can identify regions of similar density, and points that have a substantially lower density than their neighbors (the latter are considered to be outliers). This approach can be naturally applied to the anomalous trajectory deviation detection as deviated points that are really close to the correct trajectory could be really close in distance, but they would have a lower density compared with the points that actually belong to the correct one, being accurately detected as deviations. Hence, the points belonging to the correct route are used to learn the correct clusters. The strength of the LOF algorithm is that it takes both local and global properties of datasets into consideration, i.e., it can perform well even in datasets where anomalous samples have different underlying densities because the comparison is carried out with respect to the surrounding neighborhood. For the reader's convenience, a brief description of the LOF theory is provided in the following[3].

The algorithm is based on two metrics, namely the K-distance of a point A, denoted by $D_K(A)$, and its K-neighbors, which is the set $N_K(A)$ composed by those points that are in or on the circle of radius D_K with respect to the point A. Note that K is a hyperparameter to be chosen and fixed for computing the clusters. Also note that this implies $|N_K(A)| \geq K$, where $|N_K(A)|$ is the number of points in $N_K(A)$. With these two quantities, the reachability

---

[3] For a more detailed description, the reader is gently referred to [47].

distance between two arbitrary points A and B is defined as

$$RD_K(A,B) = \max\{D_K(A), d(A,B)\}, \tag{B.12}$$

where $d(A,B)$ is the distance between points A and B. Figure B.3 illustrates the RD_K concept. This means that if a point B lies within the K-neighbors of A, the reachability distance will be $D_K(A) = 3$ (the radius of the circle containing points C, D and E), else the reachability distance will be the distance between A and B. In the example, $RD_K(A,B) = 6$.

Note that the distance measure is problem-specific, being in our case the Euclidean distance between the different features extracted by the VGG19 network. From (B.12), the local reachability density (LRD) of a point A is defined as the inverse of the average reachability distance of A from its neighbors, i.e.,

$$LRD_K(A) = \left(\sum_{B \in N_K(A)} \frac{RD_K(A,B)}{|N_K(A)|} \right)^{-1}. \tag{B.13}$$

According to the LRD, if neighbors are far from the point (i.e. more the average reachability distance), less density of points are presented around a particular point. Note this would be the distance at which the point A can be reached from its neighbors, meaning this measures how far a point is from the nearest cluster of points, acquiring low values of LRD when the closest cluster is far from the point. This finally give rise to the concept of LOF, which is given by

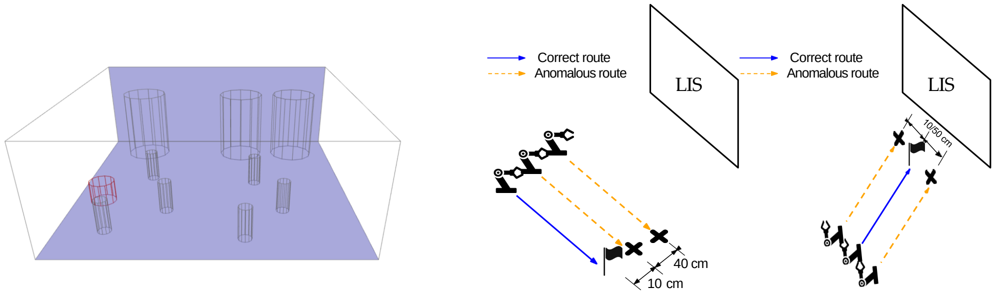
$$LOF_K(A) = \frac{\sum_{B \in N_K(A)} LRD_K(B)}{LRD_K(A)|N_K(A)|}. \tag{B.14}$$

Observe that, if a point is an inlier, the ratio is approximately equal to one because the density of a point and its neighbors are practically equal. In turn, if the point is an outlier, its LRD would be less than the average LRD of neighbors, and hence the LOF would take large values. In our specific problem, we propose using the LOF values to determine whether a point belong to the correct trajectory or from any other point due to a robot deviation.

5.3 Dataset format

With the algorithm and the model introduced, the remaining component to fully characterize the proposed ML solution is the dataset. In our considered problem, the dataset is obtained by sampling the received signal power at each element of the LIS while the robot moves along the trajectories. Formally, we can define the possible trajectories (including those composed by both correct and anomalous points) as the set of points in the space $\mathbf{P}_t \in \mathbb{R}^{N_p \times 3}$ being N_p the total number of points in the route. Let us assume the system is able to obtain N_s samples at each channel coherence interval $\forall$ $\mathbf{p}_j \in \mathbf{P}_t$, being $\mathbf{p}_j$ for $j = 1, \ldots, N_p$ an arbitrary point of the route. Hence, the dataset consists of $T = N_p \times N_s$ samples (monocromatic radio image snapshots of received power). Each sample is a gray-scale image which is obtained by mapping the received power into the range of [0, 255]. To that end, we apply min-max feature scaling, in which the value of each pixel $m_{i,j}$ for $i = 1, \ldots, M$ and $j = 1, \ldots, N_p$ is obtained as

$$m_{i,j} = \left\lceil m_{\text{MIN}} + \frac{(w_{i,j} - w_{\text{MIN},j})(m_{\text{MAX}} - m_{\text{MIN}})}{w_{\text{MAX},j} - w_{\text{MIN},j}} \right\rceil, \tag{B.15}$$



(a) Use case scenario. **(b)** Parallel deviation. **(c)** Normal deviation.

Fig. B.4: Simulated scenario.

where $w_{i,j}$ are the elements of $\mathbf{w}_j$, i.e. $w_{i,j} = \|h_{i,j} + n_{i,j}\|^2$, $m_{\text{MAX}} = 255$ and $m_{\text{MIN}} = 0$, and

$$w_{\text{MAX},j} = \max_{\{i=1,\dots,M\}} \mathbf{w}_{i,j}, \quad w_{\text{MIN},j} = \min_{\{i=1,\dots,M\}} \mathbf{w}_{i,j} \tag{B.16}$$

are the maximum and minimum received power value from a point $\mathbf{p}_j$ along the surface.

The input structure supported by VGG19 is a RGB image of $n_c = 3$ channels. Due to our monocromatic measurements, our original gray-scale input structure is a one-channel image. To solve this problem, we expand the values by copying them into a $n_c = 3$ channels input structure.

Once the feature extraction is performed, the output is $n_c = 512$ channels of size $n_w = 7$ and $n_h = 7$ pixels. Since LOF works with vectors, the data is reshaped into an input feature vector formed by $7 \times 7 \times 512 = 25088$ dimensions, meaning our dataset is $\{x^{(i)}\}_{i=1}^{T}$, where $x^{(i)}$ is the i-th n-dimensional training input features vector (being $n = 25088$) and $x_j^{(i)}$ is the value of the j-th feature.

6 Model validation

In order to validate both sensing solutions, namely the statistical hypothesis testing and the radio-based image sensing algorithm, we carried out an extensive set of simulations to analyze the performance of the systems in a simple, yet interesting, industrial scenario. To properly obtain the received power values, we use a ray tracing software, therefore capturing the effects of the multipath propagation in a reliable way. Specifically, we consider ALTAIR FEKO WINPROP [48].

6.1 Simulated scenario

The baseline set-up is described in Fig. B.4a, a small size industrial scenario of size 484 m^2. We address the detection of the deviation of the target robot (highlighted in red color) in 2 cases: *i)* it follows a fixed route parallel (Fig. B.4b), and *ii)* the correct route is normal to the bottom wall, in which the LIS is deployed (Fig. B.4c). To evaluate the performance in the detection of anomalies, we consider that the robot may deviate from the correct route at any point, and we test the ability of both systems to detect potential deviations at a distance of, at least, $\Delta d = 50$ cm and $\Delta d = 10$ cm. These two distances correspond to the cases $\Delta d \gg \lambda$ and $\Delta d \approx \lambda$, respectively, denoting λ the wavelength.

Table B.1: Parameters

Frequency (GHz)	Tx Power (dBm)	Nray paths	Antenna type	Antenna Spacing (cm)	Propagation model
3.5	20	10	Omni	$\frac{\lambda}{2}$	Free Space

For the aforementioned cases, we simulate in the ray tracing software N_p points, which correspond to different positions of the robot in both the correct and anomalous routes. Then, N_s radio image snapshots of the measurements are taken at every $\mathbf{p}_j$, $j = 1, \ldots, N_p$. The most relevant parameters used for simulation are summarized in Table B.1.

In our simulations, we set $N_p = 367$ and $N_s = 10$, thus the dataset is composed of $T = N_p \times N_s = 3670$ radio propagation snapshots containing images of both anomalous and non-anomalous situations, as described in Section 5.3. Out of $N_p = 367$, $N_c = 185$ are the snapshots corresponding to the correct route, meaning we have $T_c = N_c \times N_s = 1850$ correct data samples, while the remaining are anomalous points. To train the algorithm with the correct points, we split the correct dataset into a 80% training set 10% validation set and the remaining 10% for the test set. During the training phase, the optimum value of $K = 3$ (the LOF parameter) is obtained by maximizing the accuracy score in the correct validation set. The training procedure was performed in an Intel Xeon machine with 32 CPUs taking around 15 seconds in the offline scanning period.

6.2 Received power and noise modeling

The complex electric field arriving at the i-th antenna element at sample time t, $\widetilde{E}_i(t)$, can be regarded as the superposition of each path, i.e.[4],

$$\widetilde{E}_i(t) = \sum_{n=1}^{N_r} \widetilde{E}_{i,n}(t) = \sum_{n=1}^{N_r} E_{i,n}(t) e^{j\phi_{i,n}(t)}, \tag{B.17}$$

where N_r is the number of paths and $\widetilde{E}_{i,n}(t)$ is the complex electric field at i-th antenna from n-th path, with amplitude $E_{i,n}(t)$ and phase $\phi_{i,n}(t)$. From (B.17), and assuming isotropic antennas, the complex signal at the output of the i-th element is therefore given by

$$y_i(t) = \sqrt{\frac{\lambda^2 Z_i}{4\pi Z_0}} \widetilde{E}_i(t) + n_i(t), \tag{B.18}$$

---

[4] Note that the electric field also depends on the point $\mathbf{p}_j$. However, for the sake of clarity, we drop the subindex j throughout the following subsections.

with λ the wavelength, $Z_0 = 120\pi$ the free space impedance, Z_i the antenna impedance, and $n_i(t)$ is complex Gaussian noise with zero mean and variance σ^2. Note that (B.18) is exactly the same model than (B.1); the only difference is that we are explicitly denoting the dependence on the sampling instant t. For simplicity, we consider $Z_i = 1\,\forall\, i$. Thus, the power $w_i(t) = \|y_i(t)\|^2$ is used at each temporal instant t both to perform the hypothesis testing in (B.4) and to generate the radio image, as pointed out before. Finally, in order to test the system performance under distinct noise conditions, the average SNR over the whole route, $\overline{\gamma}$, is defined as[5]

$$\overline{\gamma} \triangleq \frac{\lambda^2}{4\pi Z_0 M T \sigma^2} \sum_{t=1}^{T} \sum_{i=1}^{M} |\widetilde{E}_i(t)|^2, \tag{B.19}$$

where M denotes the number of antenna elements in the LIS.

6.3 Noise averaging strategy

The statistical solution presented in Section 4 has been derived taking into account the presence of noise, and consequently it has implicit mechanisms to reduce its impact in the performance. However, the presence of noise may be more critical in the radio image sensing, since it impacts considerably the image classification performance [49].

Referring to (B.2) and (B.18), since we are considering only received powers, the signal at the output of the i-th antenna detector is given by

$$w_i = \left\| \sqrt{\frac{\lambda^2 Z_i}{4\pi Z_0}} \widetilde{E}_i + n_i \right\|^2, \tag{B.20}$$

where we have dropped the dependence on t. Also, let us assume the system is able to obtain S extra samples at each channel coherence interval $\forall\, \mathbf{p}_j \in \mathbf{P}_t$. That is, at each point $\mathbf{p}_j$, the system is able to get $N_s' = N_s \times S$ samples. Since the algorithm only expects N_s samples from each point, we can use the extra samples to reduce the noise variance at each pixel. To that end, the value of each pixel $m_{i,j}$ is not computed using directly $w_{i,j}$ as in (B.15) but instead

$$w'_{i,j} = \frac{1}{S} \sum_{s=1}^{S} w_{i,j,s}, \tag{B.21}$$

where $w_{i,j,s}$ denote the received signal power at each extra sample $s = 1, \ldots, S$. Note that, if $S \to \infty$, then

$$w'_{i,j}\Big|_{S\to\infty} = \mathbb{E}[w_{i,j}|h_{i,j}] = \|h_{i,j}\|^2 + \sigma^2, \tag{B.22}$$

meaning that the noise variance at the resulting image has vanished, i.e., the received power at each antenna (conditioned on the channel) is no longer a random variable. Observe that the image preserves the pattern with the only addition of an additive constant factor σ^2. This effect is only possible if the system would be able to obtain a very large number S of samples within each channel coherence interval.

---

[5] This is equivalent to averaging over all the points $\mathbf{p}_j$ of the trajectory $\mathbf{P}$.

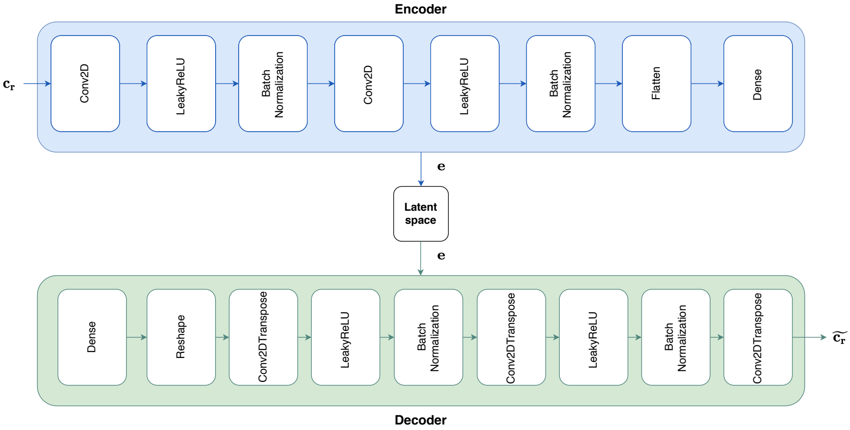


Fig. B.5: Denoising autoencoder architecture

6.4 Stacked Denoising Autoencoder for image Super-Resolution

An autoencoder is a type of neural network which tries to learn a representation (denoted as encoding) from the data given as input, usually for dimensionality reduction purposes. Along with the encoding side, a reconstructing side is learnt, where the autoencoder tries to reconstruct from the reduced encoded data a representation as close as possible to its original input. There are several variations of the basic model in order to enforce the learned representation to fulfill some properties [50]. Among all of them, we are interested in the DAEs [51].

The goal of the denoising autoencoder is to reconstruct a "clean" target version from a corrupted input. In our context, let us assume that we can obtain a target image $\mathbf{t} \in \mathbb{N} \cap [0,255]^N$ and a corrupted input $\mathbf{c} \in \mathbb{N} \cap [0,255]^M$ result of the received power mapping explained in (B.15). Also, consider $N \gg M$ and that $\mathbf{t}$ was obtained from a less noisy environment, i.e., the average SNR $\overline{\gamma}_t$ is greater than that of $\mathbf{c}$, denoted by $\overline{\gamma}_t \geq \overline{\gamma}_c$. Then, we can perform a resizing of both images such as $R : [0,255]^d \to \mathbb{R} \cap [0,1]^R$, meaning we resize both images towards the same dimension R and we normalize the values dividing by 255, being $\mathbf{t_r} = R(\mathbf{t})$ and $\mathbf{c_r} = R(\mathbf{c})$ the target and corrupted input used to train our DAE. Note that, although the two images ($\mathbf{t_r}$ and $\mathbf{c_r}$) are identical in dimension (R pixels) after the resizing procedure, the resolution of the target one is higher because it is obtained in a more favorable scenario (larger SNR) and from an initially higher number of pixels N.

To illustrate the approach, a one hidden layer explanation is made for simplicity. The denoising autoencoder can be split into two components: the encoder and the decoder, which can be defined as transformations Φ and Ψ such that $\Phi : \mathbb{R} \cap [0,1]^R \to \mathcal{F}$, $\Psi : \mathcal{F} \to \mathbb{R} \cap [0,1]^R$. Then, the encoder stage of the DAE takes the input $\mathbf{c_r}$ and maps it into $\mathbf{e} \in \mathbb{R}^l = \mathcal{F}$ as

$$\mathbf{e} = \rho(\mathbf{W}\mathbf{c_r} + \mathbf{b}), \tag{B.23}$$

being l the dimension of the compressed representation of the data, known as the latent space, ρ the element-wise activation function, $\mathbf{W}$ the weight matrix and $\mathbf{b}$ the bias vector. These weights and biases are randomly initialized and updated iteratively through backpropagation. After this process, the decoder stage maps $\mathbf{e}$ to the reconstruction $\widetilde{\mathbf{c_r}}$ of the same shape as $\mathbf{c_r}$

$$\widetilde{\mathbf{c_r}} = \rho(\mathbf{W}'\mathbf{e} + \mathbf{b}'), \tag{B.24}$$

being ρ the activation function, and $\mathbf{W}'$ and $\mathbf{b}'$ the parameters used in the decoder part of the network. In our specific case, the reconstruction error, also known as loss, is given by the mean squared error (MSE) of the pixel values of the target image and the reconstructed image (R pixels), that is[6]

$$\begin{aligned}\mathcal{L}(\mathbf{t_r}, \widetilde{\mathbf{c_r}}) &= \frac{\sum \|\mathbf{t_r} - \widetilde{\mathbf{c_r}}\|^2}{R} \\ &= \frac{\sum \|\mathbf{t_r} - \sigma(\mathbf{W}'\sigma(\mathbf{W}\mathbf{c_r} + \mathbf{b}) + \mathbf{b}')\|^2}{R}.\end{aligned} \tag{B.25}$$

For the sake of reproducibility, a detailed summary of the proposed architecture is provided in Figure B.5. We have used the Keras [52] library, so the description of the layers corresponds to its notation. The ADAM optimizer with a learning rate $\alpha = 0.001$, exponential decay for the 1st and 2nd moment estimates $\beta_1 = 0.9$ and $\beta_2 = 0.999$ and $\epsilon = 10^{-7}$ have been used for updating the gradient, minimizing the MSE loss function. For the encoder part, Conv2D layers with filter size 64 and 32 have been used, kernel sizes of 3x3 and stride = 2. The activation function LeakyReLU has been used with a slope coefficient $\beta = 0.2$. Then, a Batch Normalization layer has been used to maintain the mean output close to 0 and the output standard deviation close to 1. The Flatten layer is used to reshape the output into a vector to feed the Dense layer with a number of neurons $l = 16$ which corresponds to the dimension of the latent space. The dimension l was determined by analyzing the learning curves in the training procedure.

In the decoder part, a Dense layer is used again to recover the previous size of the feature vector while the reshaping recovers the initial 2D input structure. Then, Conv2DTranspose layers have been used to perform the reconstruction of the input structure, having an identical configuration than in the encoder side but changing the order of the filters (32 and 64). The LeakyReLU activations and the Batch Normalization are identical. The last layer is composed by 1 filter, kernel size of 3x3 and stride = 1. This last layer is for recovering completely the size as the input structure. Furthermore, the DAE network trains itself to augment the resolution of the input image, because it will remove artifacts resulting from a lower resolution, by learning from a high resolution target. Then, this reconstructed image will be used for our anomaly detection algorithm. This is advantageous for our problem, leading to a strategy for improving the performance of the system.

6.5 Performance metrics

To evaluate the prediction effectiveness of our proposed method, we resort on common performance metrics that are widely used in the related literature. Concretely, we are focusing on the F1-Score which is a metric based on the Precision and Recall metrics. First, we need to

---

[6] Note that the summation is made along all the pixel values. However, for the sake of clarity, we drop the subindex in this expression.

describe what we consider as a positive or negative event. In our problem, TP and FP stand for True and False Positive (anomalous event) while TN and FN for True and False Negative (correct event). In this way, the applied metrics are defined as follows:

- Precision positive (PP) and negative (PN) as the proportion of correct predictions of a given class

$$\text{PP} = \frac{\text{TP}}{\text{TP+FP}}, \qquad \text{PN} = \frac{\text{TN}}{\text{TN+FN}}. \tag{B.26}$$

- Recall positive (RP) and negative (RN) as the proportion of actual occurrences of a given class which has been correctly predicted.

$$\text{RP} = \frac{\text{TP}}{\text{TP+FN}}, \qquad \text{RN} = \frac{\text{TN}}{\text{TN+FP}}. \tag{B.27}$$

- Positive F1-Score (PF_1) and Negative F1-Score (NF_1) as the harmonic mean of precision and recall:

$$PF_1 = 2 \cdot \frac{\text{PP} \cdot \text{RP}}{\text{PP} + \text{RP}}, \qquad NF_1 = 2 \cdot \frac{\text{PN} \cdot \text{RN}}{\text{PN} + \text{RN}}. \tag{B.28}$$

Note that although the training procedure is fully unsupervised, for our specific evaluation we know the labels of the data samples, meaning we can calculate these metrics, well-known in the supervised learning literature.

7 Numerical results and Discussion

We here present some numerical results in order to analyze the performance of both proposals (statistical test and radio image sensing) in our evaluation setup described in Section 6. Generally, in the considered industrial setup, it would be more desirable to avoid undetected anomalies (which may indicate some error in the robot or some external issue in the predefined trajectory) than obtaining a false positive. Hence, all the figures in this section shows the algorithm performance in terms of the PF_1 metric.

Also, we mainly focus our results on the radio image sensing algorithm since it is the proposal with a larger number of tunable parameters, whilst the statistical hypothesis testing is used as a benchmark of the ML based solution.

7.1 Impact of sampling and noise averaging

First, we evaluate the impact of both available number of samples and the noise averaging technique in the radio image sensing algorithm. To that end, we consider a LIS compounded by $M = 32 \times 32$ antennas and a spacing $\Delta s = \lambda/2$ for a $\Delta d = 10$ cm parallel deviation.

We evaluate two approaches: *i)* using the S extra samples directly as input to the algorithm, being $Tc = N_c \times N_s \times S$, and *ii)* using the S extra samples for averaging. For this particular case, $N'_s \in \{1000, 100\}$. Then $\forall\ \mathbf{p}_j$ we use $S = \frac{N'_s}{N_s}$ samples for obtaining N_s S-averaged samples for training the algorithm, being still $Tc = N_c \times N_s = 1850$. Note that the number of

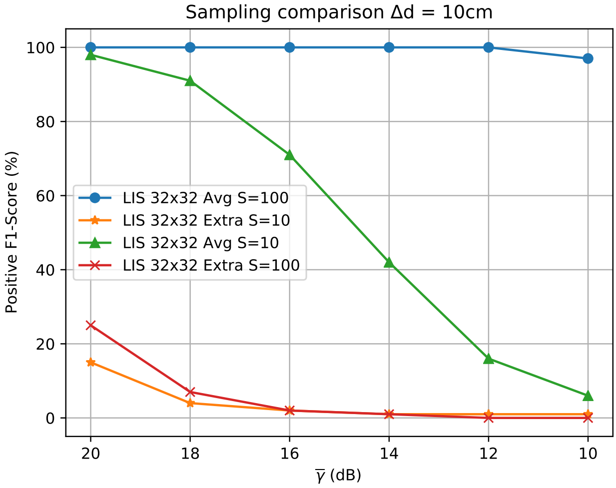


Fig. B.6: PF_1 score for radio image sensing with $M = 32 \times 32$ antennas, inter-element distance of $\Delta s = \lambda/2$, correct route parallel to the LIS, anomalous points placed at $\Delta d = 10$ cm, and different numbers of samples.

samples N'_s would depend on the sampling frequency and the second order characterization of the channel, i.e., the channel coherence time and its autocorrelation function.

Figure B.6 shows the performance of the system when using S extra samples and S averaged ones respectively. As highlighted previously, noise contribution is critical in image classification performance, leading to not achieving a valuable improvement when augmenting the number of samples presented to the algorithm. However, when performing the averaging, results are significantly improved due to the noise variance reduction. As expected, when noise level is higher, more samples are needed to preserve the pattern by averaging, being $N'_s = 1000$ the one which yields a better performance. For the following discussions, this sampling strategy will be used, meaning we are using $S = 100$ extra samples.

7.2 Impact of antenna spacing

The next step is evaluating the impact of inter-antenna distance in the ML sensing solution. We fix the aperture to 5.44×5.44 m and $S = 100$ averaged samples. Then, we assess the performance in both $\Delta d = 50/10$ cm for the parallel deviation, and we analyze different spacings with respect to the wavelength ($\lambda/2$, λ and 2λ).

The performance results for the distinct configurations are depicted in Fig. B.7. As observed, the spacing of 2λ — which is far from the concept of LIS — is presenting really inaccurate results showing that the spatial resolution is not enough. We can conclude that the quick variations along the surface provide important information to the classifier performance.

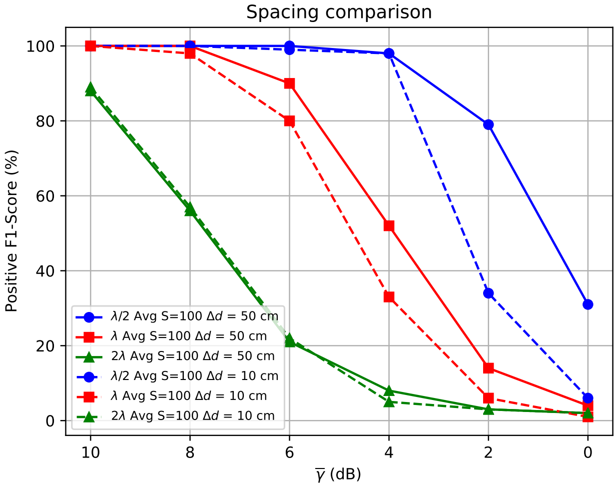


Fig. B.7: PF_1 score for radio image sensing with fixed LIS aperture of 5.44×5.44 m, correct route parallel to the LIS, anomalous points placed at $\Delta d = 10$ and $\Delta d = 50$ cm, $S = 100$ samples, and different number of antennas and spacings.

Besides, this information becomes more important the lower the distance between the routes is. Specially in the range of $\overline{\gamma} \in [10, 4]$ for $\lambda/2$ where the detection is almost identical regardless of the extra precision needed to detect the deviation when the routes are closer. Furthermore, the effect of antenna densification for a given aperture is highlighted and it can be seen that the lowest spacing leads to the best results.

7.3 LIS aperture comparisons

In this case, LIS with different apertures have been evaluated. The spacing is fixed to $\lambda/2$, $S = 100$ averaged samples are used while the deviation is $\Delta d = 10$ cm parallel.

Looking at Fig. B.8, the aperture plays a vital role in the sensing performance. Increasing the number of antennas leads to a higher resolution image, being able to capture the large-scale events occurring in the environment more accurately. Note the usage of incoherent detectors is yielding to a good performance when the aperture is large enough. The key feature for this phenomena is the LIS pattern spatial consistency, i.e., the ability of representing the environment as a continuous measurement image.

7.4 DAE for image Super-resolution evaluation

In this case, the impact on performance by using DAE is evaluated and compared to the hypothesis test in (B.4). We fix the aperture to $M = 32 \times 32$, for a parallel deviation of

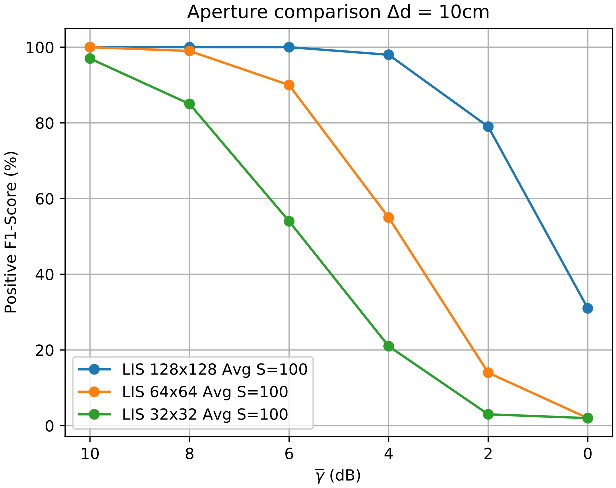


Fig. B.8: PF_1 score for radio image sensing with variable aperture, inter-antenna distance of $\lambda/2$, correct route parallel to the LIS, anomalous points placed at $\Delta d = 10$ and $S = 100$ samples.

$\Delta d = 10$ cm and an antenna spacing of $\lambda/2$.

For this evaluation, the performance is analyzed in 4 cases: *i)* no pre-processing of images performed, *ii)* $S = 100$ averaging strategy, *iii)* image resolution augmentation using DAE, and *iv)* The hypothesis test proposed in Algorithm 1. For the DAE, we assume we have access to a target reference image $\mathbf{t} \in \mathbb{N} \cap [0, 255]^N|_{N=128\times128}$ with $\overline{\gamma} = 10$ dB and our corrupted input is $\mathbf{c} \in \mathbb{N} \cap [0, 255]^M|_{M=32\times32}$ with $\overline{\gamma} \in [10, -10]$ dB. Then, we finally resize both images $R : [0, 255]^d \to \mathbb{R} \cap [0, 1]^R|_{R=224\times224}$, obtaining images of $\mathbf{t_r} \in \mathbb{N} \cap [0, 1]^{224\times224}$ and $\mathbf{c_r} \in \mathbb{N} \cap [0, 1]^{224\times224}$ pixels.

Regarding Fig. B.9, one can see the raw-data (blue line) is yielding to a really poor performance. This is expected taking into account noise can interfere significantly in the local density of the clusters, leading to wrong results. Also, the noise averaging strategy is good enough when noise contribution is negligible, meaning that for improving the results in lower SNR scenarios we would need to obtain a higher S which would be unpractical. Finally, the usage of DAE for image super-resolution outperforms both methods, allowing to improve the system performance and even work in quite unfavourable SNR scenarios. In turn, the hypothesis test derived in Section 4 provides in general the best performance.

However, we must take into account that the statistical test is built based on some key a priori knowledge, namely Gaussian noise with known variance. In the context of estimation, the radio image sensing solution can be seen as a non-parametric approach, which is valid for any baseline distribution and no further assumptions are required. Nevertheless, the performance of the ML solution (when DAE is employed), presents almost no difference with respect to the ad-hoc test up to 2 dB of average SNR. This is a promising result, since the application of more

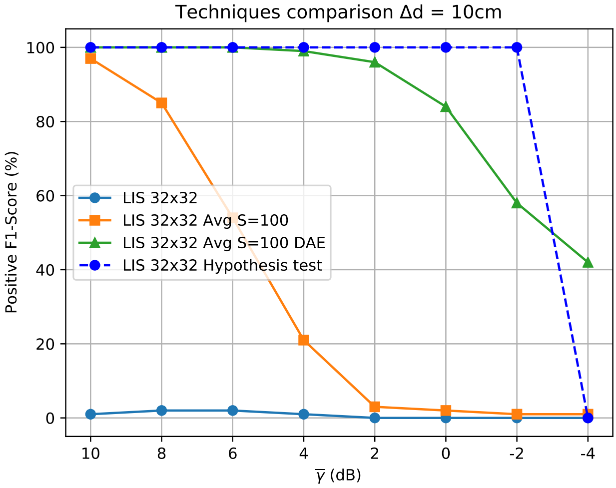


Fig. B.9: Comparison between radio image sensing and the statistical solution in (B.4) for $M = 32 \times 32$ antennas, correct route parallel to the LIS, spacing of $\lambda/2$ and different processing techniques.

refined image processing techniques may lead to an increase in performance. Also, note that here we are considering a rather simple scenario, where the scatterers do not move. In a more realistic environment, with the rapid changes in the channel and the temporal dependencies due to the relative positions between users and scatterers in movement, ML-based sensing seems a promising solution to learn temporal dependencies in those scenarios where classical solutions become impractical.

7.5 Route deviations evaluation

We evaluate now the impact of the separation of deviations and different types of routes in both radio image sensing and the statistical test. To that end, we fix the aperture to $M = 32 \times 32$, and a antenna spacing of $\lambda/2$. We will be using all the improvements in the preprocessing of the images to leverage the performance of the ML system.

In Fig. B.10, we can see the performance of the system under different deviations and SNR conditions. We can see the system works better the closer the deviation of the routes are. This is an advantage of our proposed approach, the closer the routes are, the more accurate the reconstruction of the DAE is, taking into account the corrupted image $\mathbf{c_r}$ will be more similar to the target image $\mathbf{t_r}$, allowing a better augmentation of the image resolution, so the correct clusters can be learned more accurately. In this way, the ML proposed algorithm works better in the cases a standard wireless sensing system would be more prone to failure. Also, the parallel deviations are easier to detect than the normal deviations. The path loss of the points in the parallel routes remains almost identical regardless of the specific point, making it easier

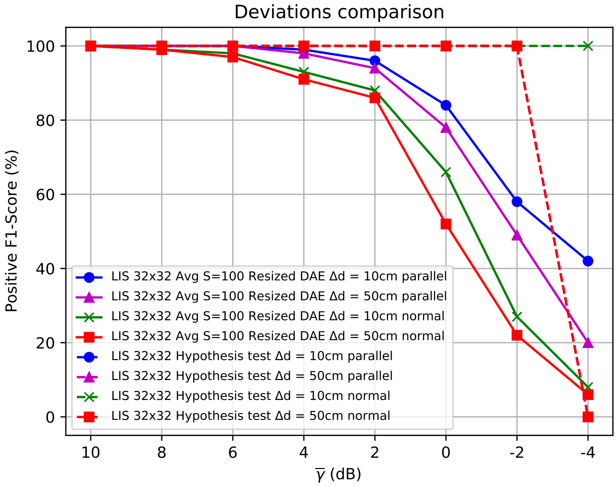


Fig. B.10: Comparison between radio image sensing and the statistical solution in (B.4) for $M = 32 \times 32$ antennas, different correct routes and spacing of $\lambda/2$.

to detect. It is important to highlight the SNR definition presented in (B.19) can influence in the pattern acquisition in the normal deviation cases when points are far from the LIS, which will have a significantly lower instantaneous SNR leading to a more difficult detection.

Note that the abrupt decrease on performance for the hypothesis test is due to the fact that we are using a pointwise test to perform a detection over a whole route (collection of points), as shown in Algorithm 1. Whilst the ML algorithm performs an anomaly detection over the whole correct route, the proposed statistical test is a pointwise comparison, i.e., it checks the validity of the null-hypothesis for each point in the correct route separately. This implies that, in order to detect a point as anomalous, the test has to reject H_0 on all the correct points. Consequently, failing in a single point is equivalent to fail in all the points.

7.6 Performance evaluation under changing environment

Finally, we here evaluate the performance of our system when a major change in the scenario occurred, i.e., the relative positions between the scatterers and the transmitter has changed considerably and thus the pattern capture in the radio image no longer matches the original one used in the training phase. Note that, although the considered scenario for testing was assumed to be fixed, we may be interested in extrapolate the performance of the proposed solutions when dealing with environmental changes. To that end, we evaluate the anomaly detection accuracy of both the hypothesis test and the ML solution. We fix the aperture to $M = 32 \times 32$, for a parallel deviation of $\Delta d = 10$ cm and an antenna spacing of $\lambda/2$. Again, we will use all the improvements in the preprocessing of the images to leverage the performance of

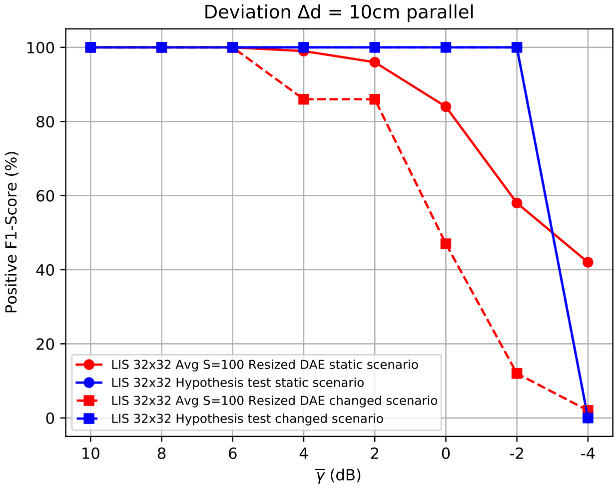


Fig. B.11: Performance of radio image sensing and the statistical solution in (B.4) for $M = 32 \times 32$ antennas, correct route parallel to the LIS, anomalous points placed at $\Delta d = 10$ and spacing of $\lambda/2$ in a changing environment.

the ML system. Fig. B.11 shows the performance of the system under a changing environment. The hypothesis test is robust to an environmental change, as its performance remains similar as the static case. With respect to the ML solution, in the range of $\overline{\gamma} \in [0, -4]$ drops significantly. However, in the proposed scenario, we assume a change in the environment is a really unlikely event, leading to a worsening in the performance in some SNR cases.

8 Conclusions

We have made the first step towards the use of LIS for sensing the propagation environment, exploring and proposing two different solutions: *i)* an statistical hypothesis test based on a generalization of the likelihood ratio, and *ii)* a ML based algorithm, which exploits the high density of antennas in the LIS to obtain radio-images of the scenario. We provide a complete characterization of the statistical solution, and also pave the way to the use of ML technique to improve the performance in the second case. As an example, we have shown that the use of denoising autoencoders considerably boosts the performance of the ML algorithm. Both proposals are tested in an exemplary industrial scenario, showing that, up to relatively low values of SNR, the performance of the two presented techniques is rather similar. The ML solution implies a larger computational effort than the statistical test, but in turn does not require any a priori knowledge, as is the case of the test in which the variance of the noise is assumed in order to reduce analytical complexity. Finally, the results obtained in this system motivate a further study with more complex detectors of I/Q components to quantify the potential performance gain obtained from using I/Q receivers, i.e., analyzing the trade-off between the system complexity and its performance.

9 Acknowledgements

This work has received funding from the European Union's Horizon 2020 research and innovation programme under the Marie Sklodowska-Curie grant agreement No 813999 WindMill project. Also, this work has been partially supported by EU H2020 RISE-6G project and the Danish Council for Independent Research DFF-701700271.

References

[1] C. J. Vaca-Rubio, P. Ramirez-Espinosa, R. J. Williams, K. Kansanen, Z.-H. Tan, E. de Carvalho, and P. Popovski, "A Primer on Large Intelligent Surface (LIS) for Wireless Sensing in an Industrial Setting," *arXiv preprint arXiv:2006.06563*, 2020.

[2] J. G. Andrews, S. Buzzi, W. Choi, S. V. Hanly, A. Lozano, A. C. Soong, and J. Zhang, "What will 5G be?," *IEEE J. Sel. Areas Commun.*, vol. 32, no. 6, pp. 1065–1082, 2014.

[3] E. G. Larsson et al., "Massive MIMO for next generation wireless systems," *IEEE Commun. Mag.*, vol. 52, no. 2, pp. 186-195, 2014.

[4] N. Shlezinger, G. C. Alexandropoulos, M. F. Imani, Y. C. Eldar, and D. R. Smith, "Dynamic metasurface antennas for 6G extreme massive MIMO communications," *IEEE Wireless Communications*, 2021.

[5] C. Huang, S. Hu, G. C. Alexandropoulos, A. Zappone, C. Yuen, R. Zhang, M. Di Renzo, and M. Debbah, "Holographic MIMO surfaces for 6G wireless networks: Opportunities, challenges, and trends," *IEEE Wireless Communications*, vol. 27, no. 5, pp. 118–125, 2020.

[6] S. Hu et al., "Beyond Massive MIMO: The Potential of Data Transmission With Large Intelligent Surfaces," *IEEE Trans. Signal Process.*, vol. 66, no. 10, pp. 2746-2758, 2018.

[7] E. Basar, M. Di Renzo, J. de Rosny, M. Debbah, M.-S. Alouini, and R. Zhang, "Wireless communications through reconfigurable intelligent surfaces," *IEEE Access*, vol. 7, pp. 116753–116773, 2019.

[8] E. Björnson et al., "Massive MIMO is a reality - What is next?: Five promising research directions for antenna arrays," *Digit. Signal Process.*, vol. 94, pp. 3-20, 2019, `https://doi.org/10.1016/j.dsp.2019.06.007`.

[9] M. Di Renzo, A. Zappone, M. Debbah, M.-S. Alouini, C. Yuen, J. de Rosny, S. Tretyakov, and G. C. Alexandropoulos, "Smart radio environments empowered by reconfigurable intelligent surfaces: How it works, state of research, and the road ahead," *IEEE Journal on Selected Areas in Communications*, vol. 38, no. 11, pp. 2450–2525, 2020.

[10] G. C. Alexandropoulos, G. Lerosey, M. Debbah, and M. Fink, "Reconfigurable intelligent surfaces and metamaterials: The potential of wave propagation control for 6G wireless communications," *arXiv preprint arXiv:2006.11136*, 2020.

[11] M. Di Renzo, M. Debbah, D.-T. Phan-Huy, A. Zappone, M.-S. Alouini, C. Yuen, V. Sciancalepore, G. C. Alexandropoulos, J. Hoydis, H. Gacanin, and others, "Smart radio environments empowered by reconfigurable AI meta-surfaces: An idea whose time has come," *EURASIP Journal on Wireless Communications and Networking*, vol. 2019, no. 1, pp. 1–20, 2019.

[12] Q. Wu and R. Zhang, "Intelligent reflecting surface enhanced wireless network via joint active and passive beamforming," *IEEE Transactions on Wireless Communications*, vol. 18, no. 11, pp. 5394–5409, 2019.

[13] H. Huang et al., "Deep-Learning-Based Millimeter-Wave Massive MIMO for Hybrid Precoding," *IEEE Trans. Veh. Technol.*, vol. 68, no. 3, pp. 3027-3032, 2019.

[14] M.-A. Badiu and J. P. Coon, "Communication through a large reflecting surface with phase errors," *IEEE Wireless Commun. Lett*, 2019.

[15] H. Wymeersch, J. He, B. Denis, A. Clemente, and M. Juntti, "Radio localization and mapping with reconfigurable intelligent surfaces," *arXiv preprint arXiv:1912.09401*, 2019.

[16] R. Alghamdi et al., "Intelligent Surfaces for 6G Wireless Networks: A Survey of Optimization and Performance Analysis Techniques," *arXiv preprint:2006.06541 [eess.SP]*, 2020.

[17] J. He, H. Wymeersch, L. Kong, O. Silvén, and M. Juntti, "Large intelligent surface for positioning in millimeter wave MIMO systems," in *2020 IEEE 91st Vehicular Technology Conference (VTC2020-Spring)*, 2020, pp. 1–5.

[18] H. Shen et al., "Secrecy Rate Maximization for Intelligent Reflecting Surface Assisted Multi-Antenna Communications," *IEEE Commun. Lett.*, vol. 23, no. 9, pp. 1488-1492, 2019.

[19] J. D. V. Sánchez et al., "Physical Layer Security of Large Reflecting Surface Aided Communications with Phase Errors," *IEEE Wireless Commun. Lett.*, vol. , no. , pp. 1-1, 2020, `https://doi.org/10.1109/LWC.2020.3029816`.

[20] C. Huang, G. C. Alexandropoulos, C. Yuen, and M. Debbah, "Indoor signal focusing with deep learning designed reconfigurable intelligent surfaces," in *2019 IEEE 20th International Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*, 2019, pp. 1–5.

[21] D. Dardari, "Communicating With Large Intelligent Surfaces: Fundamental Limits and Models," *IEEE J. Sel. Areas Commun.*, vol. 38, no. 11, pp. 2526-2537, 2020, `https://doi.org/10.1109/JSAC.2020.3007036`.

[22] S. Hu et al., "Beyond Massive MIMO: The Potential of Positioning With Large Intelligent Surfaces," *IEEE Trans. Signal Process.*, vol. 66, no. 7, pp. 1761-1774, 2018.

[23] S. Hu, F. Rusek, and O. Edfors, "Cramér-Rao lower bounds for positioning with large intelligent surfaces," in *2017 IEEE 86th Vehicular Technology Conference (VTC-Fall)*, 2017, pp. 1–6.

[24] H. Wang et al., "RT-Fall: A real-time and contactless fall detection system with commodity WiFi devices," *IEEE Trans. Mobile Comput.*, vol. 16, no. 2, pp. 511-526, 2016.

[25] Q. Pu, S. Gupta, S. Gollakota, and S. Patel, "Whole-home gesture recognition using wireless signals," in *Proc. 19th Annual Inter. Conf. Mobile Comput. & Netw.*, 2013, pp. 27–38.

[26] Y. Zhao, N. Patwari, J. M. Phillips, and S. Venkatasubramanian, "Radio tomographic imaging and tracking of stationary and moving people via kernel distance," in *2013 ACM/IEEE Inter. Conf. Inf. Process. Sensor Networks (IPSN)*, 2013, pp. 229–240.

[27] F. Adib et al., "Real-time breath monitoring using wireless signals," in *Proc. 20th Annual Inter. Conf. Mobile Comput. Netw.*, pp. 261-262, 2014.

[28] M. Zhao et al., "Through-wall human pose estimation using radio signals," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, pp. 7356-7365, 2018.

[29] J. Wilson and N. Patwari, "Radio tomographic imaging with wireless networks," *IEEE Trans. Mobile Comput.*, vol. 9, no. 5, pp. 621-632, 2010.

[30] S. Bozkurt, G. Elibol, S. Gunal, and U. Yayan, "A comparative study on machine learning algorithms for indoor positioning," in *2015 International Symposium on Innovations in Intelligent SysTems and Applications (INISTA)*, 2015, pp. 1–8.

[31] J. Joung, "Machine learning-based antenna selection in wireless communications," *IEEE Commun. Lett.*, vol. 20, no. 11, pp. 2241-2244, 2016.

[32] O. T. Demir and E. Bjornson, "Channel Estimation in Massive MIMO Under Hardware Non-Linearities: Bayesian Methods Versus Deep Learning," *IEEE O. J. Commun. Soc.*, vol. 1, no. , pp. 109-124.

[33] X. Ma and Z. Gao, "Data-Driven Deep Learning to Design Pilot and Channel Estimator for Massive MIMO," *IEEE Trans. Veh. Technol.*, vol. 69, no. 5, pp. 5677-5682, 2020.

[34] H. Huang et al., "Deep Learning for Super-Resolution Channel Estimation and DOA Estimation Based Massive MIMO System," *IEEE Trans. Veh. Technol.*, vol. 67, no. 9, pp. 8549-8560, 2018.

[35] C. Wen et al., "Deep Learning for Massive MIMO CSI Feedback," *IEEE Wireless Commun. Lett.*, vol. 7, no. 5, pp. 748-751, 2018.

[36] C. Luo et al., "Channel State Information Prediction for 5G Wireless Communications: A Deep Learning Approach," *IEEE Trans. Netw. Sci. Eng.*, vol. 7, no. 1, pp. 227-236, 2020.

[37] G. D. Durgin et al., "New analytical models and probability density functions for fading in wireless communications," *IEEE Trans. Commun.*, vol. 50, no. 6, pp. 1005-1015, 2002.

[38] J. Sijbers et al., "Maximum-likelihood estimation of Rician distribution parameters," *IEEE Trans. Med. Imag.*, vol. 17, no. 3, pp. 357-361, 1998, `https://doi.org/10.1109/42.712125`.

[39] S. M. Kay, *Fundamentals of Statistical Signal Processing: Detection theory*, Prentice-Hall PTR, 1998.

[40] W. F. Scott, "On the Asymptotic Distribution of the Likelihood Ratio Statistic," *Commun. Stat. Theory Methods*, vol. 36, no. 2, pp. 273-281, 2007, `https://doi.org/10.1080/03610920600974229`.

[41] J. Idier and G. Collewet, “Properties of Fisher information for Rician distributions and consequences in MRI,” `https://hal.archives-ouvertes.fr/hal-01072813`, Oct. 2014.

[42] S. B. Provost and A. M. Mathai, *Quadratic Forms in Random Variables: Theory and Applications*, Marcel Dekker, 1992.

[43] P. Ramírez-Espinosa et al., “New Approximation to Distribution of Positive RVs Applied to Gaussian Quadratic Forms,” *IEEE Signal Process. Lett.*, vol. 26, no. 6, pp. 923-927, Jun. 2019, `https://doi.org/10.1109/LSP.2019.2912295`.

[44] T. Y. Al-Naffouri et al., “On the Distribution of Indefinite Quadratic Forms in Gaussian Random Variables,” *IEEE Trans. Commun.*, vol. 64, no. 1, pp. 153-165, 2016, `https://doi.org/10.1109/TCOMM.2015.2496592`.

[45] S. J. Pan and Q. Yang, “A survey on transfer learning,” *IEEE Trans. Knowl. Data Eng.*, vol. 22, no. 10, pp. 1345-1359, 2009.

[46] K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” *arXiv preprint arXiv:1409.1556*, 2014.

[47] M. M. Breunig, H.-P. Kriegel, R. T. Ng, and J. Sander, "LOF: identifying density-based local outliers," in *Proceedings of the 2000 ACM SIGMOD international conference on Management of data*, 2000, pp. 93–104.

[48] Winprop, Altair Engineering, Inc, `https://www.altairhyperworks.com/winprop`.

[49] P. Roy, S. Ghosh, S. Bhattacharya, and U. Pal, "Effects of degradations on deep neural network architectures," *arXiv preprint arXiv:1807.10108*, 2018.

[50] I. Goodfellow, Y. Bengio, A. Courville, and Y. Bengio, *Deep learning*, vol. 1. MIT press Cambridge, 2016.

[51] P. Vincent, H. Larochelle, I. Lajoie, Y. Bengio, P.-A. Manzagol, and L. Bottou, "Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion," *J. Mach. Learn. Res.*, vol. 11, no. 12, 2010.

[52] F. Chollet, “Keras,” 2015, `https://keras.io`.

Paper C

Radio sensing with large intelligent surface for 6G

Cristian J Vaca-Rubio, Pablo Ramirez-Espinosa, Kimmo Kansanen,
Zheng-Hua Tan, Elisabeth de Carvalho

The paper has been published in the
IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP),
pp. 1-5, 2023.

© 2023 IEEE
The layout has been revised.

Abstract

This paper leverages the potential of Large Intelligent Surfaces (LIS) for radio sensing in 6G wireless networks. By taking advantage of arbitrary communication signals occurring in the scenario, we apply direct processing to the output signal from the LIS to obtain a radio map that describes the physical presence of passive devices (scatterers, humans) which act as virtual sources due to the communication signal reflections. We then assess the usage of machine learning and computer vision methods including clustering, template matching and component labeling to extract meaningful information from these radio maps. As an exemplary use case, we evaluate this method for passive multi-human detection in an indoor setting. The results show that the presented method has high application potential as we are able to detect around 98% of humans passively even in quite unfavorable Signal-to-Noise Ratio (SNR) conditions.

1 Introduction

Sensing can be regarded as the ability of wireless systems to process the signals with the aim of describing the physical environment. There are different methodologies to perform sensing using wireless signals. Essentially, some of these methods use dedicated signals and/or specific hardware [1–8], while others use communication signals of commodity devices to perform the sensing task [9–15]. As an example of the first type, in [3–6] they employ RTI, which is a RSS-based technology for rendering physical objects in wireless networks. They create a radio map based on the RSS variations due to objects presence in the scenario by deploying nodes around the room conforming a Wireless Sensor Network (WSN). In turn, by making use of the communication signals occurring in an environment and avoiding dedicated transmissions [9, 10], one can rely on properties of the wireless channel such as the CSI using commodity Wi-Fi devices, to perform sensing tasks as human gesture recognition or fall detection. Works like the ones presented in [1, 2, 7, 8] capture the reflections of wireless signals, similar to the radar principle.

In the context of communications, the MIMO technique is a fundamental technology in 5G with the main purpose of increasing area spectral efficiency [16, 17]. Intending to push their benefits to the limit and look towards post-5G, researchers are defining a new generation of base stations that are equipped with an even larger number of antennas. The concept of LIS gained a lot of attraction. It designates a large continuous electromagnetic surface able to transmit and receive radio waves. While the potential for communications of LIS is being investigated, these devices offer possibilities that are not being understudied accurately, i.e., environmental sensing based on radio images [18].

Due to the increasing interest in both sensing and LIS, and motivated by their future integration in communication systems, in this work, we are focusing on LIS sensing capabilities. We make use of a method that enables reconstructing a radio map of the propagation environment using an indoor LIS deployment in the ceiling [19, 20]. This radio map shows the presence of active and passive (scatterers/humans) users in the environment by piggybacking the communication signals. We solve a problem of passive multi-human detection in the scenario using the reconstructed radio maps. Detecting passive humans is of great interest as we are relying on environmental radio signals and do not need dedicated devices. This could be quite to optimize beamforming towards the passive human enabling the access phase with an optimized radiation pattern, for Electromagnetic (EM) avoidance and Physical Layer Security (PLS), where the detection of the passive target is mandatory to perform beamforming. The solution is based on the k-means clustering of the radio maps, followed by the application of image processing to enhance the quality and computer vision to perform the detection. We measure the detection accuracy as the number of users detected while also verifying the positioning accuracy.

2 Problem Formulation and System Description

Let us consider an indoor scenario where U users are randomly deployed in a room. Within the U users, a subset U_a are commodity wireless devices fulfilling their communication tasks, while $U_p = U - U_a$ users are just passive human beings. The objective is, hence, sensing the position of both the U_a active and the U_p passive humans from the signals radiated by the former. For the sake of simplicity, we assume the U_a users transmit at the same frequency — representing, e.g., Wi-Fi signaling or transmissions at some cellular frequency band. To perform the sensing, we assume that an LIS of M antenna elements is placed along the ceiling, whose physical aperture comprises its whole area. The sensing problem reduces to determine, from the superposition of the received signals from each of the U_a users at every of the M LIS elements, the (x, y) coordinates of the U_p passive humans. The superposed complex baseband signal received at the LIS is given by

$$\mathbf{y} = \sum_{u=1}^{U_a} \mathbf{h_u} x_u + \mathbf{n}, \tag{C.1}$$

with x_u the transmitted (sensing) symbol from user u, $\mathbf{h_u} \in \mathbb{C}^{M\times 1}$ the channel vector from a specific position of user u to each antenna-element, and $\mathbf{n} \sim \mathcal{CN}_M(\mathbf{0}, \sigma^2\mathbf{I}_M)$ the noise vector. Please note we are considering a narrowband transmission, avoiding frequency selectivity effects.

3 LIS radio map generation

Due to the large physical aperture of the deployment in comparison with the distance between the transmitters and the LIS, spherical wave propagation needs to be taken into account, and thus the channel coefficient $h_{s,i}$ at the LIS i-th element from an arbitrary user transmission is proportional to [21]

$$h_{s,i} \; \propto \; \frac{1}{d_i} e^{-j\frac{2\pi}{\lambda}d_i}, \tag{C.2}$$

where $d_i = \sqrt{(x_i - x_u)^2 + (y_i - y_u)^2 + (z_i - z_u)^2}$ denotes the distance between the active device u and the i-th antenna, and λ is the wavelength. We are interested in determining the spherical steering vector by using (2). For that we define an array of $N \times N$ m aperture with an antenna spacing of $\Delta s = \frac{\lambda}{2}$ resulting in $N_f = \frac{N}{\Delta s} \times \frac{N}{\Delta s}$ antennas and we set a f. We then

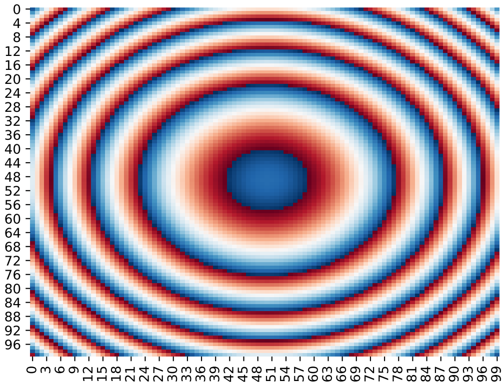


Fig. C.1: Phase representation of the designed filter based on (C.2).

emulate a transmitter in the center position of the filter $(x_u, y_u, z_u) = (\frac{N}{2}, \frac{N}{2}, 6.2)$ m[1]. Next, we compute eq (2) with respect to all the antenna elements from the designed array, obtaining $\mathbf{h_s}$. Figure C.1 shows the expected spherical pattern $\mathbf{h_s}$. We are not interested in the absolute phase values but in their variation along the space. In this way, describing the surface in a vectorized notation, we can derive a Matched Filter (MF) such that:

$$\mathbf{y_f} = \mathbf{h_s} * \mathbf{y}, \tag{C.3}$$

where $*$ denotes the convolution operator. This convolution is performed along all the LIS dimension. Then, $\mathbf{h_s} \in \mathbb{C}^{N_f \times 1}$ denotes the expected spherical pattern (steering vector) for N_f antennas LIS deployment on (C.2), $\mathbf{y}$ the received signal from (C.1) and $\mathbf{y_f} \in \mathbb{C}^{M \times 1}$ the filtered output that represents the radio map. To guarantee the same output dimension (due to the 2D convolution along the LIS), we zero-pad $\mathbf{y}$ such that the output $\mathbf{y_f} \in \mathbb{C}^{M \times 1}$. To obtain a radio map, we just need to compute the energy at the output of the MF procedure $|\mathbf{y}_f| \in \mathbb{R}^{M \times 1}$. We then map the values to the RGB scale using the function $F : \mathbb{R}^{M \times 1} \rightarrow \{[0, 255] \cap \mathbb{N}\}^{M \times 3}$ such that $\mathbf{y}_m = F(|\mathbf{y}_f|)$. Fig. C.2 shows an exemplary radio map. In the exemplary scenario, one active transmitter $U_a = 1$ is used, while three static scatterers are present in the environment. We see the three scatterers in the environment (the cylindric-like shapes) while we can also identify the highest peak representing the user transmission. The scatterers are captured because from the receiver LIS viewpoint, they act as virtual sources that are equivalent to LoS components, i.e., in (C.2) the different reflections are equivalent to a LoS path.

---

[1] The distance $z_u = 6.2$ is a parameter for the filter design. This does not imply that in the evaluation, all the transmitters or scatters are fixed at this distance. In our work, we set $f = 3.5$ GHz and $N = 4$ m.

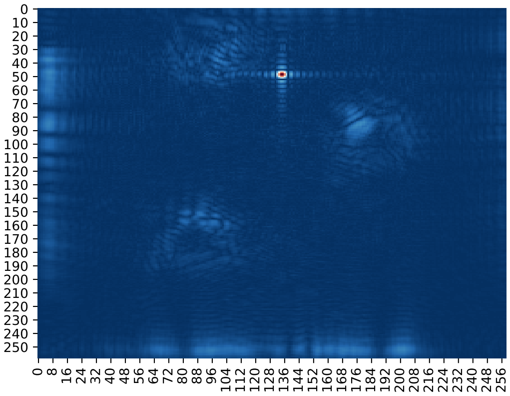


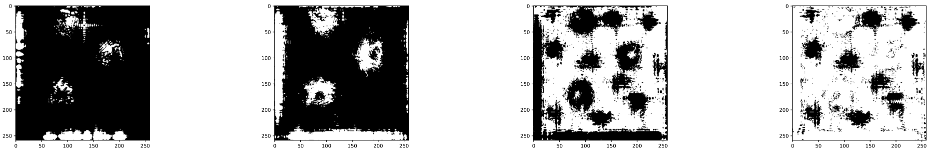
Fig. C.2: Exemplary radio map obtained in a noiseless scenario with $U_a = 1$ users by using the MF design represented in Figure C.1.

4 Passive multi-human detection based on LIS radio map

4.1 Offline scanning phase

We first take advantage of an offline scanning period phase in which we measure different transmissions of any U_a active devices to scan the static features of the propagation environment. We then obtain U_a measurements of the environment for different random active user positions when no passive humans are in the scenario. Figure C.2 shows that we have mainly two dominant ranges of pixel values, either low energy at the output of the MF (the background) or high energy (the active transmitter and scatterers). This leads us to apply a k-means clustering w.r.t. the pixel values of the radio map (with $k = 2$) to enhance the radio map through its binarization. We then define the clusterization as $K : \{[0, 255] \cap \mathbb{N}\}^{M\times 3} \rightarrow \{[0, 255] \cap \mathbb{N}\}^{M\times 1}$ such that $\mathbf{y}_c = K(\mathbf{y}_m)$.

Figure C.3a shows a clusterized version of the radio map presented in Figure C.2. It shows the enhanced areas of the static features of the environment as well as the active transmitter. We use a computer vision technique called Template Matching [22], which detects parts in an image that matches a template image, to remove the expected active transmitter pattern from the clusterized map $\mathbf{y}_c$. By combining different active transmissions along the scenario, we can combine several radio maps to obtain an enhanced version that highlights the scatterers presence in the scenario, as shown in Figure C.3b. These multiple transmission positions illuminate the scatterers from different angles. Furthermore, these map pixel values are either 0 (black) or 1 (white), being white the representation of the scatterers. We will denote this processed map as positive masking map, $\mathbf{y}^{+}_{TM}$.



(a) Exemplary clusterized k-means map obtained by using the MF map in Figure C.2.

(b) Exemplary clusterized k-means positive masking map obtained by the combination of $U_a = 10$ random transmissions.

(c) Exemplary clusterized k-means negative masking map obtained by the combination of $U_a = 10$ random transmissions.

(d) Exemplary logical OR processed negative masking map obtained by the combination of $U_a = 10$ random transmissions.

Fig. C.3: Radio map processing

By having this representation of the static elements of the environment, we can now store $\mathbf{y}_{TM}^{+}$ locally at the LIS to process new maps and remove the static elements of it when trying to detect humans passively.

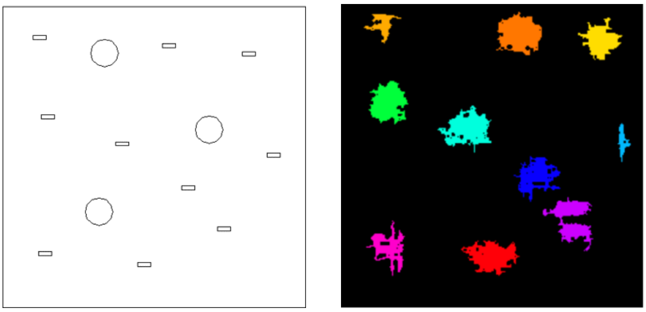
4.2 Detection phase

For this purpose, when there are passive humans in the room, we can follow the same procedure as before but obtaining a negative masking map $\mathbf{y}_{TM}^{-}$ (meaning scatterers are now black) for every temporal radio map snapshot $s \in S$. Figure C.3c shows an example of a negative masking map $\mathbf{y}_{TM}^{-}$ when there is $U_p = 10$ passive humans in the scenario. We see now the scatterers and the humans are represented in black (0 value). Next, We will use it to perform a logical OR operation (+) with the locally stored masking map $\mathbf{y}_{TM}^{+}$. Formally, we denote this operation as $\mathbf{y}_{OR} = \mathbf{y}_{TM}^{+} + \mathbf{y}_{TM}^{-}$. Furthermore, we obtain the OR map $\mathbf{y}_{OR}$, shown in Figure C.3d, which eliminates the static scatterers of the scenario and highlights the passive humans reflections. We can see in the map that there are some artifacts (salt-pepper noise) as a result of this process. To alleviate it, we define a sliding window algorithm of size $K_c \times K_c$ that set all the pixel values comprising the window size to 1 (white) if the number of black pixels in that window is lower than a defined threshold T_h. In this way, we can reduce significantly this salt-pepper noise. Figure C.4b shows the removal of the artifacts thanks to this procedure. Finally, we are interested in detecting these shapes associated to the passive human positions in the radio maps. For that, we adopt a computer vision algorithm named Component Labeling [23] which compares neighboring pixels to detect a shape that is assigned to the same label. Figure C.4a shows the exemplary groundtruth scenario in which these maps are computed while Figure C.4b shows the result of detecting the $U_p = 10$ passive humans. They are assigned to different colors (labels) for illustration purposes. Hence, we can infer the passive human positions by obtaining the center pixel coordinates of these shapes $c_p = (x_p, y_p)$. To infer the real position, we just compute $c = c_p \times \Delta s$, where Δs denotes the antenna spacing.

Algorithm 1 further summarizes the procedure.

Algorithm 2: Passive multi-human localization

```
Offline Scanning Phase:
I. Measure the $U_a$ superposed complex baseband signal at the LIS, $\mathbf{y}$, as shown in (1)
II. $K$-means clustering with $K = 2$ is applied to the processed map $\mathbf{y}_m$ such that we obtain $\mathbf{y}_c$
III. Obtaining $\mathbf{y}^{+}_{TM}$ through Template Matching to eliminate the $U_a$ active transmitters
IV. Store locally $\mathbf{y}^{+}_{TM}$ at the LIS
Detection phase: Passive multi-human detection
for each $s \in S$ do
    I. Follow same procedure I-II from Offline Scanning Phase
    II. Obtaining $\mathbf{y}^{-}_{TM}$ through Template Matching to eliminate the $U_a$ active transmitters
    III. Computing the OR map $\mathbf{y}_{OR}$
    IV. Filtering salt-pepper noise with sliding window $K_c \times K_c$ and threshold $T_h$
    V. Applying Component labelling to detect the shapes of the $U_p$ passive humans
    VI. Compute $c$ to infer the locations
end
```



(a) Exemplary groundtruth scenario with $U_p = 10$ represented as rectangles.

(b) Component Labeling applied to Figure C.3d.

Fig. C.4: Groundtruth position of the U_p humans vs the Component Labeling result.

5 Simulation, numerical results and discussion

5.1 Simulated scenario

We conducted simulations via ray tracing [24] to simulate the multipath in a reliable way. We simulate a scenario of size 10.34 × 10.34 × 8 m. We deploy an LIS with 259 × 259 elements separated $\lambda/2$. Each U_a active device transmits an arbitrary narrowband signal of 20 dBm at 3.5 GHz. The distance from which the MF is calibrated is $z_u = 6.2$. The active U_a are assumed to be ≥ 1.8 m height, being this value randomnly selected. The scatterers are modeled as metallic (with conductivity $s = 19444$ S/m, relative permittivity $\epsilon = 1$ and relative permeability $\mu = 20$)[2] cylinders of 1 m diameter and 2 m height. The passive U_p humans are model as rectangles of dimensions 0.3x0.5x1.7 m (average human dimensions obtained from [25]) with $s = 1.44$ S/m, $\epsilon = 38.1$ and $\mu = 1$ [26].

5.2 Received signal and noise modeling

From the ray-tracing simulation, the received signal in (C.1) is obtained as the complex electric field arriving at the i-th antenna element, $\widetilde{E}_i$, which can be regarded as the superposition of each ray path from every $u \in U_a$ user. Then, the complex signal at the output of the i-th element is therefore given by

$$y_i = \sqrt{\frac{\lambda^2 Z_i}{4\pi Z_0}} \widetilde{E}_i + n_i, \tag{C.4}$$

---

[2] These values are provided by the software manual [24].

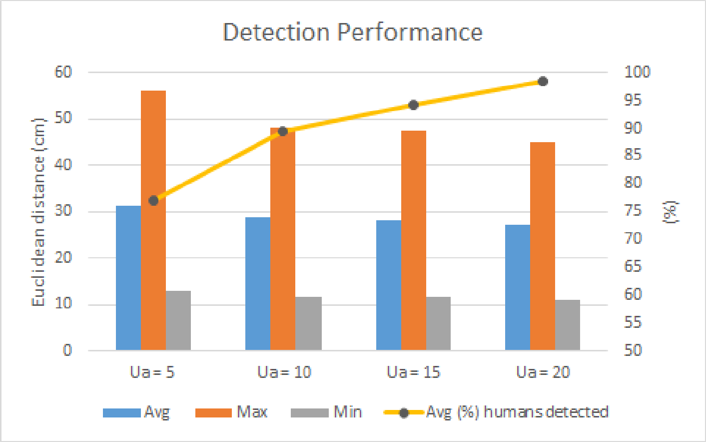


Fig. C.5: Average human detection percentage (%) and positioning errors (cm) with fixed LIS aperture of $M = 259 \times 259$, in a $\gamma = 0$ dB condition, with $S = 100$ averaging strategy and $U_p = 10$ humans in the scenario.

with $Z_0 = 120\pi$ the free space impedance and Z_i the antenna impedance. For simplicity, we consider $Z_i = 1 \, \forall \, i$. We define the average SNR, γ, is defined as

$$\gamma \triangleq \frac{\lambda^2}{4\pi Z_0 M \sigma^2} \sum_{i=1}^{M} |\widetilde{E}_i|^2. \tag{C.5}$$

We assume the system can obtain S extra samples at each channel coherence interval to perform an S-averaging, diminishing the noise variance contribution.

5.3 Passive human detection

We here leverage the performance for passive human detection in the scenario using the method described in Section 4.1. We consider $U_p = 10$ humans at arbitrary positions in the scenario.

The detection of passive humans is highly impacted by the U_a active devices positions. For the sake of generalization, we perform Monte Carlo simulations for obtaining our results under different random configurations. Figure C.5 shows the average, maximum and minimum positioning errors of the correctly detected passive humans as well as the average detected humans by using a different number of active users U_a. Please note, we are not using dedicated active transmissions for this task, but we take advantage of the wireless communications occurring from these active devices in the scenario. The results show that the number of active users does not really impact on the positioning performance as it remains similar when using a lower and a higher number of active users U_a. However, by increasing U_a, the number of passive humans detected increases. This is because the more the transmissions, the more reflections we obtain from the human body reflections leading to an easier detection of the passive humans.

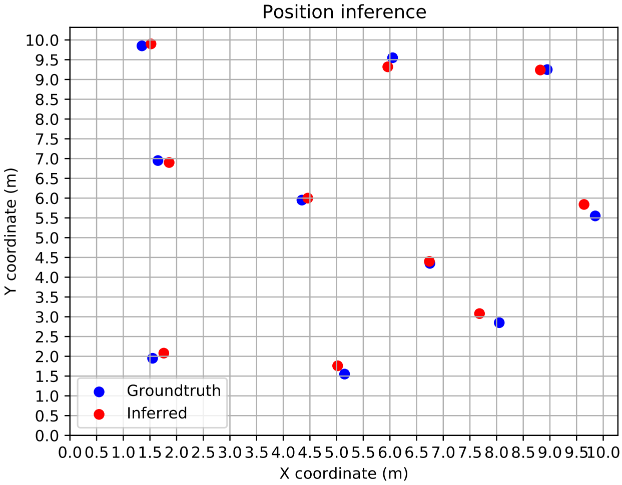


Fig. C.6: Exemplary human detection with fixed LIS aperture of $M = 259 \times 259$, in a $\gamma = 0$ dB condition, with $S = 100$ averaging strategy, $U_a = 20$ active users and $U_p = 10$ humans in the scenario.

Furthermore, the detection of this system is quite accurate, as we can detect a minimum of around 80% humans in all the configurations and the average error is around 28 cm. Figure C.6 shows an illustration of the inferred positions w.r.t. the groundtruth positions. It shows the positioning accuracy is quite high even with 10 people passively sensed.

5.4 Passive human detection distance evaluation

Finally, we here evaluate the accuracy of the detection of passive humans by comparing performance under different separations among them. As we are interested in checking the distance at which the performance may decrease significantly, we set $U_p = 2$ humans separated 25/50/75/100 cm apart, respectively. We test different separations and we evaluate the detection performance of the $U_p = 2$ passive humans. Figure C.7 shows the average detection of the humans. We can see the system achieves around 1.5/2 detections in the most challenging case (25 cm) while obtaining around 1.8/2 in the most favorable (100 cm). This shows the potential of the system, even when the separation among humans is quite small.

6 Conclusions

The presented use case shows machine learning and computer vision algorithms are a powerful tool to take into account when using an image-based LIS sensing approach. Moreover, we note that LIS is one of the technologies being considered for future 6G systems, which may change

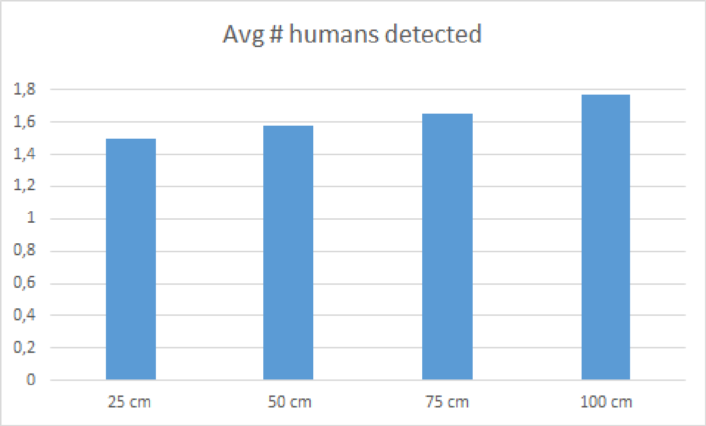


Fig. C.7: Average human detection with fixed LIS aperture of $M = 259 \times 259$, in a $\gamma = 0$ dB condition, with $S = 100$ averaging strategy, $U_a = 20$ active users and $U_p = 2$ humans in the scenario.

the relevant cost/benefit analysis in that any sensing functionality is then expected to be added onto the system rather than requiring explicit investment on extra dedicated hardware.

7 Acknowledgments

This project has received funding from the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie Grant agreement No. 813999.

References

[1] F. Adib, Z. Kabelac, H. Mao, D. Katabi, and R. C. Miller, "Real-time breath monitoring using wireless signals," in *Proceedings of the 20th annual international conference on Mobile computing and networking*, 2014, pp. 261–262.

[2] K. Stasiak and P. Samczynski, "FMCW radar implemented in SDR architecture using a USRP device," in *2017 Signal Processing Symposium (SPSympo)*, 2017, pp. 1–5.

[3] J. Wilson and N. Patwari, "Radio tomographic imaging with wireless networks," *IEEE Transactions on Mobile Computing*, vol. 9, no. 5, pp. 621–632, 2010.

[4] D. Lee and G. B. Giannakis, "A Variational Bayes Approach to Adaptive Radio Tomography," *arXiv preprint arXiv:1909.03892*, 2019.

[5] G. Kłosowski, T. Rymarczyk, P. Adamkiewicz, and M. Styła, "The use of heterogeneous deep neural network system in radio tomography to detect people indoors," in *Proceedings of the 28th Annual International Conference on Mobile Computing And Networking*, 2022, pp. 859–861.

[6] Y. Ma, W. Ning, and B. Wang, "Training-free artifact detection method for radio tomographic imaging based device-free localization," *IEEE Transactions on Vehicular Technology*, vol. 70, no. 10, pp. 10382–10394, 2021.

[7] M. Zhao, T. Li, M. Abu Alsheikh, Y. Tian, H. Zhao, A. Torralba, and D. Katabi, "Through-wall human pose estimation using radio signals," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 7356–7365.

[8] M. Zhao, Y. Tian, H. Zhao, M. A. Alsheikh, T. Li, R. Hristov, Z. Kabelac, D. Katabi, and A. Torralba, "RF-based 3D skeletons," in *Proceedings of the 2018 Conference of the ACM Special Interest Group on Data Communication*, 2018, pp. 267–281.

[9] H. Wang, D. Zhang, Y. Wang, J. Ma, Y. Wang, and S. Li, "RT-Fall: A real-time and contactless fall detection system with commodity WiFi devices," *IEEE Transactions on Mobile Computing*, vol. 16, no. 2, pp. 511–526, 2016.

[10] Q. Pu, S. Gupta, S. Gollakota, and S. Patel, "Whole-home gesture recognition using wireless signals," in *Proceedings of the 19th annual international conference on Mobile computing & networking*, 2013, pp. 27–38.

[11] Y. Zeng, D. Wu, J. Xiong, J. Liu, Z. Liu, and D. Zhang, "MultiSense: Enabling multi-person respiration sensing with commodity WiFi," *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, vol. 4, no. 3, pp. 1–29, 2020.

[12] Y. He, Y. Chen, Y. Hu, and B. Zeng, "WiFi vision: Sensing, recognition, and detection with commodity MIMO-OFDM WiFi," *IEEE Internet of Things Journal*, vol. 7, no. 9, pp. 8296–8317, 2020.

[13] X. Shen, L. Guo, Z. Lu, X. Wen, and Z. He, "WiRIM: Resolution improving mechanism for human sensing with commodity Wi-Fi," *IEEE Access*, vol. 7, pp. 168357–168370, 2019.

[14] K. Niu, X. Wang, F. Zhang, R. Zheng, Z. Yao, and D. Zhang, "Rethinking Doppler effect for accurate velocity estimation with commodity WiFi devices," *IEEE Journal on Selected Areas in Communications*, 2022.

[15] B. Yu, Y. Wang, K. Niu, Y. Zeng, T. Gu, L. Wang, C. Guan, and D. Zhang, "WiFi-Sleep: sleep stage monitoring using commodity Wi-Fi devices," *IEEE Internet of Things Journal*, vol. 8, no. 18, pp. 13900–13913, 2021.

[16] E. G. Larsson, O. Edfors, F. Tufvesson, and T. L. Marzetta, "Massive MIMO for next generation wireless systems," *IEEE communications magazine*, vol. 52, no. 2, pp. 186–195, 2014.

[17] E. Björnson, E. G. Larsson, and M. Debbah, "Massive MIMO for maximal spectral efficiency: How many users and pilots should be allocated?," *IEEE Transactions on Wireless Communications*, vol. 15, no. 2, pp. 1293–1308, 2015.

[18] C. J. Vaca-Rubio, P. Ramirez-Espinosa, K. Kansanen, Z.-H. Tan, E. de Carvalho, and P. Popovski, "Assessing wireless sensing potential with large intelligent surfaces," *IEEE Open Journal of the Communications Society*, vol. 2, pp. 934–947, 2021.

[19] C. J. Vaca-Rubio, D. Salami, P. Popovski, E. De Carvalho, Z.-H. Tan, and S. Sigg, "User localization using RF sensing: A performance comparison between LIS and mmWave radars," in *2022 30th European Signal Processing Conference (EUSIPCO)*, 2022, pp. 1916–1920.

[20] C. J. Vaca-Rubio, R. Pereira, X. Mestre, D. Gregoratti, Z.-H. Tan, E. De Carvalho, and P. Popovski, "Floor map reconstruction through radio sensing and learning by a large intelligent surface," in *2022 IEEE 32nd International Workshop on Machine Learning for Signal Processing (MLSP)*, 2022, pp. 1-6.

[21] Z. Zhou, X. Gao, J. Fang, and Z. Chen, "Spherical wave channel and analysis for large linear array in LoS conditions," in *2015 IEEE Globecom Workshops (GC Wkshps)*, 2015, pp. 1–6.

[22] R. Brunelli, *Template matching techniques in computer vision: theory and practice*. John Wiley & Sons, 2009.

[23] A. Rosenfeld and J. L. Pfaltz, "Sequential operations in digital picture processing," *Journal of the ACM (JACM)*, vol. 13, no. 4, pp. 471–494, 1966.

[24] FEKO, Altair Engineering, Inc. [Online]. Available: `https://www.altairhyperworks.com/feko`

[25] P. Potkány, M. Debnár, M. Hitka, and M. Gejdoš, "Requirements for the internal layout of wooden house from the point of view of ergonomics changes," *Zeszyty Naukowe. Quality. Production. Improvement*, 2018.

[26] P. S. Hall, Y. Hao, Y. I. Nechayev, A. Alomainy, C. C. Constantinou, C. Parini, M. R. Kamarudin, T. Z. Salim, D. T. Hee, R. Dubrovka, and others, "Antennas and propagation for on-body communication systems," *IEEE Antennas and Propagation Magazine*, vol. 49, no. 3, pp. 41–58, 2007.

Paper D

User localization using RF sensing: A performance comparison between LIS and mmWave radars

Cristian J Vaca-Rubio, Dariush Salami, Petar Popovski, Elisabeth de Carvalho, Zheng-Hua Tan, Stephan Sigg

The paper has been published in the
IEEE European Signal Processing Conference (EUSIPCO) pp. 1-6, 2022.

© 2022 IEEE
The layout has been revised.

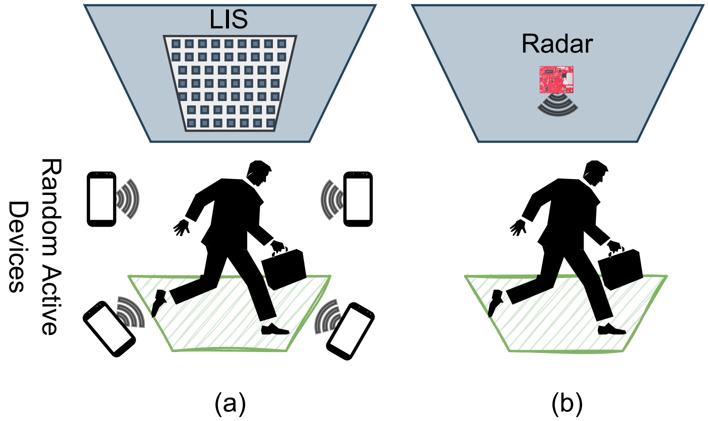


Fig. D.1: Schematic comparison of the two systems (a) LIS-based, (b) radar-based

Abstract

Since electromagnetic signals are omnipresent, Radio Frequency (RF)-sensing has the potential to become a universal sensing mechanism with applications in localization, smart-home, retail, gesture recognition, intrusion detection, etc. Two emerging technologies in RF-sensing, namely sensing through Large Intelligent Surfaces (LISs) and mmWave Frequency-Modulated Continuous-Wave (FMCW) radars, have been successfully applied to a wide range of applications. In this work, we compare LIS and mmWave radars for localization in real-world and simulated environments. In our experiments, the mmWave radar achieves 0.71 Intersection Over Union (IOU) and 3cm error for bounding boxes, while LIS has 0.56 IOU and 10cm distance error. Although the radar outperforms the LIS in terms of accuracy, LIS features additional applications in communication in addition to sensing scenarios.

Keywords— Sensing, machine learning, LIS, mmWave, radar, FMCW

1 Introduction

Recently, with the advance of robots, remote controlled and autonomous vehicles (UAVs, vehicles), localization and positioning has become a key support capability. RGB cameras [1], lidars [2], and RF sensors [3] are common approaches used for localization. Lidars are capable of providing an accurate 3D model of an environment. Their performance is affected by weather conditions though [4]. This is similar for RGB cameras, which also have a reduced performance in challenging lighting conditions [5]. RF-based sensors and LIS are resilient to lighting and weather conditions [5].

Wireless sensing is becoming integrated into future communication systems [6]. Specifically, with the increasing count of antennas added to Base Stations (BSs), researchers have exploited first Multiple-Input-Multiple-Output (MIMO), then massive MIMO, and now LIS, a continuous electromagnetic surface, comprising a large number of tightly spaced small antenna elements, capable of transmitting and receiving electromagnetic signals. LIS are capable to support communication as well as sensing [7]. Likewise, mmWave radars have been widely used in many sensing applications from gesture recognition [8–10] to localization [3].

In this work, we compare, for the first time in the literature, these two technologies to solve a sensing-based localization task as it is shown in Fig. D.1. We perform experimental measurements using an FMCW radar, namely IWR1443, operating in the 77-81 GHz RF-band. Since no LIS prototypes exist yet in the market, we perform a comprehensive simulation based on ray-tracing, mimicking the experimental environment. In particular, we will reconstruct the radio map of the propagation environment using an indoor LIS deployment in the ceiling, as also described in [11].

2 System Models and Problem Formulation

In this section, we discuss the configurations of the two systems. The LIS system is implemented in a simulation environment, while the radar system has a real-world test-bed.

2.1 LIS System Model

We consider an indoor setting in which D_a active devices are present. The devices transmit for communication. Sensing with the LIS is then subsequently performed by piggybacking these signals.

A LIS of A antenna elements is placed at the room's ceiling. We assume an ideal LIS made up of isotropic antennas. Therefore, physical effects such as mutual coupling are neglected. The sensing task is to create a radio map that captures a person in the environment by superimposing the received signals from each of the $d \in D_a$ elements at each of the A LIS components. The superposed complex baseband signal received at the LIS is given by

$$\mathbf{y} = \sum_{d=1}^{D_a} \mathbf{h}_d x_d + \mathbf{n}. \tag{D.1}$$

Here, x_d is the transmitted symbol from device d (without loss of generality, we consider $x_d = 1$), $\mathbf{h}_d \in \mathbb{C}^{N\times 1}$ is the channel vector from a specific position of device d to each antenna-element, and $\mathbf{n} \sim \mathcal{CN}_N(\mathbf{0}, \sigma^2\mathbf{I}_N)$ is the noise vector. For simplicity, we consider narowband transmissions and this removes the effect of frequency selectivity.

The large physical dimensions of the LIS, compared with the distance from the transmitters to the ceiling, lead to a spherical-wave propagation condition. The spherical-wave channel coefficient $h_{sp,a}$ at the a-th element from an arbitrary active device transmission is proportional to [12]

$$h_{sp,a} \;\propto\; \frac{1}{d_a} e^{-j\frac{2\pi}{\lambda}d_a}, \tag{D.2}$$

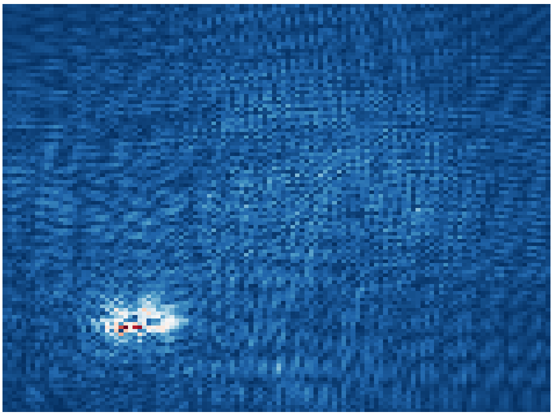


Fig. D.2: A radio map obtained for an $M = 118 \times 118$ antenna elements $\frac{\lambda}{2}$ spaced LIS in a noiseless scenario with $D_a = 5$ active devices by using a MF design for $f = 3.5$ GHz, $d = 3.2$ m and $N_f = 100\times100$ antenna elements $\frac{\lambda}{2}$ spaced.

where d_a denotes the distance between the active device and the a-th antenna. To compute the radio map, we can derive a Matched Filter (MF) such that [11]

$$\mathbf{y}_{mf} = \mathbf{h}_{sp} * \mathbf{y}. \tag{D.3}$$

In eq D.3, $*$ denotes the convolution operator, $\mathbf{h}_{sp} \in \mathbb{C}^{N_f \times 1}$ is the expected spherical pattern (steering vector) for N_f filter-antennas LIS deployment on (D.2), $\mathbf{y}$ is the received signal from (D.1) and $\mathbf{y}_{mf} \in \mathbb{C}^{A\times 1}$ is the filtered output that represents the radio map. We zero-pad $\mathbf{y}$ such that we guarantee $\mathbf{y}_{mf} \in \mathbb{C}^{A\times 1}$ dimension to perform the convolution. To obtain the radio map, we compute the energy at the output of the MF. In this way, we are measuring the energy of the signals reflected from the target. In order to design the filter, we assume knowledge of the frequency f and the distance d (the z coordinate, since the LIS is deployed at the ceiling) between the transmitter and the LIS[1]. Figure D.2 shows the detection of the target on the LIS processed radio map.

Simulated scenario

We perform simulations via ray tracing [13] in a $4.7 \times 4.7 \times 3.2$ m simulated area. We deploy a LIS with 118×118 elements separated by $\lambda/2$. The D_a active devices transmit narrowband signals of 20 dBm at 3.5 GHz and they are randomly deployed in the space (x, y, z). The

---

[1] A detailed explanation of the radio maps is given in [11]. The distance d is a parameter for the filter design and may differ from the actual distance to the transmitter in the evaluation.

distance from which the MF is calibrated is $d = 3.2$, as we set a kernel for the procedure such that it scans the entire height of the room. The human is modeled as a rectangle of dimensions 0.3x0.5x1.83 m (reasonable human dimensions obtained from [14]) with conductivity $s = 1.44$ S/m, relative permittivity $\varepsilon = 38.1$ and relative permeability $\mu = 1$ [15]. To be fair with the experimental setup, the positions of the passive human $(x, y, z = 1.83)$ m are randomly selected in a circle of radius 2.5 m.

Received signal and noise modeling

From the ray-tracing simulation, the received signal in (D.1) is obtained as the complex electric field arriving at the a-th antenna element, $\widetilde{E}_a$, which can be regarded as the superposition of each ray path $r \in N_r$ from every $d \in D_a$ device, i.e.,

$$\widetilde{E}_a = \sum_{d=1}^{D_a} \sum_{r=1}^{N_r} \widetilde{E}_{a,r,d} = \sum_{d=1}^{D_a} \sum_{r=1}^{N_r} E_{a,r,d} e^{j\phi_{a,r,d}}. \tag{D.4}$$

The complex signal at the output of the a-th element is therefore given by

$$y_a = \sqrt{\frac{\lambda^2 Z_a}{4\pi Z_0}} \widetilde{E}_a + n_a. \tag{D.5}$$

Here, λ is the wavelength, $Z_0 = 120\pi$ os the free space impedance and Z_n is the antenna impedance. For simplicity, we consider $Z_a = 1 \, \forall \, a$. Finally, we define the average Signal-to-Noise Ratio (SNR), $\overline{\gamma}$, as

$$\overline{\gamma} \triangleq \frac{\lambda^2}{4\pi Z_0 A \sigma^2} \sum_{a=1}^{A} |\widetilde{E}_a|^2, \tag{D.6}$$

where A denotes the number of antenna elements in the LIS.

2.2 Radar System Model

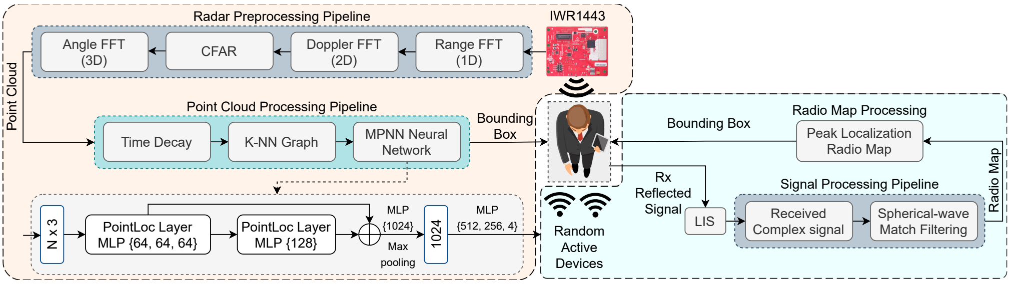
We utilize the IWR1443 mmWave FMCW radar operating at 77GHz with 4GHz bandwidth. We further process the point cloud data generated by the radar to perform localization.

Point Cloud Generation

The radar evaluation kit performs a four step processing pipeline to generate point cloud data from Analog to Digital Converter (ADC) data as shown in Fig. D.3.a.

Range Fast Fourier Transform (FFT) (1D): A chirp signal, i.e. a sinusoidal signal with increasing frequency through time, is generated by the radar. Then the mixing operation produces an intermediate frequency signal using transmitted and reflected chirps. The range of the system is linearly proportional to the intermediate signal's frequency, computed through FFT operations.

Doppler FFT (2D): The radial velocity of the target is proportional to the phase difference of multiple chirps at the range-FFT peak. To generate a peak at the velocity of target, a FFT is applied on the signal range (i.e., a 2D-FFT or a Doppler-FFT).



(a) Radar System Model **(b) LIS System Model**

Fig. D.3: (a) Overview structure of the radar-based localization system. The radar transforms the IQ samples into a point cloud through the radar processing pipeline. Then, the point cloud processing pipeline predicts a bounding box using as the localization output. (b) Overview structure of the LIS-based localization system. The LIS processes the reflected complex signal by applying the MF procedure to obtain the radio map. Then, the radio map processing part predicts a bounding box around the target user. The random D_a active devices are piggybacked to perform the sensing.

Constant False Alarm Rate (CFAR): For each virtual antenna, a pre-detection matrix is formed through summation of Doppler-FFT matrices. The CFAR algorithm [16] then filters the pre-deteciton matrix to produce peaks corresponding to targets.

Angle-FFT: For each target, an angle-FFT is applied on corresponding CFAR peaks across multiple Doppler-FFTs. Velocity-induced phase changes are Doppler-corrected before angle-FFT calculation.

Point Cloud Processing

We follow a three-step point cloud processing tool-chain to localize the target.

Time Decay: We configured the radar to sample 33 point clouds per second[2]. Since points in each frame are sparse (≈ 5 per frame), we apply time decay to increase density. Empirically, a time-window of size 5 (≈ 151 ms) is used to slide through time frames to increase the number of points in each frame.

Graph Generation: From the time-decayed point cloud, we then build a graph. Consider a point cloud $X = \{x_1, ..., x_n\} \subseteq \mathbb{R}^F$ where each point x_i is represented by a feature set $\{f_i^1, ..., f_i^F\}$. A graph-K-Nearest Neighbor (KNN) algorithm is exploited to produce the KNN-graph $\mathcal{G} = \{X, \mathcal{E}\}$ where $\mathcal{E} \subseteq X \times X$ is the set of directed edges between each point and its nearest neighbours in the Euclidean space.

Graph Processing using Message Passing Neural Network (MPNN): Following a similar approach as [17], each node's representation is updated using the proposed MPNN layer. In each layer, an aggregation function is applied to the representation of the node itself and to its neighbors' features:

---

[2] we will refer to the points generated from a single chirp as frames in the following

$$
\begin{aligned}
h_i^0 &= x_i, \\
h_i^l &= \underset{j:(i,j)\in\mathcal{E}}{\Gamma}(\Delta_\theta(h_i^{l-1}, h_j^{l-1})),
\end{aligned}
\tag{D.7}
$$

in which, h_i^l is point i's representation in layer l, message function $\Delta_\theta : \mathbb{R}^F \times \mathbb{R}^F \rightarrow \mathbb{R}^{F'}$ is a learnable function with parameters θ which is implemented using Multi Layer Perceptron (MLP), and Γ is a channel-wise symmetric aggregation operator (e.g. Σ, max, or mean) applied on the messages of the edge emanating from each neighbor. To capture not only local but also global structures of the input point cloud, we use $\Delta_\theta(h_i, h_j) = \bar{\Delta}_\theta(h_i, h_j - h_i)$ meaning that for each edge, we concatenate the features of the incident node of the edge with the difference between the two nodes connected by the edge.

The details of the localization model is shown in Fig. D.3. We empirically use the proposed layer in Eq.D.7, PointLoc, twice in the architecture of the model. Then we concatenate the output of the two layers followed by an MLP to achieve a vector representation of the input point cloud with length of 1024. Finally, after applying a few layers of MLP to extract fine-grained features, we use a linear layer with 4 neurons and without activation function to output the four dimensions of the bounding box. The loss function is Mean Squared Error (MSE).

2.3 Main Assumptions

The main assumptions for both approaches are as follows. With respect to LIS, we assume that mutual coupling is ignored. It is commonly represented using a coupling matrix that takes into account the influence of nearby antennas [18] but the influence can be compensated after estimation, and hence does not affect any conclusion drawn. Radar and LIS have a Line-of-Sight (LOS) to the target. Also, to design the filter we need to assume a-priori the frequency f and the distance from a transmitter to the LIS d. However, this last parameter is not a very strong assumption (cf. Section 2.1). We therefore set it to be the distance as if the transmitter were on the floor. This means that the transmitters can be at different distances and are not limited by the filter design.

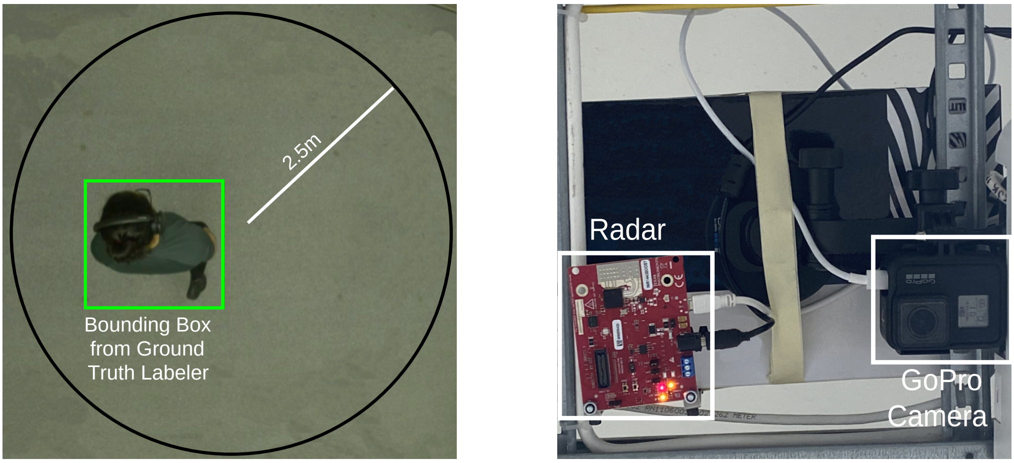
The field of view of the radar is -55° to +55°.

3 Implementation and Evaluation

In this section, we explain the way we collected data in the real-world environment for the mmWave radar and simulation environment for LIS. Then, we elaborate on the model training process, evaluation metrics, and finally the localization performance comparison between the radar and the LIS.

3.1 Dataset

To collect the radar data (cf. Fig. D.4a), we marked a circular area with 2.5m radius on the floor and asked participants to walk arbitrarily inside the circle. A IWR1443 radar and a GoPro camera were mounted in 3m height over the center of the circle record ground truth point cloud



(a) (b)

Fig. D.4: (a) The experiment environment which is a circle with 2.5m radius marked on the ground. (b) An IWR1443 radar and a GoPro camera installed on the ceiling with 3m height.

and video data (Fig. D.4b). We labelled the video using the Ground Truth Labeler application. The radar data was labeled manually by matching the bounding boxes from the video with the point cloud data. Three subjects participated in the experiment and 15,000 samples have been collected. Training, validation and testing is performed following a 60/10/30 split.

For the LIS, we simulated the same circular area with 2.5m radius and simulated random positions of the target. The positions are processed by obtaining radio maps using $D_a = 5$ active devices randomly deployed in the room. 15,000 samples were recorded.

3.2 Radar Model Training Process

We used a computer with 32GB of RAM and an Nvidia GeForce 940MX GPU for training and inference. The model for the radar pipeline is implemented based on PyTorch and PyTorch Geometric [19] frameworks. An early stopping scheme with a patience of 100 epochs is exploited to prevent from over-fitting. In other words, if no improvement is observed on the validation set within the patience period, we stop the training process and save the best model.

MSE is used as the loss function to train the model, Adam Optimizer [20] with step-decay strategy to train the network:

$$L_r = L_i \cdot d_r^{\lfloor \frac{e}{e_r} \rfloor}. \tag{D.8}$$

Here, L_r is the learning, L_i is the initial value of the learning rate, d_r is the drop rate after every e_r epochs, e is the current epoch and $\lfloor \cdot \rfloor$ is the floor operator. In our experiments L_i, d_r, and e_r are 0.001, 0.5, and 20, respectively.

3.3 LIS Radio Map Based Localization

We process the radio map by obtaining the indices of the maximum peak value to infer the human position, i.e.

$$(x_c, y_c) = \arg\max_{xp,yp} |\mathbf{y_{mf}}| \times \Delta s, \tag{D.9}$$

In this equation, x_c and y_c denote the inferred position, x_p and y_p the position in the pixel domain and Δs the antenna spacing.

3.4 LIS Bounding Box Creation

As we are not using a learning algorithm, but want to compare LIS and radar systems, we compute manually a bounding box considering the center the inferred positions.

As we know the dimensions of our target (detailed in Section 2.1) we can compute a bounding box from the center position such that

$$(x_{min}, y_{min}) = (x_c - \frac{L}{2}, y_c - \frac{W}{2}), \tag{D.10}$$

$$(x_{max}, y_{max}) = (x_c + \frac{L}{2}, y_c + \frac{W}{2}), \tag{D.11}$$

with $L = 0.5$ m and $W = 0.3$ m (cf. Section 2.1).

3.5 Evaluation Metric

To evaluate the performance of the models we use IOU and the Euclidean distance between the center of the ground truth bounding box and the predicted bounding box. The IOU is defined by:

$$IOU_i = \frac{|P_i \cap G_i|}{|P_i \cup G_i|}, \tag{D.12}$$

where P_i and G_i are predicted and ground truth bounding boxes for sample i. We report the average IOU and Euclidean distance between the centers of the ground truth and the predicted bounding boxes on the test set for each of the approaches.

3.6 Localization Comparison

As shown in Fig. D.5, the average IOU for the LIS in localization is 0.56 while the Radar it is 0.71. This result suggests that the radar is more accurate in estimating the bounding box for the target. We have a similar result for the average distance error of the bounding boxes. The radar outperforms the LIS with only 3cm in contrast to 10cm average error. These results suggest that mmWave radars are more accurate than LIS, but in scenarios where the accuracy of sensing can be compromised, LIS can provide simultaneous communication and sensing capabilities without a need for a third-party device. Also, there might be potential scenarios in which LIS would outperform the radar. For example, decreasing the inter-antenna

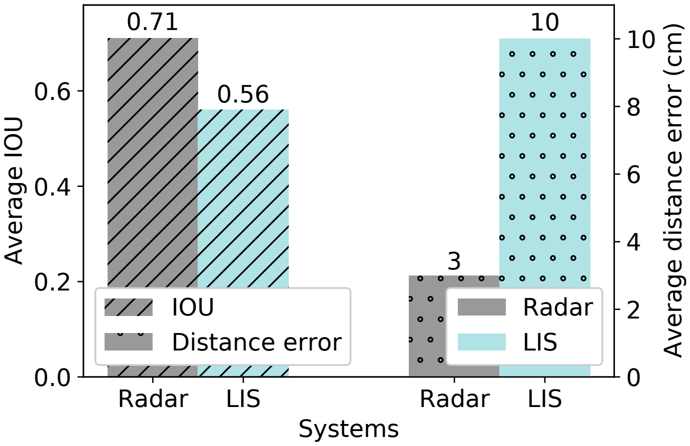


Fig. D.5: Average IOU and distance error for radar and LIS

spacing would lead to more accurate localization results for the LIS. However, we showed the common inter-antenna distance design ($\frac{\lambda}{2}$) to provide a close comparison to the radar.

4 Conclusion

So far, mmWave radars have been considered the standard de facto for indoor localization given their incredibly cheap components and localization accuracy. However, the breakthroughs in communications systems are leading to devices that can integrate both sensing and communications. LISs arise as a brand-new device which communications capabilities are expected to go beyond massive MIMO. However, their sensing functionality is still under investigation. In this paper, we have shown a comparison among LIS and mmWave radars in an indoor sensing-based localization task. Results show that LIS is not that different from mmWave radars in localization accuracy, with the advantage of leading to a new system that can integrate both sensing and communications. In most scenarios there is no need to have a very accurate localization, including but not limited to industrial localization systems or localizing users for beamforming. In those scenarios, LIS can also act as a sensing device providing communications and sensing capabilities at the same time.

5 Acknowledgements

This project has received funding from the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie Grant agreement No. 813999.

References

[1] K. Kononov, R. Lavrenov, L. Gavrilova, and T. Tsoy, "External RGB-D Camera Based Mobile Robot Localization in Gazebo Environment with Real-Time Filtering and Smoothing Techniques," in *Electromechanics and Robotics*, 2022, pp. 223–234.

[2] Y. Zhang, L. Wang, X. Jiang, Y. Zeng, and Y. Dai, "An efficient lidar-based localization method for self-driving cars in dynamic environments," *Robotica*, vol. 40, no. 1, pp. 38–55, 2022.

[3] S. Palipana, B. Pietropaoli, and D. Pesch, "Recent advances in RF-based passive device-free localisation for indoor applications," *Ad Hoc Networks*, vol. 64, pp. 80–98, 2017.

[4] Y. Li, Y. Liu, Y. Wang, Y. Lin, and W. Shen, "The millimeter-wave radar SLAM assisted by the RCS feature of the target and IMU," *Sensors*, vol. 20, no. 18, p. 5421, 2020.

[5] S. Palipana, D. Salami, L. A. Leiva, and S. Sigg, "Pantomime: Mid-air gesture recognition with sparse millimeter-wave radar point clouds," *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, vol. 5, no. 1, pp. 1–27, 2021.

[6] M. Latva-aho and K. Leppänen, "Key drivers and research challenges for 6G ubiquitous wireless intelligence (white paper)," *6G Flagship research program, University of Oulu, Finland*, 2019.

[7] C. J. Vaca-Rubio, P. Ramirez-Espinosa, K. Kansanen, Z.-H. Tan, E. de Carvalho, and P. Popovski, "Assessing Wireless Sensing Potential with Large Intelligent Surfaces," *arXiv preprint arXiv:2011.08465*, 2020.

[8] D. Salami and S. Sigg, "Zero-Shot Motion Pattern Recognition from 4D Point-Clouds," in *MLSP 2021*, 2021, pp. 1–6.

[9] D. Salami, R. Hasibi, S. Savazzi, T. Michoel, and S. Sigg, "Integrating Sensing and Communication in Cellular Networks via NR Sidelink," *arXiv preprint arXiv:2109.07253*, 2021.

[10] D. Salami, R. Hasibi, S. Palipana, P. Popovski, T. Michoel, and S. Sigg, "Tesla-Rapture: A Lightweight Gesture Recognition System from mmWave Radar Sparse Point Clouds," *IEEE Transactions on Mobile Computing*, 2022.

[11] C. J. Vaca-Rubio, P. Ramirez-Espinosa, K. Kansanen, Z.-H. Tan, and E. de Carvalho, "Radio sensing with large intelligent surface for 6G," in *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2023, pp. 1–5.

[12] Z. Zhou, X. Gao, J. Fang, and Z. Chen, "Spherical wave channel and analysis for large linear array in LoS conditions," in *Globecom Workshops*, 2015, pp. 1–6.

[13] FEKO, Altair Engineering, Inc. Available online: `https://www.altairhyperworks.com/feko`.

[14] P. Potkány, M. Debnár, M. Hitka, and M. Gejdoš, "Requirements for the internal layout of wooden house from the point of view of ergonomics changes," *Zeszyty Naukowe. Quality. Production. Improvement*, 2018.

[15] P. S. Hall, Y. Hao, Y. I. Nechayev, A. Alomainy, C. C. Constantinou, C. Parini, M. R. Kamarudin, T. Z. Salim, D. TM Hee, R. Dubrovka, et al., "Antennas and propagation for on-body communication systems," *IEEE Antennas and Propagation Magazine*, vol. 49, no. 3, pp. 41–58, 2007.

[16] M. A. Richards, J. Scheer, W. A. Holm, and W. L. Melvin, "Principles of modern radar," 2010.

[17] Y. Wang, Y. Sun, Z. Liu, S. E. Sarma, M. M. Bronstein, and J. M. Solomon, "Dynamic graph cnn for learning on point clouds," *ACM Transactions on Graphics (TOG)*, vol. 38, no. 5, pp. 1–12, 2019.

[18] T. Su and H. Ling, "On modeling mutual coupling in antenna arrays using the coupling matrix," *Microwave and Optical Technology Letters*, vol. 28, no. 4, pp. 231–237, 2001.

[19] M. Fey and J. E. Lenssen, "Fast graph representation learning with PyTorch Geometric," *arXiv preprint arXiv:1903.02428*, 2019.

[20] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

Paper E

Floor map reconstruction through radio sensing and learning by a large intelligent surface

Cristian J Vaca-Rubio, Dariush Salami, Petar Popovski, Elisabeth de Carvalho, Zheng-Hua Tan, Stephan Sigg

The paper has been published in the
IEEE Workshop on Machine Learning for Signal Processing (MLSP), pp. 1–6, 2022.

© 2022 IEEE
The layout has been revised.

Abstract

Environmental scene reconstruction is of great interest for autonomous robotic applications, since an accurate representation of the environment is necessary to ensure safe interaction with robots. Equally important, it is also vital to ensure reliable communication between the robot and its controller. Large Intelligent Surface (LIS) is a technology that has been extensively studied due to its communication capabilities. Moreover, due to the number of antenna elements, these surfaces arise as a powerful solution to radio sensing. This paper presents a novel method to translate radio environmental maps obtained at the LIS to floor plans of the indoor environment built of scatterers spread along its area. The usage of a Least Squares (LS) based method, U-Net (UN) and conditional Generative Adversarial Networks (cGANs) were leveraged to perform this task. We show that the floor plan can be correctly reconstructed using both local and global measurements.

Keywords— Sensing, Computational Imaging, LIS, Machine Learning for Communication.

1 Introduction

Mobile robots with mapping devices have been used to estimate indoor floor plans. The state-of-the-art methods are usually based on optical sensors to obtain the indoor maps in detail. For instance, Light Detection and Ranging (LIDAR) [1], depth cameras [2] and RGB cameras [3]. Although these methods achieve acceptable accuracy, they have some limitations. LIDAR might not be able to capture all types of materials meanwhile approaches based on cameras are dependent on the lighting conditions. To circumvent this, acoustic-based approaches such as microphones [4] and ultrasonic sensors [5] are robust to lighting conditions. However, they have a limited sensing range and might malfunction in noisy environments.

On a related note, radar devices such as millimeter Wave (mmWave) radars have become popular for indoor sensing applications. They actively transmit RF signals to monitor the reflections to sense nearby scatters' parameters such as range, speed or angle. Hence, they can be used indoors in poor lighting conditions. These radars have been used in applications such as human sensing [6] and floor-plan reconstruction [7]. However, mmWave radars work in high frequency bands, leading to short wavelengths. This leads to high energy attenuation over distance and weak penetrability through walls.

In the context of 6G, sensing has become a fundamental feature. For instance, the authors in [8] explore this sensing-style capabilities of a mMIMO BS to jointly learn an antenna selection and a range-azimuth map of a beamforming gain. With the increasing number of receivers, LIS becomes a natural extension of the massive MIMO technology which designates a continuous electromagnetic surface able to transmit and receive radio waves. In practice, they are planar arrays conformed by a huge amount of closely spaced tiny antenna elements. There is a vast range of studies analyzing its application in communications [9]. However, very little has been studied regarding its sensing capability [10]. Consequently, in [11] we presented a method that enables reconstructing a radio map of the propagation environment using an indoor LIS deployment in the ceiling. This allows for tracking both active and passive users.

Motivated by these results and on the increasing interest in both sensing and LIS, in this work, we further explore the capabilities of LIS and provide a floor plan reconstruction based on the LIS received signal. We leverage the usage of Deep Generative Model (DGM) to learn a map from the complex-valued received signals at the LIS to the RGB image of the corresponding floor plan. Differently from previous works [1, 4, 7], we are mainly concerned with enabling sensing capabilities in a system primarily designed for communication. Consequently, such applications further exploit the usability of already deployed hardware. We assess the reconstruction performance with the original (ground-truth) floor plan composed of multiple elements representing, for example, robots, furniture and appliances

2 System and Problem Formulation

Consider an indoor factory scenario in which K_a active devices are deployed. These devices can be anything using a transmitter such as robots, smartphones or IoT devices. More importantly, since they are active devices, we can assume these devices to be communicating with a receiver through a wireless channel. In a real-world scenario, this communication often assists with the task at hand, e.g., sending commands to the robots. In our scenario, this receiver is a LIS placed on the factory's roof and, apart from assisting with the communication task, the LIS also tries to map the current environment of this factory. This environment sensing can be divided into two steps:

1. Mapping from the *environmental signals* into a radio map of the environment from the LIS viewpoint, similar to [11]. This allows converting the complex-valued data into a RGB real-valued data which is more natural to neural networks.
2. Translation of the radio map into its current floor plan pattern, i.e., the disposition of the elements in this environment. In summary, based on the raw complex-valued signals received at the LIS, we aim to reconstruct the current arrangement of the passive (non-active) elements present in the environment.

For the second step, we rely on conditional Generative Adversarial Networks (cGANs) which has been widely used in the literature for the task of image-to-image reconstruction [12].

Moreover, we assume this LIS to be equipped with $N = N_x \times N_y$ antenna elements and its physical aperture comprises its whole area. Concretely, we consider a square LIS composed of isotropic antennas and physical effects such as mutual coupling are ignored. A bit more formally, the sensing problem consists in obtaining a radio map that describes the environment from the superposition of the received signals from each of the $1 < k \leq K_a$ users at every element of the LIS. This map contains information on the K_a active devices involved in the scenario as well as the K_p passive objects/scatters.

The superposed complex baseband signal received at the LIS is given by

$$\mathbf{y} = \sum_{k=1}^{K_a} \mathbf{h}_k x_k + \mathbf{n}, \tag{E.1}$$

with x_k the transmitted (sensing) symbol from user k (we consider $x_k = 1$ without loss of generality), $\mathbf{h}_k \in \mathbb{C}^{N\times 1}$ the channel vector from a specific position of user k to each antenna-element, and $\mathbf{n} \sim \mathcal{CN}_N(\mathbf{0}, \sigma^2\mathbf{I}_N)$ the noise vector. To avoid frequency selectivity, we consider a narrowband transmission.

As a consequence of the large physical dimensions of the LIS in comparison with the distance from the transmitters to the roof, we need to account for spherical wave propagation during the modeling. The spherical-wave channel coefficient $h_{sp,n}$ at the LIS n-th element from an arbitrary active device transmission is proportional to [13]

$$h_{sp,n} \propto \frac{1}{d_n} e^{-j\frac{2\pi}{\lambda}d_n}, \tag{E.2}$$

where $d_n = \sqrt{(x_n - x_k)^2 + (y_n - y_k)^2 + (z_n - z_k)^2}$ denotes the distance between the active device k and the n-th antenna. To obtain the radio map, expressing the LIS in a vectorized version for ease of notation, we can derive a MF procedure such that [11]

$$\mathbf{y}_{mf} = \mathbf{h}_{sp} * \mathbf{y}, \tag{E.3}$$

denoting $*$ the spatial convolution operator, $\mathbf{h}_{sp} \in \mathbb{C}^{N_f \times 1}$ the expected spherical pattern (steering vector) for N_f antennas LIS deployment on (E.2), $\mathbf{y}$ the received signal from (E.1) and $\mathbf{y}_{mf} \in \mathbb{C}^{N \times 1}$ the filtered output signal. As the convolution operator would reduce the output dimension (Due to the 2D convolution along the received signal at the LIS), we zero-pad $\mathbf{y}$ such that we guarantee $\mathbf{y}_{mf} \in \mathbb{C}^{N \times 1}$ dimension. To obtain a radio map, we just need to compute the energy at the output of the MF procedure $|\mathbf{y}_{mf}| \in \mathbb{R}^{N \times 1}$. We then map the values to the RGB scale such that $F : \mathbb{R}^{N \times 1} \rightarrow \{[0, 255] \cap \mathbb{N}\}^{N \times 3}$

$$\mathbf{y}_m = F(|\mathbf{y}_{mf}|) \tag{E.4}$$

that represents the radio map. Please note, we have to know the frequency f and assume a distance z from the transmitter to the LIS to design the filter[1]. In our work, we design a filter for $f = 3.5$ GHz, $z = 8$ m and $N_f = 100 \times 100$ antenna elements $\frac{\lambda}{2}$ spaced[2]. Let us assume we can obtain S samples at each channel coherence interval. In this way, we can use these extra samples to perform an S-averaging of the received signal measurements at the LIS viewpoint, reducing the noise variance contribution and enhancing the quality of the obtained radio map. Figure E.1(a) shows a ground-truth floor plan. Looking at Figure E.1(b)-(d) we can see three exemplary radio maps representing the ground-truth floor plan. The radio map captures the reflections of the scatters (rectangular and square shapes representing walls/objects) that act as virtual sources. The S-averaging effect in reducing the noise contribution leads to an enhancement in the radio map quality. The target of our model will be translating these radio maps to the ground-truth floor plan.

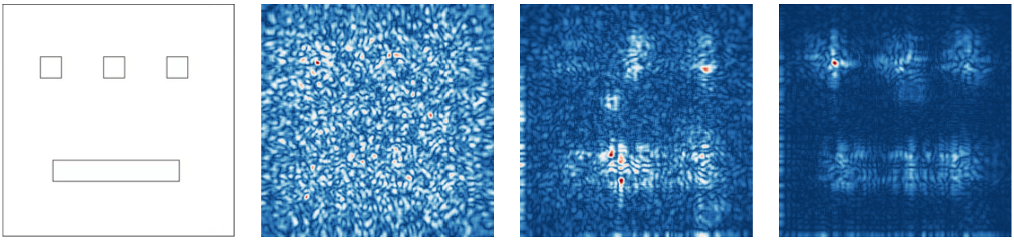
2.1 Received signal and noise modeling

In order to simulate the propagation environment in the most reliable way, we resort to ray tracing [14]. From the ray-tracing simulation, the received signal in (E.1) is obtained as the

---

[1] For a more detailed explanation of the radio maps, we gently refer the readers to [11]. The distance z is a parameter for the filter design. This does not imply that in the evaluation, all the transmitters or scatters are fixed at this distance.

[2] Note we use a fixed kernel with $z = 8$ m which corresponds to the height of the building. We do not need to calibrate the MF to specific distances.



(a) Floor Plan **(b)** $S = 1$ **(c)** $S = 100$ **(d)** $S = 1000$

Fig. E.1: Radio maps with its corresponding floor plan obtained over S-averaging channel samples acquired by LIS in a $\gamma = -10$ dB SNR condition.

complex electric field arriving at the n-th antenna element, $\widetilde{E}_n$, which can be regarded as the superposition of each ray path $r \in N_r$ from every $k \in K_a$ user, i.e.,

$$\widetilde{E}_n = \sum_{k=1}^{K_a} \sum_{r=1}^{N_r} \widetilde{E}_{n,r,k} = \sum_{k=1}^{K_a} \sum_{r=1}^{N_r} E_{n,r,k} e^{j\phi_{n,r,k}}. \tag{E.5}$$

Then, the complex signal at the output of the n-th element is therefore given by

$$y_n = \sqrt{\frac{\lambda^2 Z_n}{4\pi Z_0}} \widetilde{E}_n + n_n, \tag{E.6}$$

with λ the wavelength, $Z_0 = 120\pi$ the free space impedance and Z_n the antenna impedance. For simplicity, we consider $Z_n = 1 \,\forall\, n$. Finally, we define the SNR, γ, as

$$\gamma \triangleq \frac{\lambda^2}{4\pi Z_0 N \sigma^2} \sum_{n=1}^{N} |\widetilde{E}_n|^2, \tag{E.7}$$

where N denotes the number of antenna elements in the LIS.

3 Reconstruction learning

3.1 Least Squares

This method tries to naively find the best linear mapping $\mathbf{W}^* \in \mathbb{R}^{N \times N}$ from the i-th radio map $\mathbf{y}_m^{(i)} \in \mathbb{R}^N$ into the ith floor plan $\mathbf{x}^{(i)} \in \mathbb{R}^N$ by minimizing the average least square error

$$\mathbf{W}^* = \arg\min_{\mathbf{W}} \frac{1}{\mathrm{T}} \sum_{i=1}^{\mathrm{T}} \left(\mathbf{x}^{(i)} - \mathbf{W}\mathbf{y}_m^{(i)}\right)^2 \tag{E.8}$$

over the T training samples. Prediction is then performed on a new sample by $\hat{\mathbf{x}}^{(j)} = \mathbf{W}\mathbf{y}_m^{(j)}$. Despite its simplicity, similar solutions have been widely applied in many signal-to-image reconstruction, such as medical applications [15] and synthetic aperture radars sensing [16]. In our scenario, LS method often returns a noisy version of the environment reconstruction. Hence, to simulate the best possible set of post-processing operations that improve the LS estimator, we redefine $\hat{\mathbf{x}}^{(j)} = \min(\mathbf{x}^{(j)}, \hat{\mathbf{x}}^{(j)})$ with the entry-wise minimum operator, i.e., we remove the noise area that is present outside of the area of interest.

3.2 U-Net

U-nets are essentially autoencoder networks with skipped connections [17]. At the encoder side, we learn the feature mapping of an image while converting it to a vector. U-Net extends this vector to a segmented picture, utilizing the same feature maps as were used for compression at the encoder side (i.e. skipped connections). This would keep the image's structural integrity and provide information to the decoder to perform the segmentation. Image segmentation can be applied to our specific problem. We can see the groundtruth floor plan $\mathbf{x}$ as the segmentation of the radio map $\mathbf{y}_m$ composed of two classes, background and shapes. To reduce training time and prevent overfitting, we will use a pre-trained model for the encoder, the mobileNet-v2 [18], while for the decoder we will use the up-sampling block of Pix2Pix [12]. Finally, the U-Net minimizes the Binary Cross-Entropy (BCE) loss.

3.3 Conditional Generative Adversarial Network

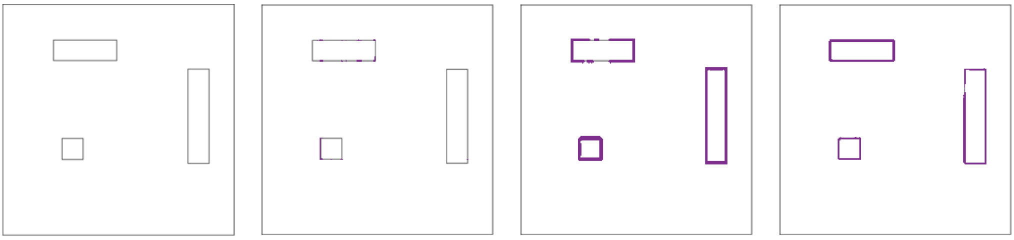
We design our problem as a cGANs learning procedure by designing a generator for the distribution p_g over the floor plan data of the environment $\mathbf{x}$, given as conditional information the radio map $\mathbf{y}_m$ from (D.3). The generator models a mapping function from a prior noise distribution $p_{\mathbf{z}}(\mathbf{z})$ to the floor plan space $G(\mathbf{z}|\mathbf{y}_m; \theta_g)$. Similarly, we design a discriminator $D(\mathbf{x}|\mathbf{y}_m; \theta_d)$ that outputs a scalar value representing the probability that $\mathbf{x}$ came form training data instead of p_g. The target min-max cost function is given by

$$\mathcal{L}_{cGANs} = \min_{G} \max_{D} \mathbb{E}_{\mathbf{x}\sim p_{\text{data}}(\mathbf{x})}[\log D(\mathbf{x}|\mathbf{y}_m)] + \mathbb{E}_{\mathbf{z}\sim p_{\mathbf{z}}(\mathbf{z})}[1 - \log D(G(\mathbf{z}|\mathbf{y}_m))]. \tag{E.9}$$

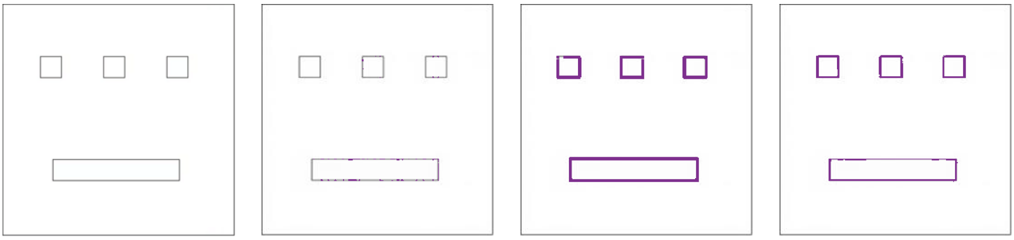
More specifically, we re-implement a well-known model used in the field of image-to-image translation [19]. The main difference with respect to the general cGANs framework resides in the loss function. The authors of [19] improve the adversarial loss by incorporating a feature matching loss. It works by extracting features from several layers of the discriminator, trying to match these in-between representations among the real and the fake data. The new loss function can be expressed as

$$\mathcal{L}^*_{cGANs} = \min_{G} \max_{D} \mathcal{L}_{cGANs} + \lambda \mathbb{E}_{\mathbf{x},\mathbf{y}_m,\mathbf{z}} \sum_{l=1}^{L} \frac{1}{N_l} [\|D^{(l)}(\mathbf{x}|\mathbf{y}_m) - D^{(l)}(\mathbf{z}|\mathbf{y}_m)\|_1], \tag{E.10}$$

where L denotes the number of layers, N_l denotes the number of elements in each layer and λ is an hyper-parameter that controls the weight of the terms. The model will learn a mapping between the wireless environmental signals and its corresponding floor plan, i.e., it will perform a translation from the radio map $\mathbf{y}_m$ to the floor plan $\mathbf{x}$.



(a) Scenario 1 - Original (c) Scenario 1 - LS (e) Scenario 1 - U-Net (g) Scenario 1 - cGANs



(b) Scenario 2 - Original (d) Scenario 2 - LS (f) Scenario 2 - U-Net (h) Scenario 2 - cGANs

Fig. E.2: Visual representation of the two original (a)-(b) floor plans and their and reconstructions (c)-(h) using the three different methods. Reconstructions are highlighted in purple and the original floor plans in black.

4 Dataset description

In the ray-tracing simulation, we consider two scenarios of size $10.34 \times 10.34 \times 8$ m with different scatters arrangement. We deploy an LIS with 259×259 elements separated $\lambda/2$ apart. Every K_a active device transmits a narrowband signal of 20 dBm at 3.5 GHz. The distance from which the MF is calibrated is $z = 8$ m (building height). The scatters are modeled as metallic (with conductivity $s = 19444$ S/m, relative permittivity $\epsilon = 1$ and relative permeability $\mu = 20$) and the walls as brick (with $s = 0.078$ S/m, $\epsilon = 4$ and $\mu = 1$)[3]. Figure E.2(a)-(b) show the distribution of the metallic elements in the two floor plans considered in this work. The dataset is obtained by sampling the received signal at each element of the LIS from all the different active K_a in the scenario. Then the resulting signal is processed following the process described in Section II to generate the radio map.

To guarantee generalization, we conform a dataset composed by a selection of $S \in [1, 100, 1000]$ and $K_a \in [5, 20]$ for the two scenarios presented. Finally, we are primarily interested in poor signaling conditions, hence, we set $\gamma = -10$ dB which means that there is much noise in the communication between the active transmitters and the LIS. Then, formally, the dataset is denoted as $\{\mathbf{y}_m^{(i)}, \mathbf{x}^{(i)}\}_{i=1}^{T}$, where $\mathbf{y}_m^{(i)}$ is the i-th N-dimensional training input features vec-

---

[3] These values are provided by the software manual [14].

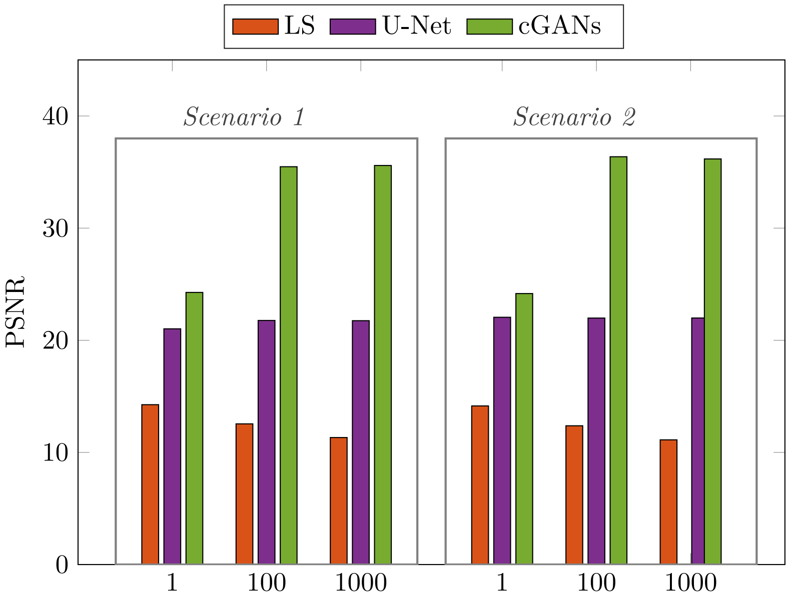


Fig. E.3: Comparison of floor plan reconstruction quality based on the Average Peak Signal-to-Noise Ratio (PSNR) metric along the test set for varying number of S.

tor (radio map) and $\mathbf{x}^{(i)}$ is the i-th N-dimensional target vector (floor plan). Our dataset is composed of 2400 samples which are split into 70/10/20 for training, validation and test set, respectively.

5 Numerical results and Discussion

In order to validate the proposed methodology, we rely on standard image quality metrics, to evaluate the performance of the reconstructed environment using the three methods considered in this work (LS, U-Net and cGANs), specifically, we evaluate these methods using PSNR and Structural Similarity Index Measure (SSIM) [20] as quality metrics. They provide a general evaluation of how similar the reconstruction is to the original floor plan. However, this evaluation is performed based on the whole image. Hence, we also calculate the distance (in centimeters) between the centres of the scatters in the original and predicted environment. This further provides us with a way to assess local spatial predictions.

Let us first analyse the general reconstructions capability of each of these methods. Figure E.3 compares the average PSNR obtained on the test set by the three methods considering different S-averaging of the received signal. As expected, the original signal $S = 1$ provides little information and hence makes it unfeasible to obtain good quality reconstruction (perfect reconstruction would lead to a maximum PSNR of 48 dB). Moreover, we notice that specifically

	Method	PSNR	SSIM	Error distance (cm)		
				$K_a = 5$	$K_a = 20$	Average
Scenario 1	LS	12.71 ± 1.29	0.19 ± 0.03	-	-	-
	U-Net	21.51 ± 0.24	0.93	5.68 ± 5.67	3.51 ± 3.50	4.59 ± 4.59
	cGANs	**31.77 ± 0.3**	**0.95**	**4.58 ± 4.50**	**1.96 ± 1.95**	**3.27 ± 3.27**
Scenario 2	LS	11.896 ± 1.09	0.232 ± 0.04	-	-	-
	U-Net	21.51 ± 0.17	0.94	**5.27**	**3.64 ± 8.02**	**4.45 ± 4.01**
	cGANs	**32.66 ± 0.24**	**0.95**	$7.99 + \pm 8.9$	7.89	7.94 ± 4.45

Table E.1: Average qualitative comparison of LS, U-Net and cGANs in terms of PSNR, SSIM and scattering centroids error distance.

for the linear mapping (LS), the quality of the reconstruction drastically decays as S-averaging increases. This happens despite the post-processing mechanisms performed (see Sec. 3) and the quality of the data used for training - we also considered training different models for specific scenarios, i.e., S-averaging and number of active transmitters, but obtained similar results. Consequently, it is reasonable to assume that it is extremely hard to find a direct linear mapping from the input radio map to the desired output. This further motivates the application of machine learning to this problem. In fact, for U-Net and cGANs, we notice the opposite behaviour, i.e., the larger S, the larger the PSNR obtained. In all scenarios the best results are obtained using cGANs. The overall generalisation of the methods is compared by considering the average reconstruction results over all different signal configurations $S \in [1, 100, 1000]$, $K_a \in [5, 20]$ and for the two different scenarios. Table E.1 contains the PSNR, SSIM and average distance error of the predicted scattered objects[4]. The best results are highlighted. Indeed, the average quantity of these metrics is consistent with what is explained above. More interesting, though, is the error distance (in centimetres) between the central position of the scattered objects and their reconstruction. For the original floor plan, these centres are the middle point of each rectangle. In the reconstructed images, we obtain these central positions by approximating polygons to the scatter. The cGANs exhibit the best performance in most of the comparative analyses. It obtains the highest PSNRs and SSIM in both scenarios and the smallest average distance for *Scenario 1*. Specifically, for *Scenario 2*, U-Net obtains smaller error distances among the true and the predicted position of the scatters. However, in real-world space, this difference is almost negligible. An exemplary visual reconstruction for $S = 100$ is provided in Figure E.2(e)-(h) for both U-Net and cGANs. Finally, since LS provides little information (see Figure E.2(c)-(d) for a visual comparison) on the reconstructed scenario, it is impossible to perform any analysis on the error distance between the true location of the scatters and their predicted locations.

---

[4] We do not denote the variance of the results when it is negligible.

6 Conclusion

We have demonstrated a proof-of-concept that it is possible to learn an environmental scene reconstruction using signals received at a LIS while it is performing communication. This sensing procedure can be done in parallel and without interfering with the communication task. We have shown that using both U-Net and cGANs it is possible to accurately (with less than 16 cm error) estimate the central position of the scatters present in the environment directly from these image reconstructions. In future work, we plan to explore real-world environments as well as explore the learning environment to enhance communication performance.

7 Acknowledgments

This work has been partially funded by the European Commission under the Windmill project (contract 813999) and the Spanish government under the Aristides project (RTI2018-099722-B-I00).

References

[1] H. Surmann, A. Nüchter, and J. Hertzberg, "An autonomous mobile robot with a 3D laser range finder for 3D exploration and digitalization of indoor environments," *Robotics and Autonomous Systems*, vol. 45, no. 3-4, pp. 181–198, 2003.

[2] Y. Zhang, C. Luo, and J. Liu, "Walk&Sketch: Create Floor Plans with an RGB-D Camera," in *Proceedings of the 2012 ACM Conference on Ubiquitous Computing*, 2012, pp. 461–470.

[3] R. Gao et al., "Indoor Floor Plan Reconstruction via Mobile Crowdsensing," in *Proceedings of the 20th Annual International Conference on Mobile Computing and Networking*, 2014, pp. 249–260.

[4] B. Zhou et al., "BatMapper: Acoustic Sensing Based Indoor Floor Plan Construction Using Smartphones," in *Proceedings of the 15th Annual International Conference on Mobile Systems, Applications, and Services*, 2017, pp. 42–55.

[5] K. S. Chong and L. Kleeman, "Feature-based Mapping in Real, Large Scale Environments using an Ultrasonic Array," *The International Journal of Robotics Research*, vol. 18, no. 1, pp. 3–19, 1999.

[6] A. D. Singh et al., "RadHAR: Human Activity Recognition from Point Clouds Generated through a Millimeter-wave Radar," in *Proceedings of the 3rd ACM Workshop on Millimeter-wave Networks and Sensing Systems*, 2019, pp. 51–56.

[7] C. X. Lu et al., "See Through Smoke: Robust Indoor Mapping with Low-cost mmWave Radar," in *Proceedings of the 18th International Conference on Mobile Systems, Applications, and Services*, 2020, pp. 14–27.

[8] T. Weiss et al., "Joint Optimization of System Design and Reconstruction in MIMO Radar Imaging," in *2021 IEEE 31st International Workshop on Machine Learning for Signal Processing (MLSP)*, 2021, pp. 1–6.

[9] R. J. Williams et al., "Multiuser MIMO with Large Intelligent Surfaces: Communication Model and Transmit Design," in *ICC 2021-IEEE International Conference on Communications*, 2021, pp. 1–6.

[10] C. J. Vaca-Rubio et al., "Assessing wireless sensing potential with large intelligent surfaces," *IEEE Open Journal of the Communications Society*, vol. 2, pp. 934–947, 2021.

[11] C. J. Vaca-Rubio, P. Ramirez-Espinosa, K. Kansanen, Z.-H. Tan, and E. de Carvalho, "Radio sensing with large intelligent surface for 6G," in *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2023, pp. 1–5.

[12] P. Isola et al., "Image-to-Image Translation with Conditional Adversarial Networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

[13] Z. Zhou et al., "Spherical wave channel and analysis for large linear array in LoS conditions," in *2015 IEEE Globecom Workshops (GC Wkshps)*, 2015, pp. 1–6.

[14] FEKO, Altair Engineering, Inc. [Online]. Available: `https://www.altairhyperworks.com/feko`

[15] N. Vaswani, "LS-CS-Residual (LS-CS): Compressive Sensing on Least Squares Residual," *IEEE Transactions on Signal Processing*, vol. 58, no. 8, 2010.

[16] M.-S. Kang and K.-T. Kim, "Compressive sensing based SAR imaging and autofocus using improved Tikhonov regularization," *IEEE Sensors Journal*, vol. 19, no. 14, pp. 5529–5540, 2019.

[17] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional Networks for Biomedical Image Segmentation," in *International Conference on Medical Image Computing and Computer-Assisted Intervention*, 2015, pp. 234–241.

[18] M. Sandler et al., "MobileNetV2: Inverted Residuals and Linear Bottlenecks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 4510–4520.

[19] T. Wang et al., "High-resolution Image Synthesis and Semantic Manipulation with Conditional GANs," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 8798–8807.

[20] A. Hore and D. Ziou, "Image Quality Metrics: PSNR vs. SSIM," in *2010 20th International Conference on Pattern Recognition*, 2010, pp. 2366–2369.

Paper F

OnRMap: An online radio mapping approach for large intelligent surfaces

Herman L Santos, Victor Croisfelt, Cristian J Vaca-Rubio, Taufik Abrão,
Petar Popovski

The paper has been accepted in the
IEEE International Conference on Communications (ICC), 2023.

© 2023 IEEE

The layout has been revised.

Abstract

We introduce OnRMap, an online radio mapping (RMap) approach for the sensing and localization of active users (AUs), devices that are transmitting radio signals, and passive elements (PEs), elements that are in the environment and are illuminated by the AUs' radio signals. OnRMap processes the signals received by a large intelligent surface and produces a radio map (RM) of the environment based on signal processing techniques. The method then senses and locates the different elements without the need for offline scanning phases, which is important for environments with spatial layouts that change frequently. Empirical results demonstrate that OnRMap presents a higher localization accuracy than an offline method, but the price paid for being an online method is a moderate reduction in the detection rate.

Keywords— Large intelligent surface (LIS), sensing, localization, radio mapping (RM).

1 Introduction

LIS is an important concept on the evolution path of wireless multi-antenna systems. It originally refers to a continuous electromagnetic surface able to transmit and receive radio waves [1]. In practice, a LIS is envisioned as a collection of closely-spaced antenna elements deployed across a large 2D surface, which can be easily integrated into the propagation environment, *e.g.*, placed on walls or ceilings. In addition to its well-investigated communication capabilities [1, 2], LIS also holds potential for radio sensing and localization [3], with applications in self-driving vehicles, unmanned aerial vehicles, or autonomous robots. These use cases normally require the construction of radio maps (RMs), whose process can exploit the RF signals emitted by the wireless devices in the environment of interest, termed active users (AUs). Moreover, static and/or dynamics objects that are not transmitting RF signals, termed passive elements (PEs), can also be detected/sensed and located by exploiting the multipath components of the RF signals transmitted by the AUs. We refer to radio mapping (RMap) as the process of obtaining RMs, which is the main subject of this paper.

In [4, 5], the authors used a LIS to obtain RMs, treating the received signals at the LIS as a digital image and creating an RMap method based on techniques from digital image processing and computer vision. Despite the good detection performance of AUs and PEs, digital image processing requires offline processing, which is not suitable for environments with frequently-changing spatial layouts. Other previous RMap methods rely on the discretization of outdoor [6] and indoor [7] static environments, evaluating the path between AUs and PEs in a pixel-like manner. The critical problem with pixel-based approaches is that good detection performance requires increased pixel granularity, resulting in exponential complexity. In contrast, the authors of [8, 9] used the related concept of radio tomographic imaging (RTI) that allowed them to obtain an RM of moving PEs (humans) by imaging their attenuation in a wireless sensor network (WSN) comprised of fixed-located sensors in a square area. However, this demands dedicated sensors, making it more expensive than methods that exploit widespread wireless AUs.

This work proposes *OnRMap*, an online RMap approach based on classical signal processing techniques. In contrast to [4, 5], OnRMap eliminates the need for offline scanning phases, being more robust to dynamic environments at the cost of a reasonably lower detection rate in

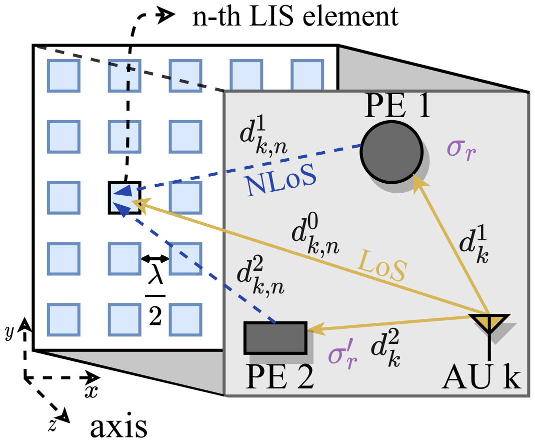


Fig. F.1: Indoor communication scenario with a LIS installed on the ceiling. The LOS (AU-to-LIS) and non-line-of-sight (NLoS) (AU-to-PEs-to-LIS) rays from a single AU are depicted. PEs can be different objects characterized by different parameters σ_r.

comparison to [4]. The numerical results indicate that OnRMap provides a higher localization accuracy of the PEs.

2 System Model

Consider the indoor communication system within an enclosed room and where a LIS is placed on the ceiling, as illustrated in Fig. F.1. The LIS covers the whole room beneath it and is composed of a uniform planar array (UPA) containing $N = N_x \cdot N_y$ antenna elements equally spaced by $\lambda/2$, where λ is the carrier wavelength. Within the room, there are R PEs and K AUs.

2.1 Channel Model

Assume that when the K AUs transmit uplink (UL) signals, those impinge at the LIS either as a result of LOS and/or as NLoS propagation. The latter occurs due to reflections of the transmitted signals on the PEs present in the room. For mathematical tractability, we ignore the NLoS components resulting from reflections on the wall and floor in the formulation; besides, we assume one reflection per PE. Let $\mathbf{h}_k \in \mathbb{C}^{N\times 1}$ denote the channel vector of the k-th AU to the LIS. Considering an indoor scenario, we use the spherical wavefront assumption and adopt the Saleh-Valenzuela model [10] with one ray per PE, where the n-th entry of $\mathbf{h}_k$ is

$$h_{k,n} = \underbrace{\beta^0_{k,n} e^{-j\frac{2\pi}{\lambda} d_{k,n}}}_{\text{LoS}} + \underbrace{\sum_{r=1}^{R} \beta^r_{k,n} e^{-j\frac{2\pi}{\lambda} d^r_{k,n}}}_{\text{NLoS}}, \tag{F.1}$$

where superscript $[\cdot]^r$ denotes the r-th multipath component from a total of $R+1$ components; specially, $\beta^0_{k,n} \in \mathbb{R}^+$ denotes the LOS component. The Euclidean distance from the AU to the n-th LIS antenna is denoted by $d^0_{k,n}$, while $d^r_{k,n}$ denotes the distance of the n-th LIS' element to the r-th PE. The channel gain of the r-th multipath component can be modeled as [11]:

$$\beta^r_{k,n} = \frac{\lambda}{4\pi} \frac{\sqrt{\sigma_r} e^{-j\Delta\phi_{r,n}}}{(d^r_k + d^r_{k,n})} \tag{F.2}$$

with d^r_k representing the distance between the k-th AU and the r-th PE, σ_r denoting the reflection loss by the r-th PE, and $\Delta\phi_{r,n} = 2\pi(d^r_k + d^r_{k,n} - d^0_{k,n})/\lambda$ is the phase difference (Doppler shift) between the LOS and the r-th NLoS components. The parameter σ_r models a random variable subject to conductivity, relative permittivity, and permeability of the PEs, whose value can differ for different types of PEs.

3 Radio Mapping: An Overview

Suppose the channel estimation (CHEST) phase of a LIS system. The K AUs transmit using K orthogonal pilots such that the received signal $\mathbf{Y} \in \mathbb{C}^{N \times K}$ at the LIS is:

$$\mathbf{Y} = p\mathbf{H} + \mathbf{W}, \tag{F.3}$$

where p is the UL transmit power, which is assumed to be equal to all the AUs, $\mathbf{H} = [\mathbf{h}_1, \mathbf{h}_2, \ldots, \mathbf{h}_K] \in \mathbb{C}^{N \times K}$ is the channel matrix, and $\mathbf{W} \in \mathbb{C}^{N \times K}$ is the receiver noise matrix whose entries are independent and identically distributed (i.i.d.) according to $\mathcal{CN}(0, \sigma^2_w)$. To exploit $\mathbf{Y}$ for RMap, a spatial MF $\hat{\mathbf{h}} \in \mathbb{C}^{N_f \times 1}$ was proposed in [4, 5], whose filter coefficients are computed as:

$$\hat{h}_n = t_n e^{-j\frac{2\pi}{\lambda} d_n}, \quad \forall n \in \{1, \ldots, N_f\}, \tag{F.4}$$

where t_n is the n-th element of a weighting vector $\mathbf{t} \in \mathbb{R_+}^{N_f \times 1}$, $e^{-j\frac{2\pi}{\lambda} d_n}$ defines a spherical wave phase-matching component, adjusted to the Euclidean distance $d_n = (d^2_{x,n} + d^2_{y,n} + d^2_{z,n})^{1/2}$ between a reference (x, y, z) point in space and the n-th LIS antenna, and $N_f \leq N$ is the number of taps of the filter. Let denote the k-th column of $\mathbf{Y}$ in Eq. (F.3) as $\mathbf{Y}_k \in \mathbb{C}^{N_x \times N_y}$ with $\mathbf{y}_k = (\mathbf{Y}_k)$. We also let $\underline{\hat{\mathbf{h}}} = [\hat{\mathbf{h}}, \mathbf{0}] \in \mathbb{C}^{N \times 1}$ be the MF after zero-padding according to $\max(N_f, N)$. Then, the contribution $\boldsymbol{\Upsilon}_k \in \mathbb{C}^{N_x \times N_y}$ to the primary RM from the k-th AU is obtained as follows:

$$\boldsymbol{\Upsilon}_k = \underline{\hat{\mathbf{H}}} \otimes \mathbf{Y}_k, \tag{F.5}$$

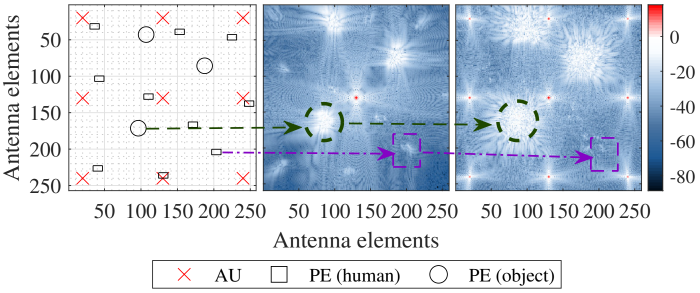


Fig. F.2: RMs showing one drawback of the MF used in [4] in the indoor scenario specified in Appendix 8 with $K = 9$ AUs and $R = 13$ PEs, where $R_{\text{obj}} = 3$ are metallic objects and $R_{\text{hum}} = 10$ are humans. *Left*: ground-truth. *Center*: single-AU $K = 1$; *Right*: superposed $K = 9$ AUs. When the signals are superposed, it is noticeable that PEs with higher reflection loss (humans w/ violet line) get occluded, while the others (objects w/ green line) got enhanced.

where $\underline{\hat{\mathbf{h}}} = (\underline{\hat{\mathbf{H}}})$. The primary RM $\boldsymbol{\Upsilon} \in \mathbb{C}^{N_x \times N_y}$ is [4]:

$$\boldsymbol{\Upsilon} = \left(\sum_{k=1}^{K} \mathbf{Y}_k \right) \otimes \underline{\hat{\mathbf{H}}}. \tag{F.6}$$

When designing the filter, the authors of [4] considered the components $d_{x,n}$ and $d_{y,n}$ of d_n to be the distances of each n-th LIS element to the center of the room, while the focal height $d_{z,n}$ is a parameter subject to design. Further processing can be cast over $\boldsymbol{\Upsilon}$ to perform sensing and localization.

We highlight two main drawbacks in the pre-processing from [4]. First, by using the combination in Eq. (F.6), the orthogonality among the signals from different AUs is lost. Thisnon-orthogonal superposition may incur loss of relevant information, as the PEs are being irradiated from different angles. This can produce different Dopplers' shifts, as indicated by Eq. (F.2), which can in turn mitigate the contribution of the NLoS components of interest. In other words, the contribution of the PEs can be *overshadowed* in the RMs. Second, the filter structure is capable of matching the phase of the LOS components, but outputs some *distortion*, especially in the neighborhood where the LOS ray impinges.

To illustrate the drawbacks, we consider the scenario described in Appendix 8, where we have $K = 9$ AUs and $R = 13$ PEs with $R_{\text{obj}} = 3$ of them being cylindrical metallic objects and $R_{\text{hum}} = 10$ humans. Fig. F.2 contains (a) the ground truth RM, where the AUs are represented by red crosses, while the PEs by the geometrical forms – metallic objects by the circles and the humans by the rectangles – (b) the RM of a single AU; and (c) the RM of the superposed signals from the $K = 9$ AUs. The MF filter used was the same as in [4], with $t_n = 1/d_n$ computed as in (F.4). Fig. F.2(b) exposes the effect of the distortion around the LOS components with power as high as the rays reflected by PEs.From Fig. F.2(c), we see how the metallic objects are highlighted while the humans get occluded, making it more difficult to correctly detect the latter.

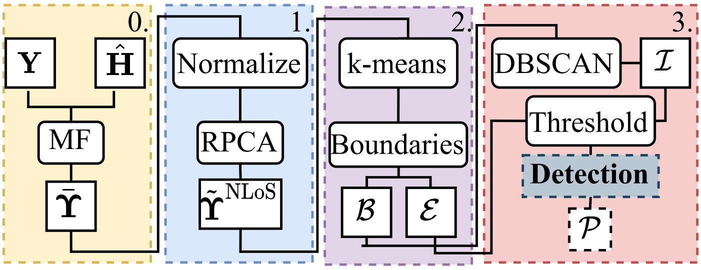


Fig. F.3: OnRMap architecture.

4 OnRMap

Here we present OnRMap, an online RMap approach based on classical signal processing theory, which removes the need for offline scanning phases from [4, 5]. We detail the four stages that comprise OnRMap until we sense and locate the AUs and PEs, with an emphasis on the PEs, which are the most challenging to be detected.

4.1 OnRMap: An Overview

OnRMap consists of four steps, illustrated in Fig. F.3.

Step 0. *Signal Acquisition and Primary RM.* In this initial step, we obtain the contributions $\boldsymbol{\Upsilon}_k$ from the AUs $\forall k \in \{1, 2, \dots, K\}$ to the primary RM by filtering the signals $\mathbf{Y}_k$ with the MF given in Eq. (F.5). Different from [4] and based on the discussion made in Section 3, we empirically chose the weight vector $\mathbf{t}$ using a two dimensional (2-D) Taylor window [12]. This selection was motivated by the empirical reduction of filtering distortion and numerical improvement in the ratio between the NLoS components. The output of this step is the matrix $\bar{\boldsymbol{\Upsilon}} \in \mathbb{R}^{N \times K}$ containing the K primary RMs stacked in columns, *i.e.*, $\bar{\boldsymbol{\Upsilon}} = [|(\boldsymbol{\Upsilon}_1)|, |(\boldsymbol{\Upsilon}_2)|, \dots \, |(\boldsymbol{\Upsilon}_K)|]$, where $|\cdot|$ here denotes element-wise absolute value.

Step 1. *Estimation of the LOS and NLoS Components.* Through this step, the primary RM in $\bar{\boldsymbol{\Upsilon}}$ is the input to the robust principal component analysis (RPCA) algorithm [13], which outputs the matrices $\hat{\bar{\boldsymbol{\Upsilon}}}^{\text{LoS}}$ and $\hat{\bar{\boldsymbol{\Upsilon}}}^{\text{NLoS}}$ corresponding to the estimations of the LOS and NLoS components of $\bar{\boldsymbol{\Upsilon}}$, respectively. We employ this method based on the observation that the NLoS components have similarities among themselves, *e.g.*, the range of the power gain and location. In contrast, LOS components are a few data points with high power gains that are normally far apart. Hence, we can interpret the NLoS components to be low-rank components of $\bar{\boldsymbol{\Upsilon}}$, whereas the LOS elements are sparse. Thus, RPCA becomes useful, since it is a low-complexity method for estimating the low-rank components of a matrix. Based on the special focus on sensing and locating the PEs, this block outputs the low-rank estimation, $\tilde{\boldsymbol{\Upsilon}}^{\text{NLoS}} \in \mathbb{R}^{N_y \times N_x}$.

Step 2. *Separation of the NLoS Components.* This step translates the NLoS estimation from the previous step into data points for the inference step. To do so, we input $\boldsymbol{(}\tilde{\boldsymbol{\Upsilon}}^{\text{NLoS}})$ in k-means clustering [14], which separates the data in two clusters that represent *foreground* (high power) and *background* (low power) classes differentiated by their power levels. The foreground forms shapes in the 2-D space and their perimeters are estimated with the Moore-Neighbor boundaries estimation algorithm [15]. The points that enclose the perimeters are stored in subsets that compound the set $\mathcal{B}$ and the total power levels each shape comprises in $\tilde{\mathbf{\Upsilon}}^{\text{NLoS}}$ are stored in the set $\mathcal{E}$, constituting the output of this step.

Step 3. *Sensing and Localization Inference.* In this final block, we use the data in $\mathcal{B}$ and $\mathcal{E}$ to infer whether each subset in $\mathcal{B}$ is a type of PE or just noise. If it is a PE, we can further identify which type of object it is, *e.g.*, a human or a metallic object, and their positions are also inferred. To do so, we employ Density-based spatial clustering of applications with noise (DBSCAN) clustering method [16] to cluster the data in $\mathcal{B}$ based on the distance between samples. It is also assigned to each cluster its total power by looking at the set $\mathcal{E}$. The classification of each cluster on which type of PE (metallic object or human) follows a decision rule based on the power each cluster has, that is, lower and upper bounds are defined, and the clusters that fit in between are considered humans, those above are considered as objects, while the rest is noise.

4.2 OnRMap: Detailed Description

Below, we give further details on the other steps apart from Step 0, which was already detailed in (F.5).

Estimation of the LoS and NLoS Components

RPCA solves the following optimization problem [13]:

$$\begin{aligned} \min \quad & \|\mathbf{L}\|_* + \lambda_{\text{RPCA}}\|\mathbf{S}\|_1 \\ \text{s.t.} \quad & \mathbf{L} + \mathbf{S} = \mathbf{M}, \end{aligned} \tag{F.7}$$

where $\mathbf{M}$ is the observation matrix, $\mathbf{L}$ and $\mathbf{S}$ are estimations of the low-rank and sparse components of $\mathbf{M}$, respectively, $\|\cdot\|_*$ and $\|\cdot\|_1$ are the nuclear and ℓ_1 norm of a matrix, respectively, and λ_{RPCA} is a scalar. In our case, from Step 0 we have the matrix $\bar{\mathbf{\Upsilon}} \in \mathbb{R}^{K\times N}$, which is obtained by stacking and normalizing the $(\mathbf{\Upsilon}_k)$, $\forall k \in \{1, 2, \dots, K\}$. Then, we decompose it as:

$$\bar{\mathbf{\Upsilon}} = \bar{\mathbf{\Upsilon}}^{\text{NLoS}} + \bar{\mathbf{\Upsilon}}^{\text{LoS}}. \tag{F.8}$$

Define $\hat{\bar{\mathbf{\Upsilon}}}^{\text{NLoS}}$ and $\hat{\bar{\mathbf{\Upsilon}}}^{\text{LoS}}$ as estimations of $\bar{\mathbf{\Upsilon}}^{\text{NLoS}}$ and $\bar{\mathbf{\Upsilon}}^{\text{LoS}}$, respectively. Then, substituting in Eq. (F.7), we have

$$\begin{aligned} \min \quad & \|\hat{\bar{\mathbf{\Upsilon}}}^{\text{NLoS}}\|_* + \lambda_{\text{RPCA}}\|\hat{\bar{\mathbf{\Upsilon}}}^{\text{LoS}}\|_1 \\ \text{s.t.} \quad & \hat{\bar{\mathbf{\Upsilon}}}^{\text{NLoS}} + \hat{\bar{\mathbf{\Upsilon}}}^{\text{LoS}} = \bar{\mathbf{\Upsilon}}, \end{aligned} \tag{F.9}$$

placing the separation of LOS and NLoS components as equivalent to the optimization problem in (F.7). The output of the algorithm $\hat{\bar{\mathbf{\Upsilon}}}^{\text{NLoS}} \in \mathbb{R}^{K\times N}$ is summed row-wise and reshaped to $\tilde{\mathbf{\Upsilon}}^{\text{NLoS}} \in \mathbb{R}^{N_y\times N_x}$; the same for the LOS-related matrix.

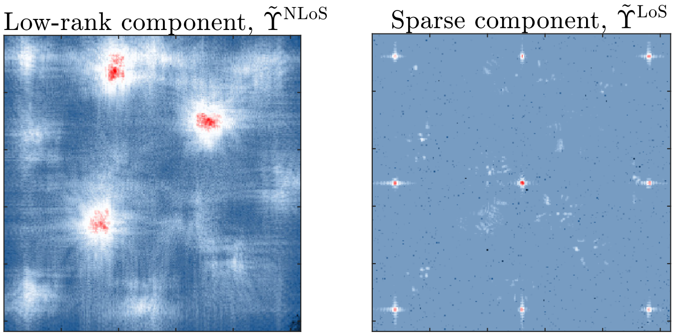

Low-rank component, $\tilde{\Upsilon}^{\mathrm{NLoS}}$ Sparse component, $\tilde{\Upsilon}^{\mathrm{LoS}}$

Fig. F.4: RPCA output. *Left:* low-rank matrix $\hat{\hat{\boldsymbol{\Upsilon}}}^{\mathrm{NLoS}}$ denoting the NLoS components. *Right*: sparse matrix $\hat{\hat{\boldsymbol{\Upsilon}}}^{\mathrm{LoS}}$ denoting the LOS components. We set the parameter $\lambda_{\mathrm{RPCA}} = 3N^{-1/2}$ based on [13] and experimentation.

To illustrate how well RPCA performs, Fig. F.4 shows the recovered low-rank (left) and sparse matrix (right). As can be seen, RPCA can estimate satisfactorily the NLoS (left) and LOS (right) components. On the other hand, we can point out two disadvantages of this technique. First, although there is a reference value for λ_{RPCA}, it may have to be tuned according to the scenario. Second, the problem of estimating low-rank and sparse components of matrices is NP-hard. RPCA solves a convex relaxed problem, incurring some information loss. However, most of the time, this method estimates satisfactory the components within less than twenty iterations. Despite that, for some realizations, part of the PEs was outputted in the sparse (LOS) component.

Separation of the NLoS Components

First, let's define the binary k-means clustering operation as $\mathfrak{K} : \mathbb{R}^{N\times 1} \mapsto \{0,1\}^{N\times 1}$. Then, we perform entry-wise $(\boldsymbol{\Upsilon}^{\mathrm{km}}) = \mathfrak{K}((\tilde{\boldsymbol{\Upsilon}}^{\mathrm{NLoS}}))$, with $\boldsymbol{\Upsilon}^{\mathrm{km}} \in \{0,1\}^{N_y\times N_x}$ being the matrix with the class of each data point in $\tilde{\boldsymbol{\Upsilon}}^{\mathrm{NLoS}}$. In particular, we name class 0 as *background* containing points with null to very low power and class 1 as *foreground* containing components above a certain power threshold. We illustrate the output of the k-means $\boldsymbol{\Upsilon}^{\mathrm{km}}$ on the left side of Fig. F.5. Note the creation of certain regions that can be interpreted as geometrical shapes. Visually inspecting and comparing the shapes with the ground truth in Fig. F.2(a), we can infer that components that represent the metallic objects are more likely to be clustered in a larger and denser area, while the humans' ones are smaller and sparser.

Based on the last observation, we employ the Moore-Neighbor boundaries estimation algorithm [15] to leverage these regions to detect the PEs. The boundary estimation algorithm works in the following way. First, a random point with value '1' (foreground) in $\boldsymbol{\Upsilon}^{\mathrm{km}}$ is picked up, it is then defined as a central point, and its eight-point neighborhood values are checked. Then, the neighbors that have value '1' to this central point are considered to be in the same region as the central point. The algorithm continues by looking for other points considering the neighborhood of the newly found points. Finally, a shape is defined when there are no other

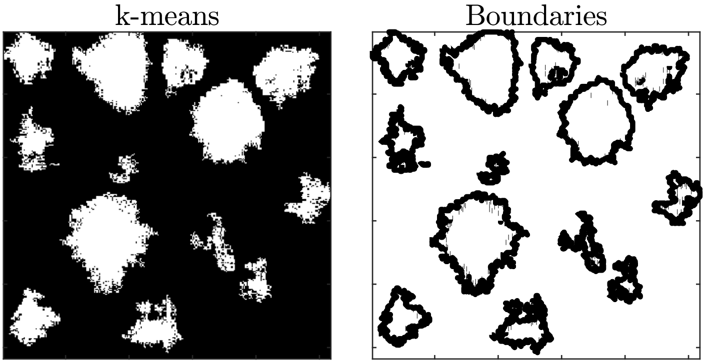


Fig. F.5: Graphic representation of *left:* the output of the binary k-means clustering, $\boldsymbol{\Upsilon}^{\mathrm{km}}$ and *right:* the estimated boundaries set $\mathcal{B}$ obtained via Moore-Neighbor boundaries estimation.

points left with the value '1' around the perimeter, where the outermost points are retrieved as a perimeter. Both perimeter and internal points are defined by (x, y) tuples representing the column and row in which they are allocated in the $\boldsymbol{\Upsilon}^{\mathrm{km}}$ matrix. The perimeter points are assigned to a subset $\mathcal{B}_s \subset \mathcal{B}$ and all the points that define a shape in $\mathcal{A}_s \subset \mathcal{A}$, where s indexes a shape. Then, we associate a value E_s to each region s depicting the total power level within that region, which is calculated as:

$$E_s = \sum_{i \in \mathcal{A}_s} \left(\tilde{\boldsymbol{\Upsilon}}^{\mathrm{NLoS}}[x_i, y_i] \right)^2 . \tag{F.10}$$

The values E_s's are further stored in the set $\mathcal{E}$. The boundaries for this scenario can be seen on the right side of Fig. F.5.

Sensing and Localization Inference

This block receives the sets $\mathcal{B}$ and $\mathcal{E}$ and does the inference process. First, the data is clustered with DBSCAN [16]. This method randomly chooses a core point by observing the data in $\mathcal{B}$ and maps the neighborhood subject to `minPts` and ϵ constraint parameters, representing the minimum number of points in a cluster and the search radius, respectively. Starting from the core point, the algorithm calculates the distances to all points in the data and assigns as a cluster those that satisfy the parameter constraints. The process is repeated until all the data have been clustered. Points that are sparsely distributed are considered noise. The indexes of the shapes that compound each cluster are assigned to the i-th cluster $\mathcal{I}_i \subset \mathcal{I}$ that maps both $\mathcal{B}$ and $\mathcal{E}$ sets.

We perform a test of the described algorithm on the illustrative scenario in Appendix 8 with `minPts` $= 2$ and $\epsilon = 2$. The algorithm outputs 63 identified clusters, which is much higher than the true number of PEs in the environment. To overcome an excessive number of clusters, we exploit the lower reflection loss of metal objects than humans, $|\sigma_r| \ll |\sigma_{r'}|$, so we can infer that clusters with high power are objects, while clusters with low powers are noise or

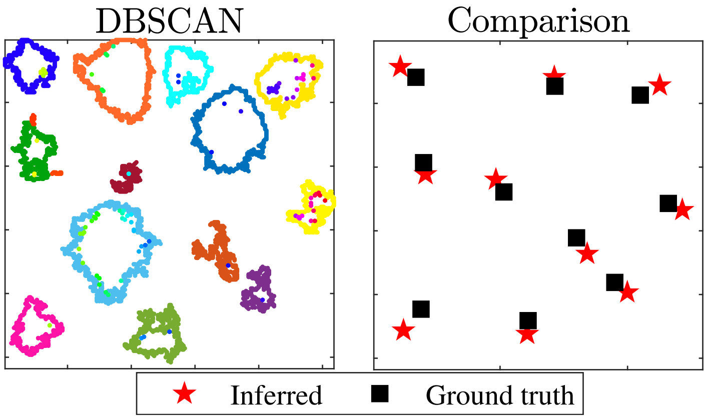


Fig. F.6: Inference step of OnRmap. *Left*: identified clusters by DBSCAN. The colors represent each cluster; the large ones are more likely to represent PEs, while the clusters with very few elements are likely noise. *Right*: inferred positions comparison against ground truth.

background distortion; consequently, clusters in the middle represent human positions. Thus, the i-th cluster is considered to be of a human class if it satisfies the following rule over $\mathcal{E}$:

$$\mathfrak{D}(i) = \begin{cases} 1, & \text{if} \quad \text{th}_{\text{min}} < \sum_{s\in\mathcal{I}_i} \mathcal{E}_s \leq \text{th}_{\text{max}}, \\ 0, & \text{otherwise}. \end{cases} \tag{F.11}$$

where th_{min} and th_{max} are the minimum and maximum threshold parameters, respectively. If the cluster passes the threshold rule, then the cluster centroid is stored in a new *set of detected humans* $\mathcal{H}$.

To evaluate how reasonable the inference is, we let $\mathbf{a} \in \mathbb{R}^{+R_{\text{hum}}\times 1}$ be the vector containing the differences between the inferred and the ground truth localization of the humans, giving the localization accuracy per human. Its r-th entry is calculated as:

$$a_r = \min\left(\left\{\|\mathcal{H}_j - \mathcal{H}_r^{\text{gt}}\| < d_{\text{th}}, j = 1, 2, \ldots, |\mathcal{H}|\right\}\right), \tag{F.12}$$

where $\mathcal{H}_j$ denotes the j-th element of $\mathcal{H}$ and $\mathcal{H}^{\text{gt}}$ is the set of ground-truth humans with the r-th element denoted as $\mathcal{H}_r^{\text{gt}}$ and whose cardinality is $|\mathcal{H}^{\text{gt}}| = R_{\text{hum}}$. Furthermore, $\|\cdot\|$ is the Euclidean norm. We adopt as $d_{\text{th}} = 1\text{m}$ the distance threshold for detection via experimentation. The value of the accuracy a_r can be *null* in case the distance is higher than the one defined by the threshold; in this case, we assume $a_r = 0$. We define the localization accuracy, LA, and the detection rate, DR, as:

$$\text{LA} = \frac{1}{R_{\text{hum}}} \sum_{i=1}^{R_{\text{hum}}} a_i, \text{ and } \text{DR} = \frac{1}{R_{\text{hum}}} \sum_{i=1}^{R_{\text{hum}}} \nVdash(a_i > 0), \tag{F.13}$$

where $\nVdash$ is the indicator function.

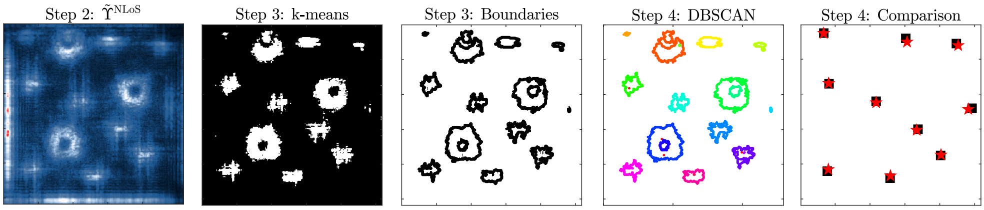


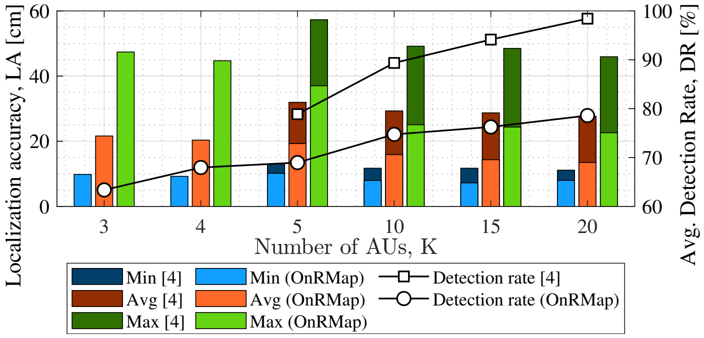
Fig. F.7: Exemplary application of OnRMap in the simulation environment from [4]. In Step 4: ⋆ is inferred position and ■ ground-truth.

By empirically setting set $\mathrm{th}_{\min} = 0.03\frac{1}{S}\sum\mathcal{E}$ and $\mathrm{th}_{\max} = 0.8\max(\mathcal{E})$, Fig. F.6 shows how the data were clustered for the illustrative example of Appendix 8. Overall, it was identified 67 clusters, where three have very high energy, corresponding to the metallic objects, ten with low to medium energy, corresponding to humans, and the rest with shallow energy, considered noise. To visually assess the quality of the results, we re-plot on the right side of the figure the ground truth points together with the inferred points. The algorithm was capable of inferring the human's positions with localization accuracy from 0.1m (best case) to 0.62m (worst case) and the average of 0.22m in this single realization. The overall complexity of the method, in terms of Big-$\mathcal{O}$ is upper bounded by $\mathcal{O}(N^2(N_f + K^2 + I_{\text{k}-\text{means}} + 2))$, where $I_{\text{k}-\text{means}}$ is the number of iterations to the convergence of k-means, which is typical 10 to 20.

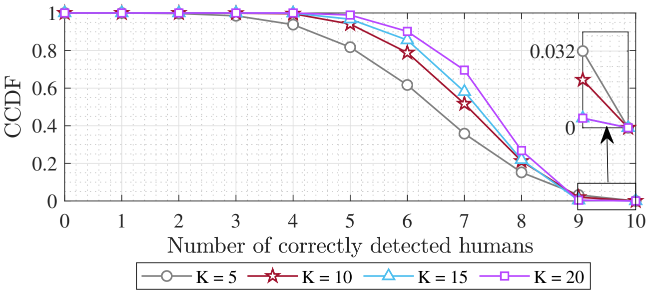
5 Numerical results

We now evaluate the effectiveness of OnRMap and compare it with the previous works [4, 5]. For a better evaluation, we consider a more complex indoor scenario as provided by Feko by Altair Engineering ray tracing, as used in [4]. *Different from the system model and toy example presented, this scenario includes more reflections from the PEs and reflections from the ground and walls.* Despite that, the other parameters are the same as in Appendix 8 with $R_{\mathrm{obj}} = 3$ metallic objects and $R_{\mathrm{hum}} = 10$ humans. *Throughout this section, we focus on showing results for human detection since the detection of AUs and metallic objects is less challenging.*

We start by illustrating a step-by-step realization of OnRMap in Fig. F.7 with $K = 20$ AUs. From the figure, we can note that OnRMap can perform well even in this more unfavorable scenario, given the increase in multipath components. In Fig. F.8, we better evaluate the performance of OnRMap by considering 1000 Monte Carlo simulations (MCSs). Fig. F.8a shows the average localization accuracy and the average detection rate of the humans when considering different numbers of AUs. We compare these results with the ones presented in [4]. The trade-off in the comparison is that we can achieve higher localization accuracy for all numbers of AU configurations (K) at the cost of lower detection rates on average. It is possible to point out two causes of the lower detection performance compared to [4]. First, some information of the signals is filtered after RPCA and does not enter in the inference process, as opposed to [4] that treats the signals from all AUs. Second, when the boundaries among PEs from different classes are too close, DBSCAN classifies them as one cluster and the decision



(a) Average localization accuracy and average detection rate, defined in (F.13), for different numbers of AUs.



(b) CCDF of the number of correctly detected humans for different numbers of AUs.

Fig. F.8: Performance evaluation for the sensing (detection) and localization of $R_{\text{hum}} = 10$ humans (PEs).

rule considers both as just one type of PE. These are considered the costs of the proposed online method in view of the offline approach. The lack of *a priori* information to compensate static PEs as in [4], compromises the DR to some extent. However, we argue that the high applicability and the significant improvement in the overall LA justify the novelty of this work.

To better understand the impact of the number of AUs in the detection rate, Fig. F.8b shows the complementary cumulative distribution function (CCDF) of the detection in terms of the number of correctly detected humans for different numbers of AUs K. Note that the effect of increasing K is to lower the variance on how many humans R_{hum} we can detect. However, the opposite occurs at the points of nine to ten correctly detected humans, highlighted in Fig. F.8b. For $K = 20$, the probability of a perfect detection (ten humans) is 0.4%, and when $K = 5$ it achieves 3.2%. A reason for this is that higher K implies more signals reflecting on PEs that can be possibly distributed next to the center of the room, while the PEs next to the walls are less exposed, and those reflections start to be interpreted as *background* (distortion/noise) by the method.

6 Conclusions

In this paper, we proposed OnRMap, an online method for sensing and localization in indoor environments equipped with LIS systems. The greatest advantages of OnRMap are due to the fact that it is based on signal processing techniques, making it an online method, that is, it does not rely on offline scanning phases. This makes the method more robust for applications where the environment is constantly changing. However, the online feature comes with the cost of an average lower detection rate. But even so, OnRMap turns out to have a fairly good location accuracy. Future works may improve the design of OnRMap to cover the observed weakness and better study its performance. Moreover, the study of methods that overcome the major drawback of having the LIS covering the whole room.

7 Acknowledgments

This work was supported in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) – Finance Code 001 and by the National Council for Scientific and Technological Development (CNPq) of Brazil under Grants 405301/2021-9, 141485/2020-5, and 310681/2019-7. Victor Croisfelt and Petar Popovski were supported by the Villum Investigator Grant “WATER” from the Velux Foundation, Denmark, and partly by the H2020 RISE-6G project financed by the European Commission under grant no. 101017011. Cristian J. Vaca-Rubio was supported by the European Union’s Horizon EUROPE research and innovation program under grant agreement No. 101037090 - project CENTRIC.

8 Appendix: An Illustrative Indoor Scenario

The parameters of the simulation scenario are summarized in Table F.1. Note that we consider a scenario with two types of PEs: i) $R_{\text{obj}} = 3$ cylindrical *metal objects* with polished surfaces $\sigma_r \sim \mathcal{U}[-10, -15]$ dB and $R_{\text{hum}} = 10$ *humans* $\sigma_{r'} \sim \mathcal{U}[-30, -75]$ dB, totaling $R = 13$ PEs.

References

[1] S. Hu et al., "Beyond Massive MIMO: The Potential of Data Transmission With Large Intelligent Surfaces," *IEEE Transactions on Signal Processing*, vol. 66, no. 10, pp. 2746–2758, 2018.

[2] D. Dardari, "Communicating With Large Intelligent Surfaces: Fundamental Limits and Models," *IEEE Journal on Selected Areas in Communications*, vol. 38, no. 11, pp. 2526–2537, 2020.

[3] C. J. Vaca-Rubio et al., "Assessing wireless sensing potential with large intelligent surfaces," *IEEE Open Journal of the Communications Society*, vol. 2, pp. 934–947, 2021.

[4] C. J. Vaca-Rubio et al., "Radio Sensing with Large Intelligent Surface for 6G," *arXiv*, 2021.

Table F.1: Simulation parameters

Parameter	Value	Parameter	Value
Environment		***Channel & System***	
# PEs	$R_{\text{obj}} = 3$	Trans. power	$p = 20$ [dBm]
	$R_{\text{hum}} = 10$	Noise power	$\sigma_w^2 = -97$ [dBm]
Room dim.	(y) = 10.36 [m]	LoS PL	$\beta^0_{k,n} = \frac{\lambda}{4\pi d^1_{k,n}}$
	(x) = 10.36 [m]	NLoS ref. loss	$\sigma_r \sim \mathcal{U}[-10, -15]$ dB
	(z) = 8 [m]		$\sigma_{r'} \sim \mathcal{U}[-30, -75]$ dB
Obj. dim	rad. = 0.43 [m]	Carr. freq.	$f = 3.75$ [GHz]
	(z) = 2 [m]	Carr. wavel.	$\lambda = 0.08$ [m]
Hum. dim.	(y) = 0.3 [m]	***Active users***	
	(x) = 0.5 [m]	# AUs	$K = 9$
	(z) = 1.7 [m]	AUs height	$z_k = 1.8$ [m]
LIS pos.	$z = 8$ [m]		
LIS elem.	$N_x = N_y = 259$		

[5] C. J. Vaca-Rubio et al., "Floor Map Reconstruction Through Radio Sensing and Learning By a Large Intelligent Surface," in *2022 IEEE 32nd International Workshop on Machine Learning for Signal Processing (MLSP)*, 2022, pp. 1–6.

[6] X. Tong et al., "Environment Sensing Considering the Occlusion Effect: A Multi-View Approach," *IEEE Transactions on Signal Processing*, vol. 70, pp. 3598–3615, 2022.

[7] X. Tong et al., "Joint Multi-User Communication and Sensing Exploiting Both Signal and Environment Sparsity," *IEEE Journal of Selected Topics in Signal Processing*, vol. 15, no. 6, pp. 1409–1422, 2021.

[8] J. Wilson and N. Patwari, "Radio tomographic imaging with wireless networks," *IEEE Transactions on Mobile Computing*, vol. 9, no. 5, pp. 621–632, 2010.

[9] M. Zhao et al., "Through-wall human pose estimation using radio signals," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 7356–7365.

[10] A. A. M. Saleh and R. Valenzuela, "A Statistical Model for Indoor Multipath Propagation," *IEEE Journal on Selected Areas in Communications*, vol. 5, no. 2, pp. 128–137, 1987.

[11] A. Goldsmith, *Wireless Communications*, Cambridge University Press, 2005.

[12] E. Brookner, *Practical Phased Array Antenna Systems*, Artech House, 1991.

[13] E. J. Candès et al., "Robust Principal Component Analysis?," *J. ACM*, vol. 58, no. 3, 2011.

[14] D. Arthur and S. Vassilvitskii, "k-means++: The Advantages of Careful Seeding," in *SODA '07: Proceedings of the Eighteenth Annual ACM-SIAM Symposium on Discrete Algorithms*, 2007, pp. 1027–1035.

[15] R. C. Gonzalez et al., *Digital Image Processing Using MATLAB*, Pearson Prentice Hall, 2004.

[16] M. Ester et al., "A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise," in *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*, 1996, pp. 226–231.

Paper G

mmWave Wi-Fi trajectory estimation with continuous-time neural dynamic learning

Cristian J Vaca-Rubio, Pu Wang, Toshiaki Koike-Akino, Ye Wang, Petros Boufounos, Petar Popovski

The paper has been published in the
EEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)
pp. 1-5, 2023.

© 2023 IEEE

The layout has been revised.

Abstract

We leverage standards-compliant beam training measurements from commercial-of-the-shelf (COTS) 802.11*ad/ay devices for localization of a moving object. Two technical challenges need to be addressed: (1) the beam training measurements are intermittent due to beam scanning overhead control and contention-based channel-time allocation, and (2) how to exploit underlying object dynamics to assist the localization. To this end, we formulate the trajectory estimation as a sequence regression problem. We propose a dual-decoder neural dynamic learning framework to simultaneously reconstruct Wi-Fi beam training measurements at irregular time instances and learn the unknown dynamics over the latent space in a continuous-time fashion by enforcing strong supervision at both the coordinate and measurement levels. The proposed method was evaluated on an in-house mmWave Wi-Fi dataset and compared with a range of baseline methods, including traditional machine learning methods and recurrent neural networks.*

1 Introduction

Wi-Fi fingerprinting is one of the popular approaches for indoor localization, driven by open firmware releases, WLAN domain knowledge for waveform calibration and preprocessing, as well as recent advances in deep learning-based feature extraction [1–3].

With commercial-of-the-shelf (COTS) Wi-Fi devices, one can fingerprint the following Wi-Fi measurements:

- coarse-grained received signal strength indicator (RSSI) [4];
- mid-grained beam training measurements at 60 GHz [5–11]
- fine-grained channel state information (CSI) at sub-7 GHz [12–15];

Refer to [10, Section II] for detailed discussions on all three types of Wi-Fi channel measurements. Traditional machine learning and advanced deep learning methods have been applied to all Wi-Fi fingerprinted measurements [16–21]. For instance, DeepFi exploits 90 CSI amplitudes from all the subcarriers at three antennas for feature extraction using an autoencoder architecture [12, 22]. More recently, a pretrained fusion network between the CSI at sub-7 GHz and the beam training measurements at 60 GHz was proposed for both localization and device-free sensing tasks [10]. Nevertheless, the majority of these approaches are *frame-based*; that is, the coordinate is inferred from the current Wi-Fi frame, without integration of past measurements or previous trajectory history.

On the other hand, *sequence-based* approaches take consecutive Wi-Fi frames as the input, and state estimation (e.g, Kalman filter-like approaches [23, 24]) and recurrent neural networks (e.g., GRU and LSTM [25]) can be applied for trajectory estimation with the RSSI [21] and CSI [26] at sub-7 GHz. However, the sequence-based formulation has NOT been applied to mmWave Wi-Fi localization due to the intermittent nature of the mid-grained beam measurement:

1) **Low beam training rate**: In Fig. G.1, during the beacon header interval (BHI), mmWave Wi-Fi of 802.11ad/ay uses directional beacons for sector level sweep (SLS) to train both initiator/responder beampatterns for subsequent data transmission. This mandatory beam training results in significant overhead to the Wi-Fi network and it is desired to limit the

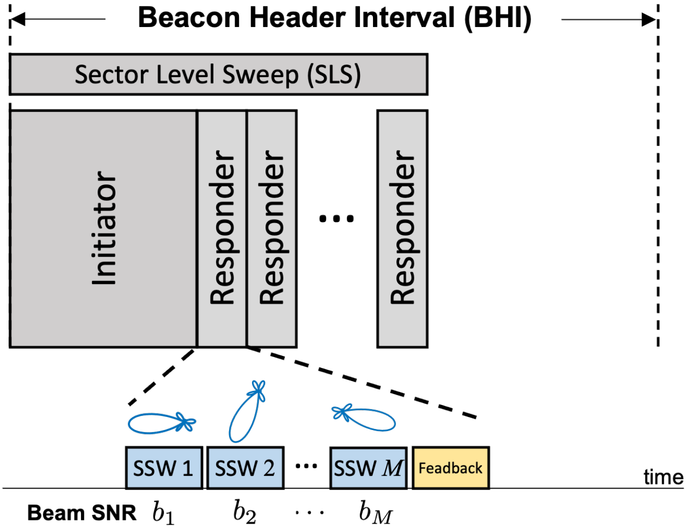


Fig. G.1: Beam training measurements (Beam SNRs) during the mandatory Sector Level Sweep (SLS) in 802.11ad/ay standards.

number of directional beampatterns within a beacon and the total number of beacons, resulting in sparsely sampled beam measurements than Wi-Fi at sub-7 GHz.

2) **Irregular sample intervals**: Consider a scenario where the access point (AP) is the initiator and the users are responders. When the responder trains its (transmitting or receiving) beampatterns, a sequence of M sector sweep (SSW) frames is sent via different beampatterns to the initiator and the initiator can compute M corresponding SNRs, $b_1, b_2, \cdots, b_M$, within a responder channel time. When multiple users exist, each user needs to contend the next responder channel time and one contending user is randomly selected. As a result, the contention-based channel access results in irregularly beam SNR measurements at AP for a given user.

To address the above challenges and inspired by recent advances in neural ordinary differential equation (NODE) [27], this paper proposes a dual-decoder neural dynamic learning framework that learns a consistent ODE, via strong supervisions from both coordinate and measurements levels at two separate decoders, to describe unknown, continuous-time latent dynamics of the intermittently-sampled mmWave Wi-Fi measurements. Compared with the original single-decoder NODE, we show that the additional supervision at the coordinate level leads to strong performance gain. The proposed method was evaluated on an in-house mmWave Wi-Fi dataset and compared with a range of baseline methods.

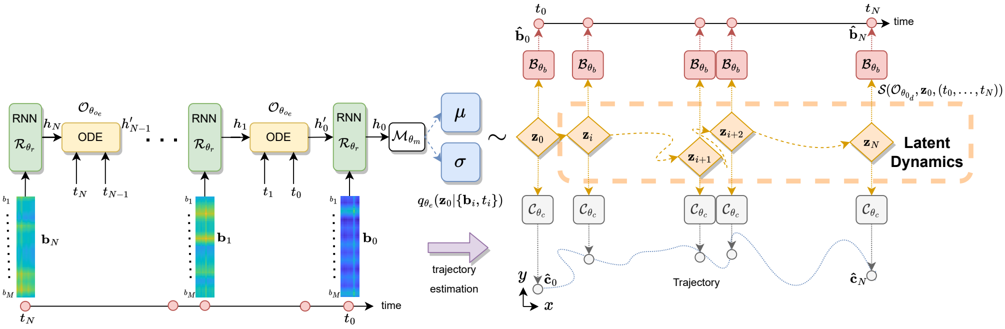


Fig. G.2: Dual-Decoder Neural Dynamic Learning framework.

2 Problem Formulation

We formulate the indoor localization as a sequence regression of beam training measurements within a period of ΔT_w seconds for trajectory estimation. Specifically, stacking the M beam SNRs during one responder channel time t_i as $\mathbf{b}_i = [b_1, b_2, ..., b_M]^T \in \mathbb{R}^{M\times 1}$, the problem of interest is to utilize beam SNR measurements $\{\mathbf{b}_i\}_{i=0}^N$ at time steps $\{t_i\}_{i=0}^N$ with irregular sample intervals to localize the object,

$$\{\mathbf{b}_i, t_i\}_{i=0}^N \to \{\mathbf{c}_i\}_{i=0}^N, \quad s.t. \quad \Delta t_i = t_i - t_{i-1} \neq \Delta t_{i+1} \tag{G.1}$$

where $\mathbf{c}_i = [x_i, y_i]^T$ consists of corresponding two-dimensional coordinates (x_i, y_i) at t_i. This is illustrated in Fig. G.2 where the trajectory estimation is to convert the set of beam SNRs $\{\mathbf{b}_i\}_{i=0}^N$ at intermittently-sampled steps $\{t_i\}_{i=0}^N$ (shown in the left bottom part) to the set of $\{\mathbf{c}_i\}_{i=0}^N$ over a continuous trajectory (shown in the right bottom part).

3 Dual-Decoder Neural Dynamic Learning

We frame our method as a latent-variable model, that we denominate as dual-decoder neural dynamic (DDND). We present our framework in Fig. G.2. Corresponding Fig. G.2 from left to right, in the subsequent sections, we introduce the encoder structure for successive $\{\mathbf{b}_i\}_{i=0}^N$ sequences, the process of learning the latent trajectory, and the learning method empowered by strong supervision to enhance the continuous trajectory learning.

Notation: θ denotes the learnable parameters in neural networks. For simplicity, we use θ_e to denote the joint parameters of all the networks comprising the encoder. We use θ_{o_e} and θ_{o_d} to denote the parameters of the networks comprising the encoder and decoder ODE parts, respectively. We also use θ_r, θ_m, θ_b and θ_c to denote the parameters of the Recurrent Neural Network (RNN), the MLP that outputs the mean and standard deviation of the encoded signal, and the two linear decoders. $\mathcal{S}$ denotes an arbitrary ODE solver.

3.1 Waveform Temporal Information Encoding

Denote a sequence of beamSNR measurements within ΔT_w as $\{\mathbf{b}_i\}_{i=0}^{N} \in \mathbb{R}^{N \times B}$ and its corresponding coordinates $\{\mathbf{c}_i\}_{i=0}^{N} \in \mathbb{R}^{N \times 2}$. We present the input as a temporal sequence to encode the underlying dynamics of the variation of the mmWave Wi-Fi signal with regard to the physical trajectory. We obtain the encoded temporal information of every measurement by forwarding the described temporal inputs through an ODE-RNN network [27]. The ODE and RNN blocks are modeled as neural networks $\mathcal{O}_{\theta_e}(.)$ and $\mathcal{R}_{\theta_e}(.)$, respectively. When forwarding a beamSNR temporal sequence, we reverse the time sequence from t_N to t_0. In this way, the encoder network learns the approximate posterior at time t_0. These Neural ODE [28] blocks are used in the encoder network to model the evolution of the hidden states $\mathbf{h} \in \mathbb{R}^{E}$, where E denotes the dimension of the hidden states. This behavior is modeled in a continuous fashion $\mathbf{h}(t)$, as a solution to an ODE initial-value problem:

$$\frac{d\mathbf{h}(t)}{dt} = \mathcal{O}_{\theta_{o_e}}(\mathbf{h}(t), t), \tag{G.2}$$

$\mathcal{O}_{\theta_{o_e}}(.)$ defines the time-reversed evolution of the observed beamSNR states as the solution of an ODE:

$$\mathbf{h}'_{i-1} = \mathcal{S}(\mathcal{O}_{\theta_{o_e}}, \mathbf{h}_i, (t_i, t_{i-1})), \tag{G.3}$$

then, the hidden state is updated for each observation as a standard RNN update:

$$\mathbf{h}_{i-1} = \mathcal{R}_{\theta_r}(\mathbf{h}'_{i-1}, \mathbf{b}_{i-1}). \tag{G.4}$$

In our approach, we want to characterize $\mathbf{z}_0$ that represents the latent initial state of the encoded trajectory. For that purpose, the mean and the standard deviation of the approximate time-reversed posterior $q_{\theta_{o_e}}(\mathbf{z}_0|\{\mathbf{b}_i, t_i\}_{i=N}^{0})$ are a function of the final hidden state of the encoder:

$$q_{\theta_e}(\mathbf{z}_0|\{\mathbf{b}_i, t_i\}_{i=N}^{0}) = \mathcal{N}(\mu_{\mathbf{z}_0}, \sigma_{\mathbf{z}_0}), \tag{G.5}$$

where

$$\mu_{\mathbf{z}_0}, \sigma_{\mathbf{z}_0} = \mathcal{M}_{\theta_m}(\mathcal{O}_{\theta_{o_e}}(\{\mathbf{b}_i, t_i\}_{i=N}^{0})), \tag{G.6}$$

where $\mathcal{M}_{\theta_m}(.)$ is a neural network translating the last hidden state of the encoder into the mean and variance of the latent initial state $\mathbf{z}_0$.

3.2 Latent Dynamics

Once estimating the approximate posterior $q_{\theta_e}(\mathbf{z}_0|\{\mathbf{b}_i, t_i\}_{i=N}^{0})$, the beamSNR variable-length input sequence $\{\mathbf{b}_i\}_{i=0}^{N}$ is encoded into a fixed-dimensional latent space embedding $\mathbf{z} \in \mathbb{R}^{L}$, where L denotes the dimension of the latent space. The latent trajectory is obtained by first sampling $\mathbf{z}_0 \sim q_{\theta_e}(\mathbf{z}_0|\{\mathbf{b}_i, t_i\}_{i=N}^{0})$ from the estimated posterior. Then, on the decoder side, another ODE $\mathcal{O}_{\theta_{o_d}}$ is modeled as a neural network. During the training, $\mathcal{O}_{\theta_{o_d}}$ will learn the latent trajectory dynamics that relate the variation of the signal and the physical trajectory while during the forward pass, it will query the latent trajectory at the specified time instants. For these matters, $\mathbf{z}_0$ is used as the initial value for the ODE solver on the decoder side:

$$\mathbf{z}_0, ..., \mathbf{z}_N = \mathbf{z}_0 + \int_{t_0}^{t_N} \mathcal{O}_{\theta_{o_d}}(\mathbf{z}_t, t)dt = \mathcal{S}(\mathcal{O}_{\theta_{o_d}}, \mathbf{z}_0, (t_0, ..., t_N)). \tag{G.7}$$

Up to this part, the beamSNR input sequence has been decoded into the latent trajectory $\{\mathbf{b}_i\}_{i=0}^N \rightarrow \{\mathbf{z}_i\}_{i=0}^N$.

3.3 Dual Decoder

In order to guarantee a suitable latent trajectory learning dynamics, we propose to condition the learning by including two linear decoders in the decoder side: waveform reconstruction $\mathcal{B}_{\theta_b}(.)$, and trajectory regression $\mathcal{C}_{\theta_c}(.)$. These two heads will take as input the latent trajectory, in order to perform the reconstruction of the input signal

$$\hat{\mathbf{b}}_i = \mathcal{B}_{\theta_b}(\mathbf{z}_i) = \mathbf{W}_b\mathbf{z}_i + \mathbf{v}_b, \tag{G.8}$$

and its corresponding trajectory regression

$$\hat{\mathbf{c}}_i = \mathcal{C}_{\theta_c}(\mathbf{z}_i) = \mathbf{W}_c\mathbf{z}_i + \mathbf{v}_c, \tag{G.9}$$

where $\mathbf{W}_b$, $\mathbf{W}_c$ denotes the weight matrices and $\mathbf{v}_b$, $\mathbf{v}_c$ the bias vectors, respectively. Note that the input for both decoder heads is the predicted latent trajectory by the decoder ODE. Also, the proposed decoders use shared weights for the input sequences. In this way, we are imposing strong supervision for every time instant in the latent trajectory by using the real trajectory and the variation of the signal as conditions to modify the learning dynamics of the latent trajectory. This leads to an enhancement in learning the continuous dynamics of the trajectory from the latent space.

3.4 Dual-Decoder Neural Dynamic Loss

We propose to train in an end-to-end encoder-decoder structure to minimize the dual-decoder loss which is given by

$$\mathcal{L} = \left[\alpha\|\hat{\mathbf{c}}_0 - \mathbf{c}_0\|_1 + \frac{1}{N}\sum_{i=1}^{N}\|\hat{\mathbf{c}}_i - \mathbf{c}_i\|_1\right] + \beta\left[\frac{1}{N}\sum_{i=0}^{N}\|\hat{\mathbf{b}}_i - \mathbf{b}_i\|_1\right] \tag{G.10}$$

where the first term corresponds to the trajectory regression, and the second term to the waveform reconstruction. We also have two hyperparameters α and β to balance their importance during the learning. In the first term, we are weighting the first coordinate of the trajectory on its own with the α factor. This is done to enhance the trajectory learning, as we are solving an ODE initial-value problem, being the first point of the trajectory determinant for the trajectory regression.

3.5 Complexity Analysis

The time complexity of an ODE-RNN depends on the number of hidden units in the recurrent layer H and the number of time steps in the input sequence T. Then, the time complexity of the forward pass of the ODE-RNN can be approximated as $\mathcal{O}(TH^2)$. Similarly, for the linear layers, an input dimension N and an output dimension M will be expressed as $\mathcal{O}(NM + M)$. In this way, our proposed framework has approximately a time complexity:

$$\mathcal{O}\left(T(H_e^2 + H_d^2 + (L+1)(B+C))\right), \tag{G.11}$$

where the sub-indexes represent the encoder and decoder ODEs, respectively, L is the dimension of the latent space, and B and C are the dimensions of the output linear decoders, respectively.

Remark: Recurrent Neural Networks (RNNs) are not the best solution to learning irregularly-sampled dynamics. Although some tricks have been performed to address this problem, such as inputting the time difference information as a feature or computing an exponential decay between observations [29], they are still not modeling the underline continuous dynamic of an irregular time series data. This highlights the difficulty of intermittent-sampled data. That is why we propose this end-to-end structure that first tries to encode the continuous dynamics of the signal along the trajectory into a latent distribution. This latent distribution represents the encoded dynamics of the motion up to the first location of the trajectory, regardless of the intermittency due to the NODE. Modeling ODEs as neural networks in both the encoder and the decoder side enables learning the continuous behavior of the data and provides great flexibility to design the decoder side.

4 Performance Evaluation

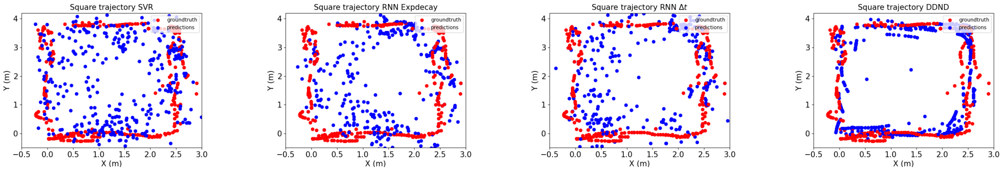
In the following, we present a performance evaluation using real-world mmWave Wi-Fi data from COTS 802.11ad devices.

4.1 mmWave Wi-Fi Localization Dataset

We use 802.11ad-compliant TP-Link Talon AD7200 routers to collect beam SNRs at 60 GHz [30]. A small in-house mmWave Wi-Fi dataset for moving object localization is collected with one router placed on a fixed stand and the other on a TurtleBot as the moving object. The TurtleBot is equipped with a LiDAR and wheel encoder for mapping and gathering location labels. For each responder channel time t_i, the Talon AD7200 router uses 36 directional beampatterns to train the beam, and, hence, we have $M = 36$ for $\mathbf{b}$.

4.2 Implementation

We use a sequence time window of $\Delta T_w = 5$ seconds to group the raw mmWave Wi-Fi beam SNR dataset into sequences with varying numbers of samples, due to the irregular sampling of the beam SNR. We split the sequences into training, validation, and test sets, respectively, with a ratio of $[0.8, 0.1, 0.1]$. We also standardize each entry b_m in the beam SNR by subtracting the mean and normalizing it with the standard deviation. The time vector $\{t_i\}_{i=0}^{N}$ within each ΔT_w is normalized into $[0, 1]$. We implement the proposed method in the Pytorch framework with a hidden state dimension of $E = 20$ and a latent dimension of $L = 20$. We use the 5th-order Runge-Kutta ODE solver for the decoder. We train the network using the Adamax optimizer with a learning rate of 0.01 and no weight decay. The model is trained with a mini-batch size of 32 sequences, and the loss weighting terms are $\alpha = 0.5$ and $\beta = 0.1$.



(a) (b) (c) (d)

Fig. G.3: Visualization of trajectory estimation over selected test sequences: (a) SVR (b) RNN Expdecay (c) RNN Δt (d) DDND

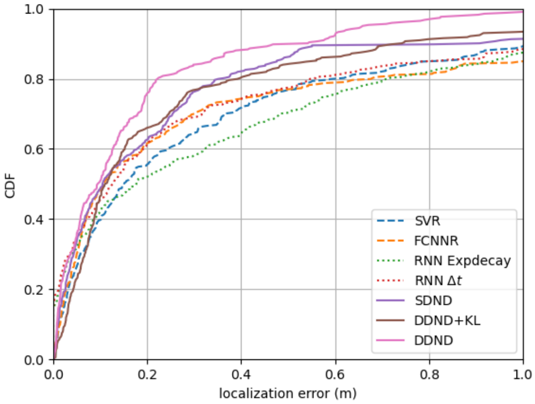


Fig. G.4: Cumulative distribution function (CDF) of localization errors.

4.3 Comparison to Baseline Methods

We evaluate the following baseline methods and variants of our framework for the ablation study:

- Frame-based: 1) support vector regressor (SVR); 2) A fully connected neural network regressor (FCNNR);
- Sequence-based: 3) An RNN with exponential decay; 4) RNN with Δt_i attached to the beam SNR;
- Variants of the proposed framework: 5) single-decoder neural dynamic (SDND); 6) DDND with KL divergence (DDND+KL); 7) DDND without KL divergence (DDND).

For the frame-based methods (i.e., 1) and 2)), the coordinate is estimated only from the current frame. The sequenced-based baseline methods (i.e., 3) and 4)) are standard RNNs with the following modifications to handle the irregularly sampled beam SNRs. For the RNN with exponential decay, it applies an exponential decay from one hidden state to the next $\mathbf{h}'_i = e^{-\Delta t_i}\mathbf{h}_i$. For the RNN Δt, the input is the concatenation of the beam SNR $\mathbf{b}_i$ and the sample interval Δt_i. For 5), we first design a single head decoder model without the waveform reconstruction $\mathcal{B}_{\theta_d}(.)$ decoder, while, for 6), we include an additional KL divergence term to our loss function of (G.10). The localization error is computed as the root mean-squared errors (RMSE) between the estimated $\hat{\mathbf{c}}$ and the ground truth $\mathbf{c}$.

Fig.G.3 shows the estimated trajectories over test sequences for selected methods. For the frame-based SVR method, the point-like coordinate estimates are scattered within the square trajectory in Fig.G.3 (a). This is slightly improved by the sequence-based RNN expdecay method in Fig.G.3 (b) and RNN Δt in Fig.G.3 (c) as more trajectory estimates are pushing towards the square trajectory. By comparing Fig.G.3 (d) to Fig.G.3 (a)-(c), it is clear to see that the proposed DDND is able to learn the underlying dynamics and have more clustered trajectory estimates around the true square trajectory.

Fig.G.4 compares the baseline methods against the proposed solution in terms of the cumulative distribution function (CDF) of the localization error. It shows that proper learning and modeling of the latent dynamics and trajectory motion of the moving object from irregularly-sampled Wi-Fi data can improve the localization performance over the frame-based and traditional sequence-based methods. The proposed method and its variants (5), 6) and 6)) are seen to have significantly fewer large localization errors in Fig.G.4. For instance, the proposed DDND method has 10% fewer localization errors that are larger than 0.6 m than all baseline methods.

Table G.1 further summarizes quantitative performance in terms of the mean, median, and location error corresponding to the 90th percentile of the CDF. It further confirms that the proposed DDND can better deal with the irregularly sampled beam SNRs and model the underlying dynamics of the latent space in a continuous-time fashion. Compared with the modified RNN methods, the DDND method almost reduces the localization error by half for all three performance metrics.

Table G.1: Localization errors (m) on beamSNR localization dataset.

	Mean	Median	CDF@0.9
SVR	0.42	0.15	1.05
FCNNR	0.46	0.11	1.43
RNN Expdecay	0.40	0.18	1.12
RNN Δt	0.33	0.13	1.09
SDND (ours)	0.34	0.11	0.88
DDND+KL (ours)	0.26	0.11	0.74
DDND (ours)	**0.17**	**0.09**	**0.52**

5 Conclusion

This paper tackles the problem of intermittently-sampled mmWave Wi-Fi beam training measurements for localization. Specifically, we proposed the dual-decoder neural dynamic framework that learns the continuous inherent latent dynamics. Performance comparison confirms the performance gain of the proposed method. We plan to scale up the mmWave Wi-Fi dataset by including localization scenarios over multiple trajectories, considering the usage of multiple devices as well as the fusion of multi-band measurement features.

6 Acknowledgements

The work has been supported by European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie Grant agreement No. 813999.

References

[1] P. Bahl and V. N. Padmanabhan, "RADAR: an in-building RF-based user location and tracking system," in *INFOCOM*, March 2000, vol. 2, pp. 775-784.

[2] Z. Yang, Z. Zhou, and Y. Liu, "From RSSI to CSI: Indoor Localization via Channel Response," *ACM Computing Surveys*, vol. 46, no. 2, Dec. 2013.

[3] S. He and S.-H. Chan, "Wi-Fi Fingerprint-Based Indoor Positioning: Recent Advances and Comparisons," *IEEE Communications Surveys Tutorials*, vol. 18, no. 1, pp. 466-490, 2016.

[4] K. Wu et al., "CSI-Based Indoor Localization," *IEEE Transactions on Parallel and Distributed Systems*, vol. 24, no. 7, pp. 1300-1309, July 2013.

[5] M. Pajovic, P. Wang, T. Koike-Akino, H. Sun, and P. V. Orlik, "Fingerprinting-Based Indoor Localization with Commercial MMWave WiFi–Part I: RSS and Beam Indices," in *2019 IEEE Global Communications Conference (GLOBECOM)*, December 2019.

[6] P. Wang, M. Pajovic, T. Koike-Akino, H. Sun, and P. V. Orlik, "Fingerprinting-Based Indoor Localization with Commercial MMWave WiFi–Part II: Spatial Beam SNRs," in *GLOBECOM*, Dec 2019.

[7] T. Koike-Akino, P. Wang, M. Pajovic, H. Sun, and P. V. Orlik, "Fingerprinting-Based Indoor Localization With Commercial MMWave WiFi: A Deep Learning Approach," *IEEE Access*, vol. 8, pp. 84879-84892, 2020.

[8] P. Wang, T. Koike-Akino, and P. V. Orlik, "Fingerprinting-Based Indoor Localization with Commercial MMWave WiFi: NLOS Propagation," in *2020 IEEE Global Communications Conference (GLOBECOM)*, Dec. 2020.

[9] J. Yu, P. Wang, T. Koike-Akino, and P. V. Orlik, "Human Pose and Seat Occupancy Classification with Commercial MMWave WiFi," in *2020 IEEE Global Communications Conference Workshops (GLOBECOM Workshops)*, Dec. 2020.

[10] J. Yu et al., "Multi-Band Wi-Fi Sensing with Matched Feature Granularity," *IEEE Internet of Things Journal*, vol. 8, pp. 1-1, 2022.

[11] T. Koike-Akino, P. Wang, and Y. Wang, "Quantum Transfer Learning for Wi-Fi Sensing," in *IEEE International Conference on Communications (ICC)*, May 2022.

[12] X. Wang, L. Gao, S. Mao, and S. Pandey, "CSI-Based Fingerprinting for Indoor Localization: A Deep Learning Approach," *IEEE Transactions on Vehicular Technology*, vol. 66, no. 1, pp. 763-776, January 2017.

[13] H. Chen, Y. Zhang, W. Li, X. Tao, and P. Zhang, "ConFi: Convolutional Neural Networks Based Indoor Wi-Fi Localization Using Channel State Information," *IEEE Access*, vol. 5, pp. 18066-18074, 2017.

[14] H. Xia et al., "Adversarial Bi-Regressor Network for Domain Adaptive Regression," in *International Joint Conference on Artificial Intelligence (IJCAI)*, 2022, pp. 3608-3614.

[15] J. Yu et al., "Multi-Modal Recurrent Fusion for Indoor Localization," in *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2022.

[16] M. Youssef and A. Agrawala, "The Horus Location Determination System," *Wireless Networks*, vol. 14, no. 3, pp. 357-374, June 2008.

[17] S. Mazuelas et al., "Robust Indoor Positioning Provided by Real-Time RSSI Values in Unmodified WLAN Networks," *IEEE Journal of Selected Topics in Signal Processing*, vol. 3, no. 5, pp. 821-831, October 2009.

[18] X. Wang, L. Gao, and S. Mao, "CSI Phase Fingerprinting for Indoor Localization With a Deep Learning Approach," *IEEE Internet of Things Journal*, vol. 3, no. 6, pp. 1113-1123, Dec. 2016.

[19] X. Wang, L. Gao, and S. Mao, "BiLoc: Bi-Modal Deep Learning for Indoor Localization With Commodity 5GHz WiFi," *IEEE Access*, vol. 5, pp. 4209-4220, 2017.

[20] C. Hsieh, J. Chen, and B. Nien, "Deep Learning-Based Indoor Localization Using Received Signal Strength and Channel State Information," *IEEE Access*, vol. 7, pp. 33256-33267, 2019.

[21] M. T. Hoang, B. Yuen, X. Dong, T. Lu, R. Westendorp, and K. Reddy, "Recurrent Neural Networks for Accurate RSSI Indoor Localization," *IEEE Internet of Things Journal*, vol. 6, no. 6, pp. 10639-10651, Dec. 2019.

[22] X. Wang, L. Gao, S. Mao, and S. Pandey, "DeepFi: Deep learning for indoor fingerprinting using channel state information," in *2015 IEEE Wireless Communications and Networking Conference (WCNC)*, March 2015, pp. 1666-1671.

[23] A. S. Paul and E. A. Wan, "Wi-Fi based indoor localization and tracking using sigma-point Kalman filtering methods," in *PLANS*, 2008, pp. 646-659.

[24] J. Wang and J. G. Park, "An Enhanced Indoor Ranging Method Using CSI Measurements with Extended Kalman Filter," in *PLANS*, 2020, pp. 1342-1348.

[25] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," *Neural Computation*, vol. 9, no. 8, pp. 1735-1780, Nov. 1997.

[26] J. Ding and Y. Wang, "WiFi CSI-Based Human Activity Recognition Using Deep Recurrent Neural Network," *IEEE Access*, vol. 7, pp. 174257-174269, 2019.

[27] Y. Rubanova, R. Chen, and D. K. Duvenaud, "Latent Ordinary Differential Equations for Irregularly-Sampled Time Series," in *Advances in Neural Information Processing Systems*, vol. 32, 2019.

[28] R. Chen et al., "Neural ordinary differential equations," in *Advances in Neural Information Processing Systems*, vol. 31, 2018.

[29] W. Cao et al., "Brits: Bidirectional recurrent imputation for time series," in *Advances in Neural Information Processing Systems*, vol. 31, 2018.

[30] D. Steinmetzer, D. Wegemer, and M. Hollick, "Talon Tools: The Framework for Practical IEEE 802.11ad Research," Available: `https://seemoo.de/talon-tools/`.

Paper H

Object Trajectory Estimation with Continuous-Time Neural Dynamic Learning of Millimeter-Wave Wi-Fi

Cristian J. Vaca-Rubio, Pu Wang, Toshiaki Koike-Akino, Ye Wang,
Petros Boufounos, Petar Popovski

The paper has been submitted in the
IEEE Journal of Selected Topics in Signal Processing (JSTSP), 2023.

© 2023 IEEE

The layout has been revised.

Abstract

In this paper, we leverage standard-compliant beam training measurements from commercial millimeter-wave (mmWave) Wi-Fi communication devices for object localization and, specifically, continuous trajectory estimation and prediction. The main challenge is that the sampling of beam training measurements is intermittent, due to the beam scanning overhead and the uncertainty of the transmission instant caused by the contention over the wireless channel. In order to cope with this intermittency, we devise a method to assist the localization by exploiting the underlying object dynamics. The method consists of a dual-decoder neural dynamic learning framework that reconstructs Wi-Fi beam training measurements at irregular time intervals and learns the unknown latent dynamics in a continuous-time fashion powered by the use of an ordinary differential equation (ODE). Utilizing the variational autoencoder (VAE) framework, we have derived a modified evidence lower bound (ELBO) loss function for the dual-decoder architecture that balances the unsupervised waveform reconstruction and supervised coordinate estimation tasks. To evaluate the proposed method, we build an in-house testbed consisting of commercial 802.11*ad routers, with a TurtleBot as a mobile user, and collect a real-world mmWave Wi-Fi beam training dataset. Our results demonstrate substantial performance improvements over a list of baseline methods, further validated through an extensive ablation study*[1]*.*

Keywords— WLAN sensing, Wi-Fi, 802.11ad/ay, 802.11bf, localization, fingerprinting, beam training, dynamic learning.

1 Introduction

Wi-Fi sensing has been an integral part of emerging integrated sensing and communications (ISAC), as corroborated by the establishment of a new 802.11bf WLAN Sensing task group for robust and reliable sensing in September 2020. It aims to make greater use of 802.11 Wi-Fi signals for reliable and secure wireless sensing towards new industrial and commercial applications in home security, entertainment, energy management (HVAC, light, device power savings), elderly care, and assisted living.

The scope of 802.11bf covers both sub-7 GHz and millimeter (mmWave) Wi-Fi sensing above 45 GHz, built on 802.11g/n/ac/ax/be and, respectively, 802.11ad/ay standards. For the mmWave Wi-Fi sensing or Directional Multi-Gigabit (DMG) sensing, the new 802.11bf opens up possibilities of reusing beam training measurements for sensing applications. Our work is to leverage such sensing-supported beam training measurement for indoor localization of moving objects such as a robot or a mobile user; see an illustration in Fig. H.1.

Along with coarse-grained received signal strength indicator (RSSI) [2–5] and fine-grained channel state information (CSI) at sub-7 GHz [6–14], mid-grained mmWave beam training measurements have been previously explored in [15–23][2]. Most of existing approaches are frame-based. That is, the object location is inferred from the current Wi-Fi frame, without

---

[1] Part of this paper was presented on ICASSP 2023 [1]

[2] Please refer to [21, Section II] for detailed discussions on all three types of Wi-Fi channel measurements.

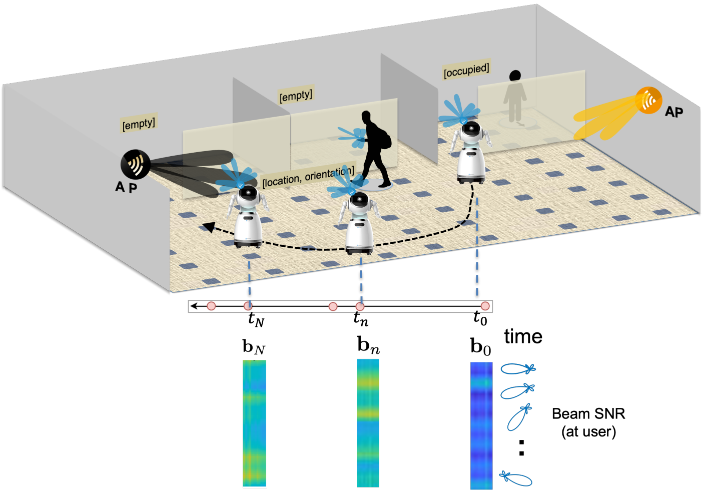


Fig. H.1: Trajectory estimation of moving objects using mmWave Wi-Fi beam training measurements.

integration of past measurements or previous trajectory history. For the frame-based setting, traditional machine learning and advanced deep learning methods have been applied to all Wi-Fi fingerprinted measurements [2, 5, 24–27]. For instance, the k-nearest neighbor (kNN), support vector machine (SVM), and decision trees (DT) were applied to the RSSI-based fingerprinting method [2, 3]. DeepFi exploits 90 CSI amplitudes from all the subcarriers at three antennas for feature extraction using an autoencoder architecture [6, 7]. More recently, a pre-trained fusion network between the CSI at sub-7 GHz and the beam training measurements at 60 GHz was proposed for both localization and device-free sensing tasks [21].

On the other hand, *sequence-based* approaches take consecutive Wi-Fi frames as the input, and state estimation (e.g, Kalman filter-like approaches [28, 29]) and recurrent neural networks (e.g., GRU and LSTM [30]) can be applied for trajectory estimation with the RSSI and CSI [5, 9–12] at sub-7 GHz; More detailed discussion on sequence-based solutions in Section. 2. However, the sequence-based formulation has NOT been applied to mmWave Wi-Fi beam training measurements due to the following fundamental technical challenges:

1) **Low beam training rate**: mmWave Wi-Fi such as 802.11ad/ay organizes access to the medium in beacon intervals (BIs), usually in the order of 100ms. A unique feature of mmWave Wi-Fi at 60 GHz is the use of directional beamforming to compensate for path loss and identify unassociated stations at a further distance. As shown in Fig. H.2, a BI consists of two main access periods: beacon header interval (BHI) and data transmission interval (DTI). While DTI is mainly used for data transmission, mmWave Wi-Fi (802.11ad/ay) devices are required to perform beam training during the BHI to initiate the data transmission. The BHI is further subdivided into three sub-intervals: beacon transmission interval (BTI), association beamforming training (A-BFT), and announcement transmission interval (ATI) for management frame

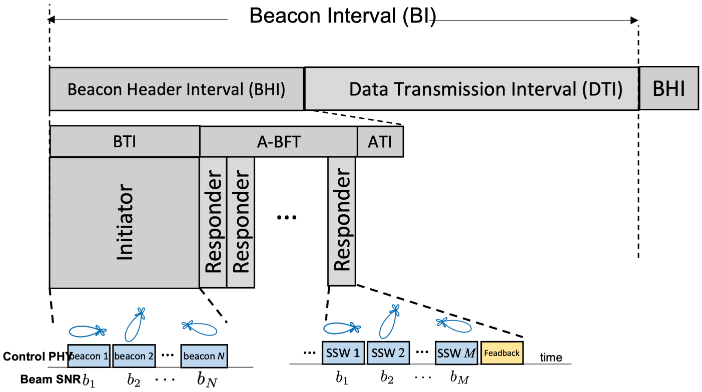


Fig. H.2: MmWave Wi-Fi beam training protocol during the mandatory beacon transmission interval (BTI) and association beamforming training (A-BFT) in 802.11ad/ay standards.

exchange between the AP and beam-trained stations.

During the BTI sub-interval, an access point (AP) sends directional beacon frames to train its transmission sector-level beampatterns, also referred to as the downlink BTI beam training. Multiple users can simultaneously compute their own received beam SNRs corresponding to each of the transmitted beampatterns using a quasi-omnidirectional receiving beampattern. This mandatory beam training results in significant overhead to the Wi-Fi network and it is desired to limit the number of directional beampatterns within a beacon and the total number of beacons, resulting in sparsely sampled beam measurements than Wi-Fi at sub-7 GHz.

2) **Irregular sample intervals**: At the A-BFT subinterval, the users or responders (e.g., mobile devices) can train its (transmitting or receiving) beampatterns by sending a sequence of (short) sector sweep (SSW) frames to the AP, as shown in Fig. H.2. Compared with the downlink BTI beam training for multiple users simultaneously, the uplink A-BFT beam training is reserved for one responder at a time. Specifically, A-BFT is slotted up to 8 slots in 802.11ad and 40 in 802.11ay. Multiple responders randomly choose one of the slots for transmitting SSW frames. Consequently, when multiple responders exist, each responder needs to contend the channel time, and one responder is randomly selected. As a result, such contention-based channel access results in irregularly sampled beam SNR measurements at AP for a given user.

To address the above challenges and inspired by recent advances in neural ordinary differential equation (ODE) [31–37], this paper proposes a dual-decoder neural dynamic learning framework that learns consistent latent dynamics described by a unified ODE for both waveform reconstruction and coordinate estimation. Compared with the original neural ODE, our dual-decoder structure enforces that the learned latent dynamics not only recover the input sequences in the waveform domain but also map to the object trajectory, thus grounding the latent dynamic learning into the physical (coordinate) space. This dual-decoder structure is further enhanced by the introduction of a modified evidence lower bound (ELBO) loss function to couple the losses from the dual decoder. It is worth noting that a conference version of the proposed method was published in [1] with limited derivation and performance evaluation. This paper significantly expands [1] with the following contributions:

1. We propose the first-of-its-kind sequence-to-sequence object trajectory estimation workflow that can handle a set of consecutive Wi-Fi measurements at irregular sample instances and directly output the whole trajectory over the same time interval.
2. We present a dual-decoder neural network that learns the underlying latent dynamics in a continuous-time fashion by exploring the neural ODE framework. The dual-decoder architecture regularizes the learnable neural ODE function in the latent space by grounding it to the coordinate-level measurable space.
3. We derive a customized cost function by extending the ELBO for the sequence-to-sequence formulation and the dual decoder structure.
4. We build an in-house mmWave Wi-Fi testbed consisting of commercial 802.11ad-compliant Wi-Fi routers and a moving robot with coordinates labels. The mmWave Wi-Fi trajectory estimation dataset was collected continuously over several hours in a span of days[3].
5. We benchmark our proposed method against a list of baseline methods including frame-based and sequence-based approaches using classic machine learning and state-of-art deep learning pipelines.
6. We shed more light on the representation capacity of the proposed method via comprehensive ablation studies on 1) sequence length, 2) supervision intensity (regular versus dense) at the coordinate decoder, 3) day-to-day generalization, 4) varying tasks (trajectory estimation versus extrapolation), and 5) inspection of latent space.

The remainder of this paper is organized as follows. The problem formulation is described in Section 2 where existing solutions are also briefly reviewed. Section 3 introduces our dual-decoder neural dynamic learning framework, the derivation of a customized cost function, and an analysis of computational complexity. Section 4 describes the data collection system and the mmWave Wi-Fi trajectory estimation dataset collected over multiple days. Simulation and experimental results are given in Section 5, followed by conclusions in Section 6.

2 Problem Formulation and Existing Solutions

2.1 Problem Formulation

We formulate the indoor localization as a sequence-to-sequence regression problem using the mmWave Wi-Fi beam training measurements over a period of ΔT_w seconds. Specifically, stacking a set of N_b beam SNRs computed at AP over an A-BFT slot t_i as $\mathbf{b}_n = [b_1, b_2, ..., b_{N_b}]^T \in \mathbb{R}^{N_b \times 1}$, the problem of interest is to utilize beam SNR measurements $\{\mathbf{b}_n\}_{n=0}^N$ at time steps $\{t_n\}_{n=0}^N$ with irregular sample intervals to localize the object,

$$\{\mathbf{b}_n, t_n\}_{n=0}^N \rightarrow \{\mathbf{c}_n\}_{n=0}^N, \quad s.t. \quad \Delta t_n = t_n - t_{n-1} \neq \Delta t_{n+1} \tag{H.1}$$

---

[3] We plan to open release our mmWave Wi-Fi trajectory estimation dataset when the paper is public. We open released our earlier mmWave Wi-Fi localization dataset at fixed locations at https://www.merl.com/demos/mmBSF.

where $\mathbf{c}_n = [x_n, y_n]^T$ consists of corresponding two-dimensional coordinates (x_n, y_n) at t_n. This is illustrated in Fig. H.3 where the trajectory estimation is to convert the set of beam SNRs $\{\mathbf{b}_n\}_{n=0}^N$ at intermittently-sampled steps $\{t_n\}_{n=0}^N$ (shown in the top left part) to the set of $\{\mathbf{c}_n\}_{n=0}^N$ over a continuous trajectory (shown in the right bottom part).

Like fingerprinting-based Wi-Fi localization methods, we collect beam SNR measurements and corresponding two-dimensional coordinate labels. As detailed in Section 4, the beam SNR data is continuously collected over several hours each day and spanning over multiple days. We then divide the beam SNRs into non-overlapping sequences and split the sequences into training and test datasets without any data leakage. The learning-based approach is to extract time-dependent features from the sequence of beam SNR measurements $\{\mathbf{b}_n, t_n\}_{n=0}^N$ and regress these features to a trajectory or a sequence of coordinate $\{\mathbf{c}_n\}_{n=0}^N$ from the training dataset. Once the model is trained, the trajectory estimation performance is evaluated in the test dataset.

2.2 Existing Solutions

The frame-based approach takes the beam SNR measurement $\mathbf{b}_n$ at a time step t_n and directly estimates the corresponding coordinate $\hat{\mathbf{c}}_n$. Specifically, we have

$$\{\mathbf{b}_n\} \rightarrow \mathbf{c}_n = \{x_n, y_n\}. \tag{H.2}$$

As we mentioned earlier, classic machine learning methods such as support vector regression (SVR) [38] and Gaussian processing (GP) [39] and deep learning pipelines such as multi-layer perceptron (MLP) [40, 41] and convolution neural network (CNNs) [42, 43] can be applied.

Similar to our formulation, sequence-based approaches take multiple consecutive beam training measurement and estimate a trajectory

$$\{\mathbf{b}_n\}_{n=0}^N \rightarrow \{\mathbf{c}_n\}_{n=0}^N, \tag{H.3}$$

with or without time sampling instances t_n. In the case of regularly sampled data, $\Delta t_1 = \Delta t_2 = \cdots = \Delta t_N$ is constant in both the training and test datasets such that t_n becomes irrelevant. In this case, standard RNN can be used to utilize historic Wi-Fi data [5, 10, 12] and learn time-dependent features for trajectory estimation. Particularly, an LSTM is to estimate the conditional probability [30]

$$p(\mathbf{c}_n | \{\mathbf{b}_i\}_{n=n-N+1}^n). \tag{H.4}$$

where each LSTM unit is trained to sequentially update time-dependent latent (hidden) variables $\mathbf{h}_n$ using previous latent variable $\mathbf{h}_{n-1}$ and the current mmWave beam training measurement $\mathbf{b}_n$

$$\begin{aligned} \mathbf{h}_n' &= \mathbf{h}_{n-1}, \\ \mathbf{h}_n &= \mathcal{R}(\mathbf{h}_n', \mathbf{b}_n; \theta), \quad n = 0, 1, \cdots, N, \end{aligned} \tag{H.5}$$

where we introduce an auxiliary variable $\mathbf{h}_n'$ for a unified interpretation and $\mathcal{R}$ represents an LSTM unit with trainable parameters θ. The detailed implementation of $\mathcal{R}$ is shown in

Appendix 8. The updated latent variable $\mathbf{h}_n$ can be used to infer the coordinate sequentially and, hence, estimate the trajectory.

When the input signal is irregularly sampled, one can augment the beam SNR with corresponding sampling interval Δt_n and feed the augmented beam SNR to the LSTM unit as

$$\begin{aligned} \mathbf{h}'_n &= \mathbf{h}_{n-1}, \quad \tilde{\mathbf{b}}_n = \left[\mathbf{b}_n^T, \Delta t_n\right]^T, \\ \mathbf{h}_n &= \mathcal{R}(\mathbf{h}'_n, \tilde{\mathbf{b}}_n; \theta), \quad n = 0, 1, \cdots, N. \end{aligned} \tag{H.6}$$

We refer to the above as *RNN*-Δ for baseline comparison in Section 5.

Furthermore, one can further damp the latent variable with an exponential decaying factor to decide the amount of hidden state should be kept from the previous time step to the current time step:

$$\begin{aligned} \mathbf{h}'_n &= \mathbf{h}_{n-1} e^{-\Delta t_{n-1}}, \\ \mathbf{h}_n &= \mathcal{R}(\mathbf{h}'_n, \mathbf{b}_n; \theta), \quad n = 0, 1, \cdots, N, \end{aligned} \tag{H.7}$$

which is referred to as *RNN-Decay*.

3 Dual-Decoder Neural Dynamic Learning

As opposed to the above existing solutions, we propose a dual-decoder neural dynamic (DDND) framework that explicitly utilizes the continuous-time ODE function and its numerical solver to propagate the time-dependent latent feature from one time instance to the next time instance with an irregular time interval and regularizes the learning of latent ODE by grounding it to both waveform and observable coordinate spaces.

More specifically, Fig. G.2 shows our DDND framwork that takes a sequence of beam SNR measurements $\{\mathbf{b}_n\}_{n=0}^N$ to the encoder (Left) and reconstructs the beam SNR measurement in one of the decoders (Top right) and outputs a sequence of coordinate estimates in the other (Bottom right). On the encoder side, the latent dynamic learning is achieved by solving a shared ODE function over the union of two sets of time instances (Center right): one for waveform reconstruction and the other for trajectory estimation. In the following, we introduce the main blocks in order.

3.1 Encoder

Our encoder follows the unrolled RNN architecture but in a reverse-time order from t_N to t_0; that is, at a time instance n, the beam SNR measurement $\mathbf{b}_n$ and a time-dependent latent feature $\mathbf{h}'_n$, properly propagated from the previous time step $\mathbf{h}_{n+1}$, are fed into a standard LSTM unit to an updated latent feature $\mathbf{h}_n$ of dimension L_h as

$$\mathbf{h}_n = \mathcal{R}_e(\mathbf{h}'_n, \mathbf{b}_n; \theta_r), \quad n = N, N-1, \cdots, 0, \tag{H.8}$$

where $\mathcal{R}_e$ is a standard LSTM update step with associated learnable parameters θ_r defined in Appendix 8. Note that θ_r is shared over all time steps in the encoder.

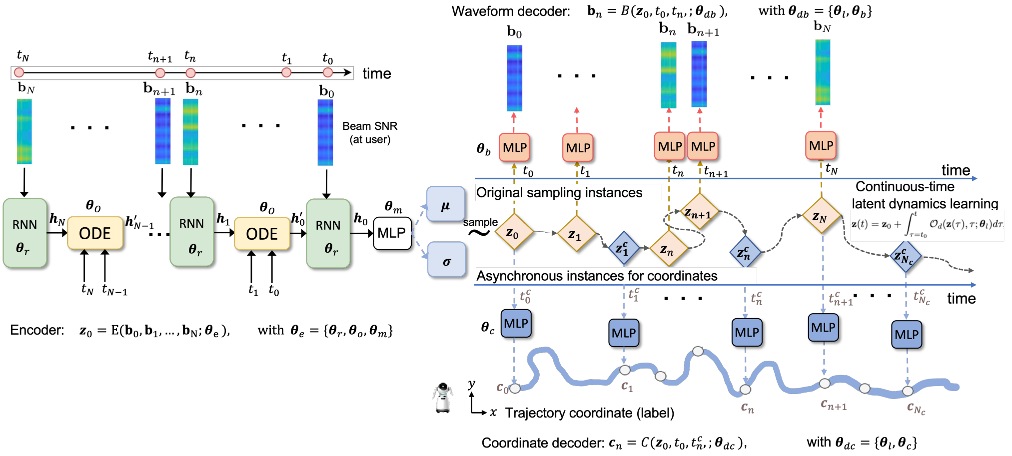


Fig. H.3: Object trajectory estimation using mmWave Wi-Fi beam training measurements with neural dynamic learning. Left: Encoder maps a sequence of irregularly sampled Wi-Fi beam SNRs into a continuous-time latent space and infers its initial condition $\mathbf{z}_0$. Top right: Waveform Decoder maps latent dynamic states in the same sampling time instances back to the original measurement space for waveform reconstruction. Middle right: Numerical ODE solvers compute latent states at any queried time instances (e.g., original sampling time instances t_n or time instances with coordinate labels t_n^c) with a trainable ODE function and sampled initial condition $\mathbf{z}_0$. Bottom right: Coordinate Decoder maps latent dynamic states into observable coordinate space at asynchronous/new queried time instances to regularize the learning of latent dynamic ODE function.

Different from RNN-Δ and RNN-Decay, we adopt a continuous-time ODE function to explicitly describe the evolving of $\mathbf{h}_{n+1}$ at time t_{n+1} to the auxiliary latent $\mathbf{h}'_n$ at time t_n. Mathematically, the continuous-time ODE function is given as [31, 32]

$$\frac{d\mathbf{h}(t)}{dt} = \mathcal{O}_e(\mathbf{h}(t), t; \theta_o), \tag{H.9}$$

where $\mathcal{O}_e$ is represented by a neural network, e.g., an MLP, parameterized by θ_o. Given the latent variable $\mathbf{h}_{n+1}$ at time t_{n+1} and $\mathcal{O}_e$, one can numerically solve the propagated auxiliary variable $\mathbf{h}'_n$ at time t_n as

$$\mathbf{h}'_n = \mathbf{h}_{n+1} + \int_{\tau=t_{n+1}}^{t_n} \mathcal{O}_e(\mathbf{h}(\tau), \tau; \theta_o) d\tau, \tag{H.10}$$

In most cases, the above integration is implemented using a numerical ODE solver, e.g., Euler and Runge-Kutta solvers:

$$\mathbf{h}'_n = \mathcal{S}(\mathcal{O}_e, \mathbf{h}_{n+1}, (t_{n+1}, t_n)), \tag{H.11}$$

where $\mathcal{S}$ represents a specific ODE solver.

By comparing (H.10) and (H.7), it is clear that the ODE-based latent propagation from t_{n+1} to t_n can be more representative for different modes in the latent space and different time intervals Δt_n, while the decay-based propagation is only exponentially monotonic from the starting point $\mathbf{h}_{n+1}$.

By iterating the latent propagation step (H.10) and the RNN unit (H.8), the input sequence $\{\mathbf{b}_n\}_{n=0}^{N}$ can be represented by the latent variable $\mathbf{h}_0$ at time t_0, as shown in the left side of Fig. G.2. We then use $\mathbf{h}_0$ of dimension L_h to generate $\mathbf{z}_0$ of dimension L to represent the initial state of the latent trajectory of the input sequence via an approximate posterior. More specifically,

$$q_{\theta_m}(\mathbf{z}_0|\mathbf{h}_0) = q_{\theta_e}(\mathbf{z}_0|\mathbf{b}_0, \cdots, \mathbf{b}_N) = \mathcal{N}(\mu, \sigma^2), \tag{H.12}$$

where $\theta_e = \{\theta_r, \theta_o, \theta_m\}$ groups all learnable parameters in the encoder, and the mean and standard deviation are mapped from $\mathbf{h}_0$ as

$$\mu, \sigma = \mathcal{M}_e(\mathbf{h}_0; \theta_m), \tag{H.13}$$

where $\mathcal{M}_e$ is a neural network, e.g., MLP, with learnable parameters θ_m that converts the last hidden state of the encoder into an output of dimension $2L$ for the mean and standard deviation vectors.

3.2 Latent Dynamics Learning

Following the variational autoencoder (VAE) [44], we first sample $\mathbf{z}_0 \sim q_{\theta_e}(\mathbf{z}_0|\mathbf{b}_0, \cdots, \mathbf{b}_N)$ according to the approximate posterior in (H.12) via the reparameterization trick,

$$\mathbf{z}_0 = \mu + \sigma \odot \epsilon, \quad \epsilon \sim \mathcal{N}(\mathbf{0}, \mathbf{I}) \tag{H.14}$$

where μ and σ are given in (H.13) from the encoder output and $\odot$ denotes an element-wise product[4].

Given the sampled initial value in the latent space $\mathbf{z}_0$, we would like to learn a unified continuous-time latent dynamic trajectory $\mathbf{z}(t)$ that can be mapped into the original sampled waveform, i.e., beam SNRs, and the labeled trajectory coordinates. The latent dynamics learning of $\mathbf{z}(t)$ is achieved by using another continuous-time ODE function $\mathcal{O}_d$ modeled by a neural network with parameters θ_l; see the middle right portion of Fig. G.2,

$$\begin{aligned} \frac{d\mathbf{z}(t)}{dt} &= \mathcal{O}_d(\mathbf{z}(t), t; \theta_l) \\ \rightarrow \mathbf{z}(t) &= \mathbf{z}_0 + \int_{\tau=t_0}^{t} \mathcal{O}_d(\mathbf{z}(\tau), \tau; \theta_l) d\tau, \end{aligned} \tag{H.15}$$

where $\mathbf{z}_0 = \mathbf{z}(t)|_{t=t_0}$. Consequently, we can resort to the numerical ODE solver to compute the latent variables at the original beam SNR sampling instances $\{t_n\}_{n=0}^{N}$ and at new (potentially

---

[4] An alternative reparameterization is to generate $\log \sigma^2$, the logarithm of the variance, at the output of MLP $\mathcal{M}_e$ and sample the initial condition as $\mathbf{z}_0 = \mu + e^{0.5 \log \sigma^2} \odot \epsilon$.

asynchronous) queried time instances $\{t_n^c\}_{n=0}^{N_c}$ for grounding the latent dynamic learning into the physical coordinate space. For the original set of beam SNR sampling instances $\mathcal{T} = \{t_0, t_1, \cdots, t_N\}$, the latent variables at t_n can be numerically solved as

$$\begin{aligned}\mathbf{z}_n = \mathbf{z}(t)|_{t=t_n} &= \mathbf{z}_0 + \int_{t_0}^{t_n} \mathcal{O}_d(\mathbf{z}(t), t; \theta_l)dt \\ &= \mathcal{S}(\mathcal{O}_d, \mathbf{z}_0, (t_0, t_n); \theta_l), \quad n = 1, \cdots, N,\end{aligned} \tag{H.16}$$

where $\mathcal{S}$ is a specific numerical ODE solver like the one used in the encoder. For the new, potentially asynchronous time instances $\mathcal{T}_c = \{t_0^c, t_1^c, \cdots, t_{N_c}^c\}$, we have

$$\begin{aligned}\mathbf{z}_n^c = \mathbf{z}(t)|_{t=t_n^c} &= \mathbf{z}_0 + \int_{t_0^c}^{t_n^c} \mathcal{O}_d(\mathbf{z}(t), t; \theta_l)dt \\ &= \mathcal{S}(\mathcal{O}_d, \mathbf{z}_0, (t_0^c, t_n^c); \theta_l), \quad n = 1, \cdots, N_c,\end{aligned} \tag{H.17}$$

where $t_0 = t_0^c$ as the same initial time instance for both encoders. Note that (H.16) and (H.17) share the same ODE function parameters θ_l and the same initial condition $\mathbf{z}_0$ for a unified latent dynamic representation.

3.3 Dual Decoder

For the decoder, we propose to use a mixture of two decoding branches: one is for unsupervised waveform reconstruction $\mathcal{M}_w$ and the other for supervised trajectory estimation $\mathcal{M}_c$. Our presented framework exploits the fact that, once $\mathcal{O}_d(\mathbf{z}(t), t; \theta_l)$ is learnt, the latent dynamic learning can be used to query any arbitrary time instant (i.e., t_n for waveform reconstruction and t_n^c for supervised trajectory estimation) within the trained time horizon. Then, the unified latent dynamic representation $\mathcal{O}_d(\mathbf{z}(t), t; \theta_l)$ is enforced not only to recover the waveform in an unsupervised fashion but also to be grounded to a two-dimensional trajectory dynamics via labelled object coordinates at asynchronous time instances. Specifically, the waveform decoder takes the computed latent variables at t_n as input and output the sequence of original beam SNRs at t_n,

$$\hat{\mathbf{b}}_n = \mathcal{M}_b(\mathbf{z}_n; \theta_b), \quad n = 1, 2, \cdots, N, \; = \mathbf{W}_{b_2}\text{ReLU}(\mathbf{W}_{b_1}\mathbf{z}_n + \mathbf{v}_{b_1}) + \mathbf{v}_{b_2},$$

where $\theta_b = \{\mathbf{W}_{b_1/b_2}, \mathbf{v}_{b_1/b_2}\}$ groups the weight matrices and bias terms and $\text{ReLU}(\mathbf{x}) = \max(0, \mathbf{x})$ is the rectified linear unit (ReLU) activation function. Similarly, the coordinate decoder

$$\hat{\mathbf{c}}_n = \mathcal{M}_c(\mathbf{z}_n^c; \theta_c), \quad n = 1, 2, \cdots, N_c, \; = \mathbf{W}_{c_2}\text{ReLU}(\mathbf{W}_{c_1}\mathbf{z}_n^c + \mathbf{v}_{c_1}) + \mathbf{v}_{c_2},$$

where $\theta_c = \{\mathbf{W}_{c_1/c_2}, \mathbf{v}_{c_1/c_2}\}$ groups the weight matrices and bias terms. For both decoders, the parameters θ_w and θ_c are shared over time instances t_n and, respectively, t_n^c.

Combining (H.18) and (H.16), we can directly decode $\hat{\mathbf{b}}_n$ from the sampled latent variable $\mathbf{z}_0$ as

$$\hat{\mathbf{b}}_n = \mathcal{M}_w(\mathcal{S}(\mathcal{O}_d, \mathbf{z}_0, (t_0, t_n); \theta_l); \theta_w) = \mathcal{B}(\mathbf{z}_0, t_0, t_n; \theta_{db}),$$

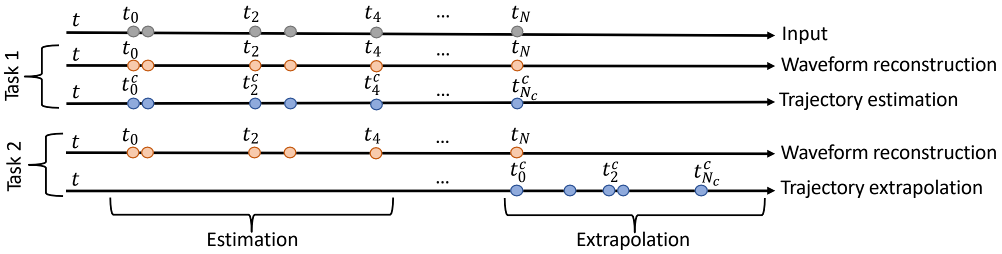


Fig. H.4: Illustration of different supervision tasks (trajectory estimation and trajectory extrapolation).

where $\mathcal{B}$ stands for the integrated waveform decoder (the latent dynamic learning and waveform decoder) for beam SNRs and $\theta_{db} = \{\theta_l, \theta_b\}$ groups all trainable parameters to decode $\hat{\mathbf{b}}_n$. Similarly, we have

$$\hat{\mathbf{c}}_n = \mathcal{M}_c(\mathcal{S}(\mathcal{O}_d, \mathbf{z}_0, (t_0, t_n^c); \theta_l); \theta_c) = \mathcal{C}(\mathbf{z}_0, t_0, t_n^c; \theta_{dc}),$$

where $\mathcal{C}$ stands for the integrated coordinate decoder (the latent dynamic learning and coordinate decoder) for **c**oordinates and $\theta_{dc} = \{\theta_l, \theta_c\}$. In this way, we are imposing strong supervision for every time instant in the latent trajectory by using the real trajectory and the variation of the signal as conditions to modify the learning dynamics of the latent trajectory. This leads to an enhancement in learning the continuous dynamics of the trajectory from the latent space.

For the supervised coordinate decoder, we can have two levels of supervision intensity,

- **Regular Supervision**: the coordinate decoder maps the continuous-time latent dynamics onto the exact same time instants of the input beam SNR sequences. i.e., $t_n = t_n^c$ and $N = N_c$.
- **Dense Supervision**: the coordinate decoder maps the latent dynamics onto more densely queried time instances with respect to those instances in the waveform decoder, i.e., $t_0 = t_0^c, t_N = t_{N_c}^c$, and $N \ll N_c$. In other words, we enforce the latent dynamic learning to be consistent over more queried time instances within the same time window.

In terms of the time horizon of latent dynamics learning, we can have two supervision tasks,

- **Trajectory Estimation**: In the trajectory estimation task, we condition the encoder on the subset of points $(t_0, ..., t_N)$ and reconstruct the same set of points in the same time interval in the decoder side for both the waveform reconstruction and the trajectory regression, i.e., $t_n = t_n^c$ and $N = N_c$. This is illustrated in Task 1 of Fig. H.4.
- **Trajectory Extrapolation**: For the trajectory extrapolation, we predict the object trajectory over the time window ΔT_w immediately after the time window ΔT_w of the input beam SNR sequences; see Task 2 of Fig. H.4. In other words, we extend the time horizon of the latent dynamics twice as that of Task 1. Particularly, we have $t_N = t_0^c$.

3.4 Customized Loss Function

In the following, we propose a customized loss function that is modified from the VAE-based loss function for the dual decoder architecture. Grouping the input sequence of beam SNRs $\mathbf{b} = \{\mathbf{b}_n\}_{n=0}^{N}$ (and similarly $\mathbf{c} = \{\mathbf{c}_n\}_{n=0}^{N_c}$), we can simplify the approximate posterior distribution of (H.12) as $q_{\theta_e}(\mathbf{z}_0|\mathbf{b})$. The original loss function is to maximize the marginal likelihood function $p(\mathbf{b})$. However, due to its intractability, one can instead maximize an evidence lower bound (ELBO) [44] modified for the dual decoder

$$\begin{aligned}\text{ELBO} =& \mathbb{E}[\log p(\mathbf{b}|\mathbf{z}_0)] + \lambda \mathbb{E}[\log p(\mathbf{c}|\mathbf{z}_0)] \\ & - \text{KL}(q_{\theta_e}(\mathbf{z}_0|\mathbf{b})||p(\mathbf{z}_0)), \\ \overset{(a)}{\approx} & \frac{1}{M}\sum_{m=1}^{M} \log p(\mathbf{b}|\mathbf{z}_0^{(m)}) + \frac{\lambda}{M}\sum_{m=1}^{M} \log p(\mathbf{c}|\mathbf{z}_0^{(m)}) \\ & + 0.5 \sum_{l}^{L} \left(1 + \log \sigma_l^2 - \mu_l^2 - \sigma_l^2\right),\end{aligned} \tag{H.18}$$

where λ is the regularization parameter on the coordinate likelihood function in addition to the original KL divergence, $p(\mathbf{z}_0)$ is the prior of $\mathbf{z}_0$ which is assumed to be [44]

$$p(\mathbf{z}_0) = \mathcal{N}(\mathbf{0}, \mathbf{I}), \tag{H.19}$$

the expectation is with respect to the posterior distribution $q_{\theta_e}(\mathbf{z}_0|\mathbf{b})$ and (a) holds since we replaces the posterior mean by the its sample mean over M samples $\mathbf{z}_0^{(m)}$ according to (H.12) and due to the fact that the KL divergence can be analytically derived between the Gaussian prior $p(\mathbf{z}_0)$ and the approximate Gaussian posterior distribution in (H.12). In addition, μ_l and σ_l denote the posterior mean and, respectively, standard deviation of the l-th element of $\mathbf{z}_0$ given the input sequence $\mathbf{b}$.

To compute the likelihood functions $\log p(\mathbf{c}|\mathbf{z}_0)$ for the coordinate decoder (similarly $\log p(\mathbf{b}|\mathbf{z}_0)$ for the waveform decoder), we invoke an independent assumption over the elements of the decoder output $\mathbf{c} = \{\mathbf{c}_0, \mathbf{c}_1, \cdots, \mathbf{c}_N\}$. This is similar to the pixel-wise independence at the decoder output used in the VAE. This implies that

$$\begin{aligned}\log p(\mathbf{c}|\mathbf{z}_0^{(m)}) &= \log p(\mathbf{c}_0, \cdots, \mathbf{c}_N|\mathbf{z}_0^{(m)}) \\ &= \sum_n \log p(\mathbf{c}_n|\mathbf{z}_0^{(m)}) \\ &= \sum_n \sum_d \log p(c_{n,d}|\mathbf{z}_0^{(m)}),\end{aligned} \tag{H.20}$$

where $d = \{1, 2\}$ denotes the 2D coordinate and $c_{n,d}$ denotes the x- and y-coordinate at time n. The element-wise likelihood function $p(c_{n,d}|\mathbf{z}_0)$ of the (x- or y-) coordinate follows a Laplace distribution as

$$p(c_{n,d}|\mathbf{z}_0^{(m)}) = \frac{1}{2a_c} e^{-\frac{\left|c_{n,d} - \mathcal{C}_d(\mathbf{z}_0^{(m)}, t_0, t_n^c; \theta_{dc})\right|}{a_c}}, \tag{H.21}$$

where a_c is a scaling parameter, and $\mathcal{C}_d(\mathbf{z}_0^{(m)}, t_0, t_n^c; \theta_{dc})$ is the d-th element of the output at the integrated coordinate decoder of (H.18) at time t_n^c. Then, it is easy to see that

$$\mathbb{E}[\log p(\mathbf{c}|\mathbf{z}_0^{(m)})] \propto -\frac{1}{a_c} \sum_{n=0}^{N_c} \left\| \mathbf{c}_n - \mathcal{C}(\mathbf{z}_0^{(m)}, t_0, t_n^c; \theta_{dc}) \right\|_1, \tag{H.22}$$

where $\|\cdot\|_1$ denotes the ℓ_1 norm. Similarly, we have

$$\mathbb{E}[\log p(\mathbf{b}|\mathbf{z}_0^{(m)})] \propto -\frac{1}{a_b} \sum_{n=0}^{N} \left\| \mathbf{b}_n - \mathcal{B}(\mathbf{z}_0^{(m)}, t_0, t_n; \theta_{db}) \right\|_1, \tag{H.23}$$

where a_b is a scaling parameter for the Laplace distribution of the beam SNR $\mathbf{b}_n$. As a result, considering $a_b = a_c = 1$, we aim to minimize the negative customized ELBO of (H.18) for the dual decoder as

$$-\text{ELBO} \propto \sum_{n=0}^{N} \|\hat{\mathbf{b}}_n - \mathbf{b}_n\|_1 + \lambda \sum_{n=0}^{N_c} \|\hat{\mathbf{c}}_n - \mathbf{c}_n\|_1 - 0.5 \sum_{l}^{L} \left(1 + \log \sigma_l^2 - \mu_l^2 - \sigma_l^2\right).$$

In implementation, the following cost function is used

$$\begin{aligned} \mathcal{L} &= \sum_{n=0}^{N} \|\hat{\mathbf{b}}_n - \mathbf{b}_n\|_1 + \lambda \sum_{n=0}^{N_c} \|\hat{\mathbf{c}}_n - \mathbf{c}_n\|_1 + (\eta - \lambda)\|\hat{\mathbf{c}}_0 - \mathbf{c}_0\|_1 \\ &= \sum_{n=0}^{N} \|\hat{\mathbf{b}}_n - \mathbf{b}_n\|_1 + \lambda \sum_{n=1}^{N_c} \|\hat{\mathbf{c}}_n - \mathbf{c}_n\|_1 + \eta\|\hat{\mathbf{c}}_0 - \mathbf{c}_0\|_1 \end{aligned} \tag{H.24}$$

where we replace the explicit KL divergence term in (H.24) by an implicit regularization term of $(\eta - \lambda)\|\hat{\mathbf{c}}_0 - \mathbf{c}_0\|_1$ with $\eta > \lambda$. Our motivation is to amplify the significance of the initial latent dynamic state $\mathbf{z}_0$ and its mapping to the physical coordinate space $\mathbf{c}_0$ in the loss function, rather than through the KL divergence term. The two hyperparameters λ and η play the tradeoff roles from the physical coordinate reconstruction $\sum_{n=1}^{N_c} \|\hat{\mathbf{c}}_n - \mathbf{c}_n\|_1$ and, respectively, the implicit regularization term $\sum_{n=0}^{N} \|\hat{\mathbf{c}}_0 - \mathbf{c}_0\|_1$ to the waveform (beam SNR) reconstruction error of $\sum_{n=0}^{N} \|\hat{\mathbf{b}}_n - \mathbf{b}_n\|_1$.

3.5 Complexity Analysis

We describe the time complexity of the proposed method by following Fig. G.2. Assuming the time complexity for the numerical ODE solvers is k at both encoder and decoder sides, the time complexity for the forward pass of the encoder (at the left side of Fig. G.2) can be approximated as $\mathcal{O}(N(k + L_h^2 + N_b L_h))$, where N is the number of time steps in the input sequence of beam SNRs and $(L_h^2 + N_b L_h)$ is from matrix products used in the LSTM update step in Appendix 8 with N_b is the input dimension and L_h the hidden state dimension.

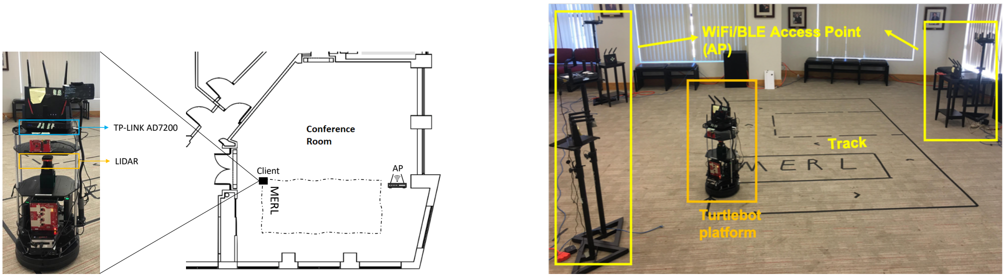
Then for the initial latent state z_0 of (H.13) using an MLP with one hidden layer of dimension M_z, the time complexity is approximately $\mathcal{O}(M_z(L_h + 2L))$, where $2L$ is the MLP output dimension. Following that, for the neural dynamic learning block on the middle right portion of Fig. G.2, the time complexity is approximately $\mathcal{O}(kN)$ and $\mathcal{O}(kN_c)$, respectively, for the two sets of queried time instances $\{t_n\}_{n=0}^{N}$ and $\{t_n^c\}_{n=0}^{N_c}$.

Finally, for the waveform decoder $\mathcal{M}_b$ of (H.18) using an MLP with one hidden layer of dimension M_b, the time complexity is $\mathcal{O}(N(LM_b + 2M_b + M_bN_b + N_b)) \approx \mathcal{O}(NM_b(L + N_b))$, where L is the latent dimension in the decoder and N_b is the output dimension of beam SNRs. Similarly for the coordinate decoders M_c of (H.18) using an MLP with one hidden layer of dimension M_c, its complexity is approximated as $\mathcal{O}(N_cM_c(L+2))$ where 2 is the dimension of the coordinate output. With all combined, the overall time complexity of the proposed method is approximately

$$\mathcal{O}(k(N + N_c) + M_z(L_h + 2L) + N_cM_c(L + 2) + N(L_h^2 + N_bL_h + M_b(L + N_b))). \tag{H.25}$$

4 mmWave Wi-Fi Testbed and Data Collection

To evaluate the proposed DDND framework and baseline comparison, we built a mmWave Wi-Fi testbed consisting of multiple commercial-of-the-shelf (COTS) 802.11ad devices to collect real-world mmWave Wi-Fi beam SNR data. Particularly, we used a pair of TP-Link AD7200 routers to acting as an AP fixed at a standing post and a mobile user; see the TurtleBot photo in the left plot of Fig. H.5 (a). The TurtleBot is equipped with a 2D scanning LiDAR sensor and a wheel encoder to map the environment and localize itself in a 2D floorplan with an accuracy of less than 1 cm. The 2D localization results are considered as the labels for training and groundtruth for test.



(a) A TurtleBot mobile user, trajectory configuration, and floorplan.

(b) A photo of data collection campaign.

Fig. H.5: An in-house mmWave Wi-Fi testbed with TP-Link AD7200 routers and a TurtleBot as a mobile user and a campaign of multi-day data collection in a conference room.

To access the raw beam SNR measurements from the COTS routers, we followed the methods described in [15] and utilized an open-source software package introduced in [45]. Specifically, we employed the Nexmon firmware patching framework [46], which allows the development of binary firmware extensions using the C programming language. By analyzing the patterns of IEEE 802.11ad beam training frames stored in the chip's memory, we were able to identify the firmware components responsible for handling these frames. Consequently, we extracted the beam SNR measurements from the corresponding memory addresses. For the TP-Link AD7200 router, an analog phased array of 32 antenna elements is used to sequentially scan over $N_b = 36$ predefined directional beampatterns for one air time for a given responder in Fig. G.1.

For data collection, we place the pair of TP-Link AD7200 routers in a corner conference room as shown in Fig. H.5 (b). The AP router is fixed at a standing post during the data collection, while the TurtleBot with the other router moves along a rectangular trajectory marked in a dash-dotted line in Fig. H.5 (a). In total, we collected two separate data sessions over two separate days with each data session lasting for multiple consecutive hours, resulting in:

- Day 1: $11.5K$ samples of beam SNR
- Day 2: $10K$ samples of beam SNR

with each beam SNR vector of dimension $N_b = 36$. The frame rate of the coordinate labels is 10 Hz and about $1 \sim 2$ Hz on average for the beam SNR measurements.

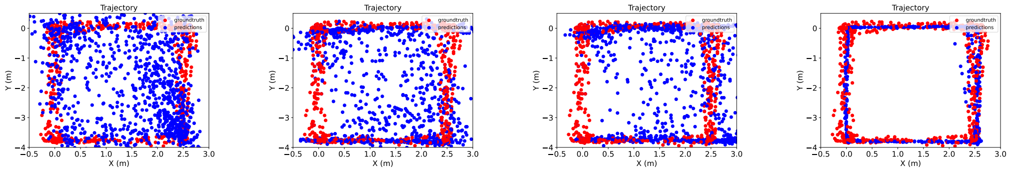
To preprocess the raw beam SNRs, we employ a sequence time window of $\Delta T_w = [2, 5, 8]$ seconds. This allows us to group the dataset into non-overlapping sequences, accommodating the irregular sampling of the beam SNR. The sequences are divided into training (0.8), validation (0.1), and test (0.1) sets, respectively. We standardize each entry $b_n, n = 1, \cdots, 36$ of the beam SNR by subtracting its mean and normalizing it with the corresponding standard deviation. Furthermore, we normalize the time vectors $\{t_n\}_{n=0}^{N}$ and $\{t_n^c\}_{n=0}^{N_c}$ within each ΔT_w seconds to the range $[0, 1]$.

5 Performance Evaluation

In the following, we evaluate the localization performance using the above datasets. We first quantify the performance against a list of baseline methods including both frame-based and sequence-based methods. Later on, we aim to provide a comprehensive ablation study of the proposed method against sequence length, supervision intensity (regular $N = N_c$ versus dense $(N < N_c)$) at the coordinate decoder, day-to-day performance generalization, and varying tasks (estimation versus extrapolation). Finally, we provide an interpretation of the learned latent dynamics.

5.1 Model Training

Our implementation of the proposed method is built using the PyTorch framework. The model incorporates a hidden state dimension of $L_h = 20$ and a latent dimension of $L = 20$. For the encoder and decoder, we utilize the 5th-order Runge-Kutta ODE solver. During training, we employ the Adamax optimizer with a learning rate of 0.01 and no weight decay. To form mini-batches, the model is trained using a batch size of 32 sequences. The proposed loss function performs an adaptive weighting strategy, to ensure stability during the training with respect



(a) (b) (c) (d)

Fig. H.6: Visualization of trajectory estimation over selected test sequences: (a) SVR (b) RNN-Decay (c) RNN-Δ (d) DDND-DS

to the amount of time samples. In this way, $\lambda = \frac{(N+1)}{\beta(N_c+1)}$, and $\eta = \alpha\lambda(N_c + 1)$ being $\alpha = 0.5$ and $\beta = 0.1$. For an exemplary case of $N = N_c = 4$, $\lambda = 10$ and $\eta = 25$.

5.2 Baseline Comparison

We here provide a performance comparison for the object trajectory estimation. We evaluate the following baseline methods and variants of our framework:

- Frame-based methods:
    1. Support Vector Regressor (SVR)
    2. Fully Connected Neural Network Regressor (FCNNR)
- Sequence-based methods (Section 2)
    1. RNN-Δ
    2. RNN-Decay
- Our DDND framework with variants:
    1. DDND-Regular Supervision + KL (DDND-RS + KL) of (H.24)
    2. DDND-Regular Supervision (DDND-RS) of (G.10)
    3. DDND-Dense Supervision + KL (DDND-DS + KL) of (H.24)
    4. DDND-Dense Supervision (DDND-DS) of (G.10)

For this baseline comparison, we set $\Delta T_w = 8$ seconds for all considered sequence-based methods.

Fig. H.6 illustrates the estimated trajectories over test sequences for the selected methods. In the frame-based SVR method (Fig. H.6 (a)), the coordinate estimates appear scattered within the square trajectory. However, there is a slight improvement observed in the sequence-based methods. In Fig. H.6 (b) and (c), which correspond to the RNN expdecay and RNN Δt methods, respectively, more trajectory estimates are pushing towards the square trajectory. A noticeable difference can be observed when comparing Fig. H.6 (d) to (a)-(c). It is evident that the proposed DDND-DS method is capable of learning the underlying dynamics, leading to more clustered trajectory estimates around the true square trajectory. The DDND-DS method demonstrates superior performance in capturing the trajectory patterns, indicating its effectiveness in trajectory estimation tasks.

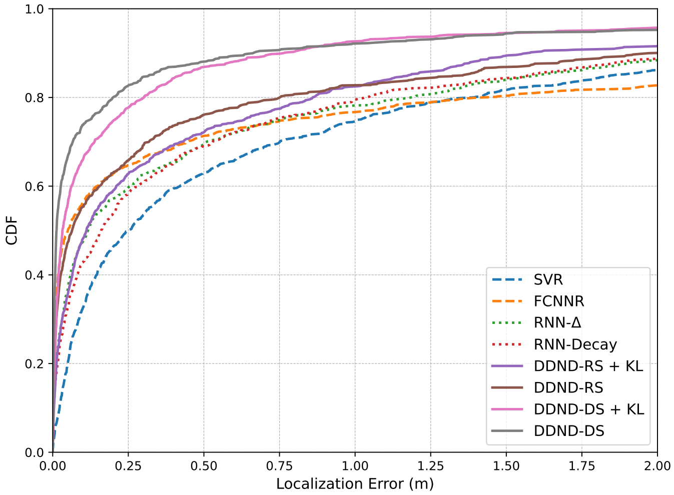


Fig. H.7: Cumulative distribution function (CDF) of localization errors.

Table H.1: Localization errors (m) on mmWave Wi-Fi dataset

	Mean	Median	CDF@0.9
SVR	0.82	0.25	2.64
FCNNR	0.92	0.05	3.04
RNN-Decay	0.74	0.16	2.19
RNN-Δ	0.76	0.11	2.36
DDND-RS + KL (ours)	0.58	0.11	1.57
DDND-RS (ours)	0.52	0.03	1.98
DDND-DS + KL (ours)	0.30	0.03	0.76
DDND-DS (ours)	**0.28**	**0.01**	**0.66**

Fig. H.7 presents a comparison of the baseline methods with the proposed solution in terms of the Cumulative Distribution Function (CDF) of root mean squared errors (RMSEs). The results highlight the impact of proper learning and modeling of latent dynamics and trajectory motion from irregularly-sampled Wi-Fi data on improving localization performance compared to frame-based and traditional sequence-based methods. The proposed method and its variants exhibit a notable reduction in large localization errors. These findings underscore the effectiveness of the proposed method in enhancing localization accuracy by capturing the underlying dynamics and motion patterns of the moving object based on irregularly-sampled Wi-Fi data. Table H.1 further quantifies the performance.

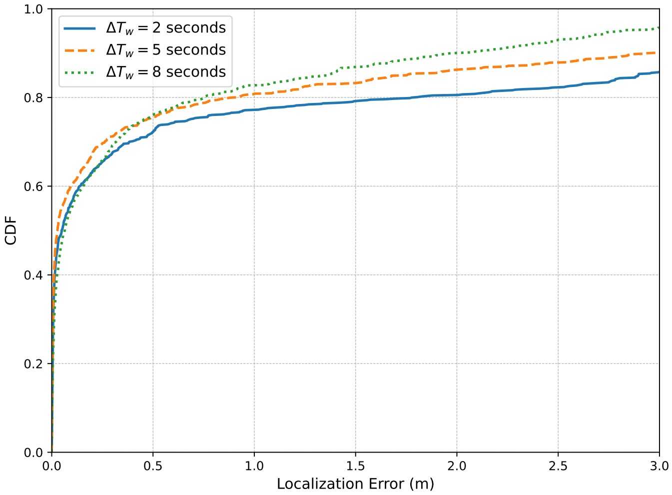


Fig. H.8: CDF curves of localization errors for different sequence lengths ΔT_w.

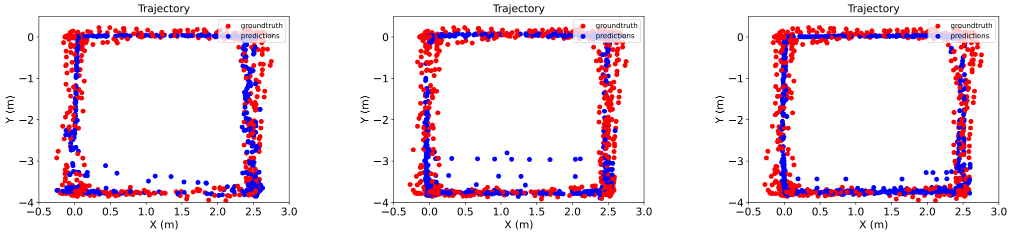
5.3 Impact of Sequence Length

In the following, we evaluate the impact of the sequence length ΔT_w on the trajection estimation performance.

Fig. H.8 show the CDF curves of the localization errors in terms of RMSE for different sequence lengths. It shows that although they exhibit similar performance in terms of median values or up to the 70-th percentile, the longer sequences are more suitable for learning the dynamics and present a lower variance in the results. Concretely, the sequence length of $\Delta T_w = 8$ seconds gives the best performance. Table H.2 further lists the quantitative localization performance for different sequence lengths ΔT_w.

Fig. H.9 shows a visual representation of the estimated trajectories over the test data for different configurations. It clearly shows our framework is capturing the dynamics of the motion of the square trajectory.

In summary, the presented results show the clear advantage of long-sequence learning for the trajectory estimation task. When using longer sequences, more information can be obtained leading to more complex dynamics that can be captured and learned in the latent



(a) (b) (c)

Fig. H.9: Visualization of trajectory estimation over selected test data: (a) $\Delta T_w = 2$s, (b) $\Delta T_w = 5$s, (c) $\Delta T_w = 8$s.

trajectory. This enables the model to better understand and predict future dynamics based on past information and translate into a significant increase in performance.

Table H.2: Localization errors (m) for different sequence lengths ΔT_w.

	Mean	Median	CDF@0.9
$\Delta T_w = 2$ sec	1.04	0.05	3.64
$\Delta T_w = 5$ sec	0.81	0.03	2.95
$\Delta T_w = 8$ sec	**0.52**	**0.03**	**1.98**

5.4 Comparison between Regular and Dense Supervision

We leverage the flexibility of the design of our framework to condition the learning dynamics in the latent space and we evaluate the trajectory estimation. Due to the intermittent sampling of the beam SNR measurements, result in irregular samples within a ΔT_w. However, we can perform stronger supervision to enhance the latent ODE dynamics, as we have the flexibility of conditioning the learning with whatever physical information we have. In our setup, the LiDAR has a fixed sampling frequency of 10 Hz that gives us way more coordinate points within a ΔT_w.

We perform this study again on $\Delta T_w = [2, 5, 8]$ seconds to assess the dense supervision enhancement with respect to the sequence length.

Fig. H.10 shows the localization errors for regular and dense supervision training. The results show when doing dense supervision the errors are decreased, incrementing the difference the longer the sequences due to the enhanced learning. In this way, we can see how dense supervision is beneficial. We further quantify this improvement by looking at Fig. H.11. It shows the enhancement in sequence length is bigger than in the previous case.

Fundamentally, dense supervision provides more accurate learning. Having more points within the sequence can provide a more accurate and detailed picture of the system dynamics.

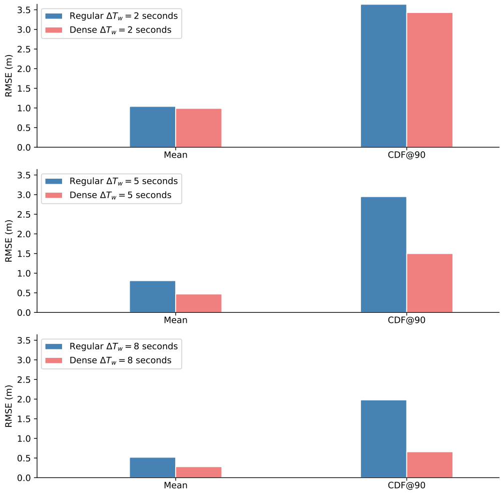


Fig. H.10: Regular vs dense supervision: quantitative performance comparison for different sequence lengths ΔT_w.

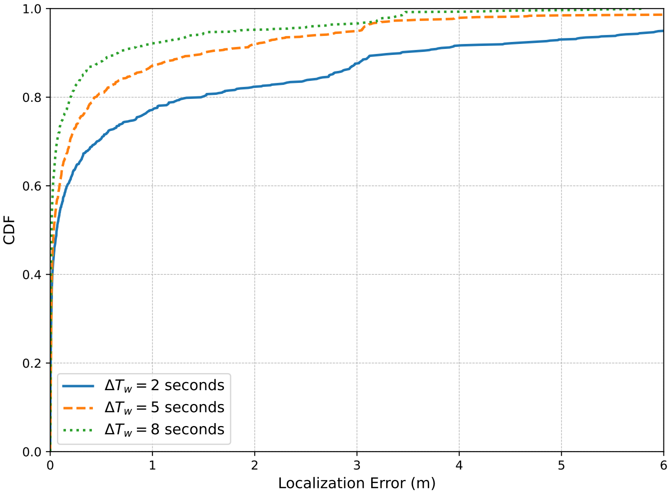


Fig. H.11: CDF curves of localization errors with dense supervision training.

The model can thus learn more accurately from the temporal dependencies, patterns, and nuances that might not be apparent or might be lost with fewer points. Also, more densely

Table H.3: Localization errors (m) for different ΔT_w seconds with dense supervision training.

	Mean	Median	CDF@0.9
Regular $\Delta T_w = 2$ sec	1.04	0.05	3.64
Regular $\Delta T_w = 5$ sec	0.81	0.03	2.95
Regular $\Delta T_w = 8$ sec	0.52	0.03	1.98
Dense $\Delta T_w = 2$ sec	0.99	0.06	3.43
Dense $\Delta T_w = 5$ sec	0.47	0.03	1.50
Dense $\Delta T_w = 8$ sec	**0.28**	**0.01**	**0.66**

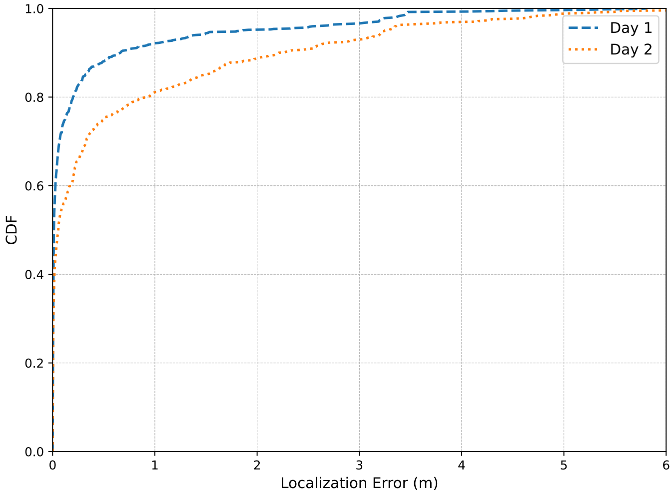


Fig. H.12: CDF curves of localization errors over separate days.

supervised sequences can enable the model to handle more complex systems and patterns.

5.5 Day-to-Day Generalization

We assess the generalization capabilities of the dense-trained model with a sequence length of $\Delta T_w = 8$ seconds by evaluating its performance on a different day of data collection. Specifically, we aim to test the model's ability to make accurate predictions when exposed to a new set of data collected on a different day. By conducting this evaluation, we can gain insights into the model's performance under varying environmental conditions. To accomplish this, we maintain the same model that was trained on the original dataset and evaluate its predictive performance on a new day's data. This allows us to analyze how well the model generalizes to unseen instances and adapts to changes in the environment. By examining the model's performance in this context, we can assess its robustness and effectiveness in real-world scenarios beyond the training data.

Fig. H.12 shows the CDF of localization error between the testing on the same day (day 1) and testing on a different day (day 2). Table H.4 further quantifies this performance. By looking at the results, we can see an acceptable performance regardless of the change in the environment. Also, if we compare to the results in Table H.3 we show the longer sequences are also beneficial for generalization, as even with the worsening in performance it does better than $\Delta T_w = 2$ or $\Delta T_w = 5$ seconds on the same day for the regular cases.

Table H.4: Localization errors (m) for $\Delta T_w = 8$ seconds with dense supervision training for two different days.

	Mean	Median	CDF@0.9
Day 1	**0.28**	**0.01**	**0.66**
Day 2	0.61	0.05	2.21

5.6 Extrapolation Performance

We aim to assess the flexibility of our decoder by evaluating its performance on an extrapolation task. To measure its effectiveness, we compare the performance of the model trained specifically for prediction and test it on extrapolation, against training the model directly for extrapolation. Additionally, we investigate the influence of different sequence lengths, specifically $\Delta T_w = [2, 5, 8]$ seconds, to examine their impact on the extrapolation results. By conducting this evaluation, we can gain insights into the decoder's ability to generalize beyond the observed data and generate accurate extrapolations. The comparison between prediction-trained and extrapolation-trained models, along with the analysis of different sequence lengths, will provide valuable information on the decoder's flexibility and performance in extrapolation tasks.

Fig. H.13 shows the CDF of the localization error for the different configurations. It shows when training prediction models and testing them for extrapolation they become worse the longer the sequence length. This shows it is difficult to perform extrapolation if just training in a prediction fashion, especially when the time window increases because the extrapolation task becomes harder. However, when we train for extrapolation, we have a huge improvement in the performance. This highlights the flexibility of our framework that can be trained for different specific tasks, thanks to the querying freedom due to the use of neural ODE on the decoder side. Table H.5 further quantifies this analysis. It shows our extrapolation method for the longest sequence exhibits quite good performance, comparable to dense prediction tasks.

Our framework models the latent dynamics for the input sequence of beam SNRs in a continuous-time fashion. This provides a more robust and accurate representation of the system dynamics, leading to an enhancement in extrapolation. Besides, the presented method continuity is well-suited to capture long-term dependencies in the data. This is particularly

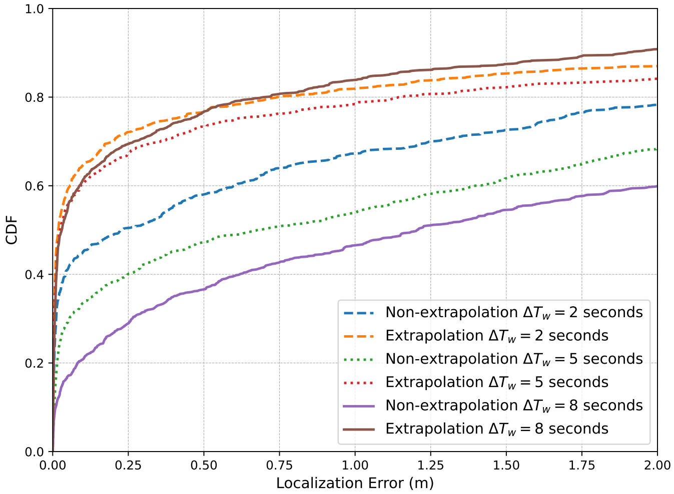


Fig. H.13: CDF curves of localization errors between trajectory estimation and extrapolation tasks.

beneficial for extrapolation tasks, where understanding long-term trends and dynamics is often crucial.

Table H.5: Localization errors (m) for extrapolation task.

	Mean	Median	CDF@0.9
Non-extrapolation $\Delta T_w = 2$ seconds	1.36	0.23	4.89
Extrapolation $\Delta T_w = 2$ seconds	0.82	0.02	3.02
Non-extrapolation $\Delta T_w = 5$ seconds	1.51	0.68	3.96
Extrapolation $\Delta T_w = 5$ seconds	0.72	0.02	2.84
Non-extrapolation $\Delta T_w = 8$ seconds	2.15	1.20	6.63
Extrapolation $\Delta T_w = 8$ seconds	**0.62**	**0.03**	**1.89**

5.7 Interpretation of Learned Latent Dynamics

Here we inspect the learned latent trajectories by plotting the first three dimensions of the latent space. In this case, we fix the sequence length as $\Delta T_w = 8$ seconds. The top row of Fig. H.14 shows estimated and ground truth trajectories for three sequences from the test

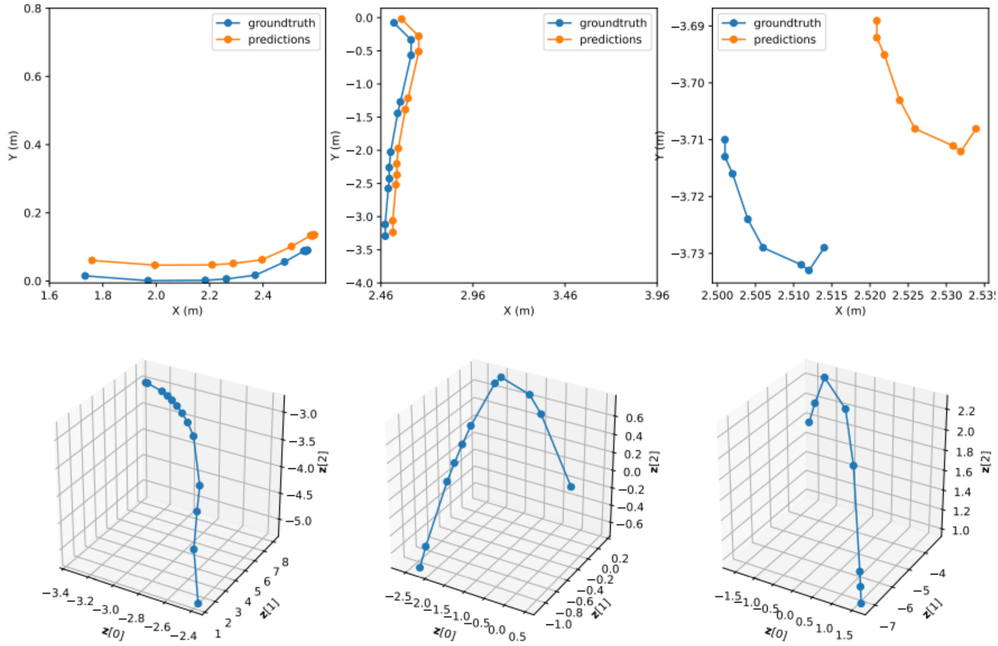


Fig. H.14: Visualization of the learned latent dynamics for the proposed method.

data. It is seen that the estimation results follow relatively well the object coordinates in various motion patterns. On the other hand, the bottom row of Fig. H.14 shows corresponding learned dynamics in the first three dimensions of the latent space at the decoder side, i.e., $\mathbf{z}(t) = \mathbf{z}_0 + \int_{\tau=t_0}^{t} \mathcal{O}_d(\mathbf{z}(\tau), \tau; \theta_l) d\tau$ of (H.15). Specifically, we sample an initial value for the latent dynamics $\mathbf{z}_0$ for each input sequence, and the learned latent dynamics are computed using the sampled $\mathbf{z}_0$ and the same learned latent ODE function $\mathcal{O}_d(\mathbf{z}(\tau), \tau; \theta_l)$ in (H.15). From the visualized results, it appears that the learned dynamics show sufficient capacity of modeling distinct patterns using the same continuous latent ODE functions.

6 Conclusion

This paper tackled the challenging task of localizing objects using intermittently sampled mmWave Wi-Fi beam training measurements. We proposed a novel solution called the dual-decoder neural dynamic framework. Through extensive performance comparisons, we demonstrated notable performance gains against baseline methods and provided a comprehensive study of the proposed method. Our method is directly compatible with upcoming Wi-Fi sensing standards, allowing seamless integration between communication and sensing of Wi-Fi devices in practical deployments.

7 Acknowledgements

The work of C.J. Vaca-Rubio was done during his visit and internship at MERL. He was also supported by European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie Grant agreement No. 813999, and in part by the European Union's Horizon EUROPE research and innovation program under grant agreement No. 101037090 - project CENTRIC.

8 Appendix: LSTM Update Step

Given the beam SNR $\mathbf{b}_n$ at time step n and the auxiliary variable $\mathbf{h}'_n$, one can use a standard LSTM unit to update the latent variable

$$\mathbf{h}_n = \mathcal{R}(\mathbf{h}'_n, \mathbf{b}_n; \theta), \quad n = 0, 1, \cdots, N, \tag{H.26}$$

where $\mathcal{R}(\cdot, \cdot|\theta)$ is implemented with the following process (with abuse of notation)

$$\tilde{\mathbf{c}}_n = \tanh\left(\mathbf{W}_{rc}\mathbf{b}_n + \mathbf{W}_{hc}\mathbf{h}'_n + \mathbf{b}_c\right), \tag{H.27}$$

$$\mathbf{f}_n = \sigma\left(\mathbf{W}_{rf}\mathbf{b}_n + \mathbf{W}_{hf}\mathbf{h}'_n + \mathbf{b}_f\right), \tag{H.28}$$

$$\mathbf{i}_n = \sigma\left(\mathbf{W}_{ri}\mathbf{b}_n + \mathbf{W}_{hi}\mathbf{h}'_n + \mathbf{b}_i\right). \tag{H.29}$$

The above process consists of three *gates*:

- a memory gate of (H.27) uses the tanh function to combine the auxiliary hidden state $\mathbf{h}'_n$ and the current input $\mathbf{b}_n$ into a value range of $(-1, 1)$.
- a forget gate of (H.28) also acts on $(\mathbf{h}'_n, \mathbf{b}_n)$ but compresses the value into $(0, 1)$ with the sigmoid function $\sigma(\cdot)$ to determine how much of the old memory should retain.
- an input gate of (H.29) compresses $(\mathbf{h}'_n, \mathbf{b}_n)$ into another value in between 0 and 1 and decides how much information we should take from the new input $\mathbf{b}_n$,

along with weight matrices $\mathbf{W}_{rc/rf/ri/hc/hf/hi}$ and bias terms $\mathbf{b}_{c/f/i}$. Then new hidden state $\mathbf{h}_n$ in (H.26) is updated as

$$\mathbf{h}_n = \tanh\left(\hat{\mathbf{c}}_n\right) \odot \mathbf{o}_n, \tag{H.30}$$

where the new memory variable $\hat{\mathbf{c}}_n$ updates its "old" memory $\hat{\mathbf{c}}_{n-1}$ passing through the "current" forget gate output $\mathbf{f}_n$ and adds new memory cell $\tilde{\mathbf{c}}_n$ weighted by the "current" input gate output $\mathbf{i}_n$:

$$\hat{\mathbf{c}}_n = \mathbf{f}_n \odot \hat{\mathbf{c}}_{n-1} + \mathbf{i}_n \odot \tilde{\mathbf{c}}_n, \tag{H.31}$$

and the output gate $\mathbf{o}_n$ is computed as

$$\mathbf{o}_n = \sigma\left(\mathbf{W}_{ro}\mathbf{b}_n + \mathbf{W}_{ho}\mathbf{h}'_n + \mathbf{W}_{co} \odot \hat{\mathbf{c}}_n + \mathbf{b}_o\right). \tag{H.32}$$

It is seen that the parameters θ in the LSTM update step is given as
$\theta = \{\mathbf{W}_{rc/rf/ri/hc/hf/hi/ro/ho/co}, \mathbf{b}_{c/f/i/o}\}$.

References

[1] C. J. Vaca-Rubio, P. Wang, T. Koike-Akino, Y. Wang, P. Boufounos, and P. Popovski, "mmWave Wi-Fi Trajectory Estimation with Continuous-Time Neural Dynamic Learning," in *ICASSP 2023*, 2023, pp. 1-5.

[2] M. Youssef and A. Agrawala, "The Horus Location Determination System," *Wireless Networks*, vol. 14, no. 3, pp. 357-374, June 2008.

[3] Z.-L. Wu, C.-H. Li, J. K.-Y. Ng, and K. Leung, "Location Estimation via Support Vector Regression," *IEEE Transactions on Mobile Computing*, vol. 6, no. 3, pp. 311-321, 2007.

[4] K. Wu, J. Xiao, Y. Yi, D. Chen, X. Luo, and L. M. Ni, "CSI-based indoor localization," *IEEE Transactions on Parallel and Distributed Systems*, vol. 24, no. 7, pp. 1300-1309, 2013.

[5] M. T. Hoang, B. Yuen, X. Dong, T. Lu, R. Westendorp, and K. Reddy, "Recurrent Neural Networks for Accurate RSSI Indoor Localization," *IEEE Internet of Things Journal*, vol. 6, no. 6, pp. 10639-10651, Dec. 2019.

[6] X. Wang, L. Gao, S. Mao, and S. Pandey, "DeepFi: Deep learning for indoor fingerprinting using channel state information," in *WCNC*, March 2015, pp. 1666-1671.

[7] X. Wang, L. Gao, S. Mao, and S. Pandey, "CSI-Based Fingerprinting for Indoor Localization: A Deep Learning Approach," *IEEE Transactions on Vehicular Technology*, vol. 66, no. 1, pp. 763-776, Jan. 2017.

[8] H. Chen, Y. Zhang, W. Li, X. Tao, and P. Zhang, "ConFi: Convolutional Neural Networks Based Indoor Wi-Fi Localization Using Channel State Information," *IEEE Access*, vol. 5, pp. 18066-18074, 2017.

[9] J. Ding and Y. Wang, "WiFi CSI-Based Human Activity Recognition Using Deep Recurrent Neural Network," *IEEE Access*, vol. 7, pp. 174257-174269, 2019.

[10] M. T. Hoang et al., "A CNN-LSTM quantifier for single access point CSI indoor localization," *arXiv preprint arXiv:2005.06394*, 2020.

[11] H. Sun et al., "WiFi based fingerprinting positioning based on Seq2seq model," *Sensors*, vol. 20, no. 13, p. 3767, 2020.

[12] J. Yu, H. M. Saad, and R. M. Buehrer, "Centimeter-Level Indoor Localization using Channel State Information with Recurrent Neural Networks," in *IEEE/ION Position, Location and Navigation Symposium (PLANS)*, 2020, pp. 1317–1323.

[13] H. Xia, P. Wang, T. Koike-Akino, Y. Wang, P. V. Orlik, and Z. Ding, "Adversarial Bi-Regressor Network for Domain Adaptive Regression," in *International Joint Conference on Artificial Intelligence (IJCAI)*, 2022, pp. 3608-3614.

[14] J. Yu, P. Wang, T. Koike-Akino, and P. V. Orlik, "Multi-Modal Recurrent Fusion for Indoor Localization," in *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2022.

[15] G. Bielsa, J. Palacios, A. Loch, D. Steinmetzer, P. Casari, and J. Widmer, "Indoor localization using commercial off-the-shelf 60 GHz access points," in *IEEE INFOCOM*, 2018, pp. 2384-2392.

[16] M. Pajovic, P. Wang, T. Koike-Akino, H. Sun, and P. V. Orlik, "Fingerprinting-Based Indoor Localization with Commercial MMWave WiFi–Part I: RSS and Beam Indices," in *GLOBECOM*, Dec. 2019.

[17] P. Wang, M. Pajovic, T. Koike-Akino, H. Sun, and P.V. Orlik, "Fingerprinting-Based Indoor Localization with Commercial MMWave WiFi–Part II: Spatial Beam SNRs," in *GLOBECOM*, Dec 2019.

[18] T. Koike-Akino, P. Wang, M. Pajovic, H. Sun, and P. V. Orlik, "Fingerprinting-Based Indoor Localization With Commercial MMWave WiFi: A Deep Learning Approach," *IEEE Access*, vol. 8, pp. 84879-84892, 2020.

[19] P. Wang, T. Koike-Akino, and P. V. Orlik, "Fingerprinting-Based Indoor Localization with Commercial MMWave WiFi: NLOS Propagation," in *GLOBECOM*, Dec. 2020.

[20] J. Yu, P. Wang, T. Koike-Akino, and P. V. Orlik, "Human Pose and Seat Occupancy Classification with Commercial MMWave WiFi," in *GLOBECOM Workshop on Integrated Sensing and Communication (ISAC)*, Dec. 2020.

[21] J. Yu, P. Wang, T. Koike-Akino, Y. Wang, P. V. Orlik, and R. M. Buehrer, "Multi-Band Wi-Fi Sensing with Matched Feature Granularity," *IEEE Internet of Things Journal*, vol. 1, pp. 1-1, 2022.

[22] A. Blanco, P. J. Mateo, F. Gringoli, and J. Widmer, "Augmenting mmWave localization accuracy through sub-6 GHz on off-the-shelf devices," in *Proceedings of the 20th Annual International Conference on Mobile Systems, Applications and Services*, 2022, pp. 477–490.

[23] T. Koike-Akino, P. Wang, and Y. Wang, "Quantum Transfer Learning for Wi-Fi Sensing," in *IEEE International Conference on Communications (ICC)*, May 2022.

[24] S. Mazuelas, A. Bahillo, R. M. Lorenzo, P. Fernandez, F. A. Lago, E. Garcia, J. Blas, and E. J. Abril, "Robust Indoor Positioning Provided by Real-Time RSSI Values in Unmodified WLAN Networks," *IEEE Journal of Selected Topics in Signal Processing*, vol. 3, no. 5, pp. 821-831, 2009.

[25] X. Wang, L. Gao, and S. Mao, "CSI Phase Fingerprinting for Indoor Localization With a Deep Learning Approach," *IEEE Internet of Things Journal*, vol. 3, no. 6, pp. 1113-1123, Dec. 2016.

[26] X. Wang, L. Gao, and S. Mao, "BiLoc: Bi-Modal Deep Learning for Indoor Localization With Commodity 5GHz WiFi," *IEEE Access*, vol. 5, pp. 4209-4220, 2017.

[27] C. Hsieh, J. Chen, and B. Nien, "Deep Learning-Based Indoor Localization Using Received Signal Strength and Channel State Information," *IEEE Access*, vol. 7, pp. 33256-33267, 2019.

[28] J. Wang and J. G. Park, "An Enhanced Indoor Ranging Method Using CSI Measurements with Extended Kalman Filter," in *PLANS*, 2020, pp. 1342-1348.

[29] A. S. Paul and E. A. Wan, "Wi-Fi based indoor localization and tracking using sigma-point Kalman filtering methods," in *PLANS*, 2008, pp. 646-659.

[30] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," *Neural Computation*, vol. 9, no. 8, pp. 1735-1780, Nov. 1997.

[31] R. Chen et al., "Neural ordinary differential equations," *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 31, 2018.

[32] Y. Rubanova, R. Chen, and D. K. Duvenaud, "Latent Ordinary Differential Equations for Irregularly-Sampled Time Series," in *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 32, 2019.

[33] J. Kelly et al., "Learning differential equations that are easy to solve," in *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 33, 2020, pp. 4370–4380.

[34] C. Finlay et al., "How to train your neural ODE: the world of Jacobian and kinetic regularization," in *International Conference on Machine Learning (ICML)*, 2020, pp. 3154–3164.

[35] A. Zhu et al., "On numerical integration in neural ordinary differential equations," in *International Conference on Machine Learning (ICML)*, 2022, pp. 27527–27547.

[36] H. H. N. Nguyen et al., "Improving Neural Ordinary Differential Equations with Nesterov's Accelerated Gradient Method," in *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 35, 2022, pp. 7712–7726.

[37] C. Challu et al., "NHITS: Neural Hierarchical Interpolation for Time Series Forecasting," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 37, no. 6, pp. 6989–6997, 2023.

[38] G. M. Mendoza-Silva et al., "Environment-aware regression for indoor localization based on WiFi fingerprinting," *IEEE Sensors Journal*, vol. 22, no. 6, pp. 4978–4988, 2021.

[39] W. Sun et al., "Augmentation of fingerprints for indoor WiFi localization based on Gaussian process regression," *IEEE Transactions on Vehicular Technology*, vol. 67, no. 11, pp. 10896–10905, 2018.

[40] M. Nowicki and J. Wietrzykowski, "Low-effort place recognition with WiFi fingerprints using deep learning," in *Automation 2017: Innovations in Automation, Robotics and Measurement Techniques*, 2017, pp. 575–584.

[41] R. Zhou et al., "Device-free localization based on CSI fingerprints and deep neural networks," in *2018 15th Annual IEEE International Conference on Sensing, Communication, and Networking (SECON)*, 2018, pp. 1–9.

[42] X. Song et al., "A novel convolutional neural network based indoor localization framework with WiFi fingerprinting," *IEEE Access*, vol. 7, pp. 110698–110709, 2019.

[43] X. Wang, X. Wang, and S. Mao, "Deep convolutional neural networks for indoor localization with CSI images," *IEEE Transactions on Network Science and Engineering*, vol. 7, no. 1, pp. 316–327, 2018.

[44] D. P. Kingma and M. Welling, "Auto-encoding variational Bayes," *arXiv preprint arXiv:1312.6114*, 2013.

[45] D. Steinmetzer, D. Wegemer, and M. Hollick, "Talon tools: The framework for practical IEEE 802.11ad research," 2017.

[46] M. Schulz, D. Wegemer, and M. Hollick, "Nexmon: The C-based firmware patching framework," *Res. Gate*, 2017.

Paper I

Proximal policy optimization for integrated sensing and communication in mmWave systems

Cristian J Vaca-Rubio, Carles Navarro Manchón, Ramoni Adeogun, Petar Popovski

The paper has been submitted in the
IEEE Wireless Communication Letters (WCL), 2023.

© 2023 IEEE

The layout has been revised.

Abstract

In wireless communication systems, mmWave beam tracking is a critical task that affects both sensing and communications, as it is related to the knowledge of the wireless channel. We consider a setup in which a Base Station (BS) needs to dynamically choose whether the resource will be allocated for one of the three operations: sensing (beam tracking), downlink, or uplink transmission. We devise an approach based on the Proximal Policy Optimization (PPO) algorithm for choosing the resource allocation and beam tracking at a given time slot. The proposed framework takes into account the variable quality of the wireless channel and optimizes the decisions in a coordinated manner. Simulation results demonstrate that the proposed method achieves significant performance improvements in terms of average packet error rate (PER) compared to the baseline methods while providing a significant reduction in beam tracking overhead. We also show that our proposed PPO-based framework provides an effective solution to the resource allocation problem in beam tracking and communication, exhibiting a great generalization performance regardless of the stochastic behavior of the system.

Keywords— mmWave, sensing, beam tracking, resource allocation.

1 Introduction

Beamforming and resource allocation methods have significantly improved as a result of the rising demand for fast and dependable wireless communication networks. A crucial task in wireless communication systems is beam alignment, which entails directing antenna arrays to maximize throughput and improve signal-to-noise ratio (SNR). Traditional beam sweeping in 5G allocates sensing resources for all users at fixed-slots periodicity. This might lead to inefficient (low-mobility scenarios) and insufficient (high-mobility) to track the necessity of beam changes. These operations are not typically adapted at short time-scales [1], and the division between uplink (UL) and downlink (DL) slots is usually kept fixed over long time frames. To address this, we aim to find a way to decide the allocation of resources between sensing, uplink, and downlink transmissions that is adaptive on a short time-scale according to the particular network situation.

In recent years, beam alignment issues in wireless networks have been successfully resolved using reinforcement learning (RL) techniques [2, 3], as well as pure deep learning (DL) solutions [4, 5]. The latter rely on the one-shot interaction with the channel, making it more prone to generalization issues, while the former do not consider the joint problem of resource allocation and beam alignment. Similarly, resource allocation is a critical task in wireless communication systems, and it has been traditionally formulated as an optimization problem [6]. However, the increasing complexity of wireless networks requires techniques beyond traditional optimization, such as Reinforcement learning (RL). For instance, RL-based methods have been proposed for joint power and radio resource allocation in 5G networks [6, 7].

Motivated by the encouraging findings of previous works, we propose a Proximal Policy Optimization (PPO) [8] solution for joint resource allocation and beam tracking in a mmWave wireless communication context. By coordinating the optimization of the resource allocation and beam tracking parameters, our system intends to mitigate the wireless channel's dynamic nature and its fluctuating channel quality. Simulation results show that, when compared to

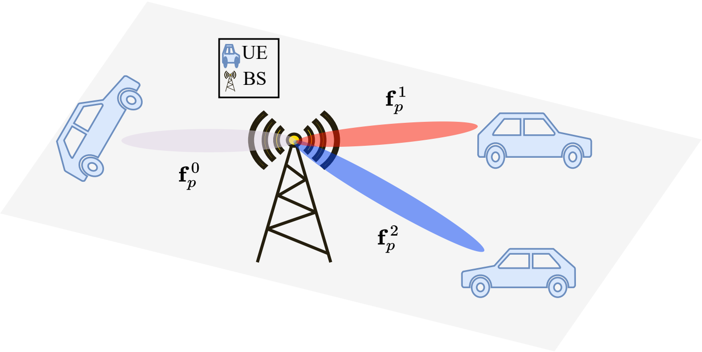


Fig. I.1: Illustration of the scenario

baseline approaches, the suggested method significantly outperforms them in terms of average packet error rate (PER), learning a more dynamic policy of slots allocation for efficient beam tracking and data transmission.

2 System Model and Problem Formulation

Consider a vehicular wireless network as depicted in Figure I.1, using a time division duplex (TDD) scheme. A mmWaveBase Station (BS) equipped with a uniform linear array (ULA) with M_t transmitting antennas communicates with U single-antenna User Equipment (UEs) in a time-slotted network with $\mathcal{K} = \{1, ..., K\}$ orthogonal channel uses per frame. In each time slot, only one user u can be scheduled. Each user u has a buffer of size B_{UL}^u for uplink packets, while the base station has U buffers (with size B_{DL}^u for the user u) for downlink traffic. At each slot $k \in \mathcal{K}$, the BS performs one action: S_u (sensing/beam tracking), DL_u (transmission of downlink packet), or UL_u (reception of uplink packet) with $u = 1, ..., U$.

2.1 Channel Model

Due to the high directivity of mmWave systems, we assume a geometrical Line-of-Sight (LoS) channel model for this work. In this way, the narrowband channel for the user u is given by

$$\mathbf{h}_u = \frac{\sqrt{M_t}}{d_u}\beta\mathbf{a}_t^\dagger(\theta_u), \tag{I.1}$$

where † denotes conjugate transposition, $\mathbf{h}_u \in \mathbb{C}^{M_t}$ denotes the channel vector for user u, d_u denotes the distance between the BS and the $u-$th user and $\beta \sim \mathcal{CN}(0, \sigma_u^2)$ is the complex fading gain with variance σ_u^2. Also, θ_u denotes the Angle of Departure (AoD) with respect to the BS array axis and the u user position. As we are considering a ULA BS, the ideal isotropic array responses are given by

$$\mathbf{a}_t(\theta_u) = \frac{1}{\sqrt{M_t}}\left[1, e^{-j\pi\cos(\theta_u)}, ..., e^{-j\pi(M_t-1)\cos(\theta_u)}\right]^T. \tag{I.2}$$

For the dynamics of the system [3], given the user position in a given time slot $\mathbf{c}_k^u = [x_k^u, y_k^u]$, its velocity $\mathbf{v} = [v_{\text{linear}}, v_{\text{angular}}]$, where $v_{\text{linear}} \sim \text{Exp}(1)$ and $v_{\text{angular}} \sim \mathcal{N}(0,1)$ and a slot duration of Δt, the user new position can be calculated as follows: *i)* compute the angular displacement, $\omega = v_{\text{angular}}\Delta t$, *ii)* compute the direction vector: $\mathbf{d} = [\cos\omega, \sin\omega]$, *iii)* compute the linear displacement: $\boldsymbol{\Delta}\mathbf{c} = v_{\text{linear}}\mathbf{d}\Delta t$, and *iv)* compute the user's new position: $\mathbf{c}_{k+1}^u = \mathbf{c}_k^u + \boldsymbol{\Delta}\mathbf{c}$. This will change the AoDs and consequently make the channel vary along our time-slotted resources. For simplicity, we assume a 2D geometric layout.

2.2 Beam Codebook

In this work, we assume a codebook-based analog beamforming architecture to beamform signals with a single RF chain at the BS. We denote by $\mathcal{F} = \{\mathbf{f}_1, ..., \mathbf{f}_{M_t}\}$ the codebook used for analog beamforming at the BS, with M_t beams. We use the common Discrete Fourier Transform (DFT)-based codebooks [9], with precoders $\mathbf{f}_i$ given by

$$\mathbf{f}_i = \frac{1}{\sqrt{M_t}}\left[1, e^{-j\pi\frac{2i-1-M_t}{M_t}}, ..., e^{-j\pi(M_t-1)\frac{2i-1-M_t}{M_t}}\right]^T, \tag{I.3}$$

where $i \in \{1, ..., M_t\}$. Then on each time slot, given the selected precoder $\mathbf{f}_i^u \in \mathcal{F}$, the received signal power $R_i^u \in \mathbb{R}$ for the u-th user can be described as

$$R_i^u = |\sqrt{P_t}\mathbf{h}_u^T\mathbf{f}_i^u s + \mathbf{n}_u|^2, \tag{I.4}$$

where T denotes transposition, P_t, $s \in \mathbb{C}$, and $\mathbf{n}_u$ denote the transmission power, the known training symbol with normalized power, and the zero mean complex Gaussian noise vector with variance σ_n^2.

2.3 Traffic model

Each user follows an independent Bernoulli process with probability P^u for generating packets in downlink and uplink in every time slot k.

$$[p_{dl}^u, p_{ul}^u] \sim \text{Bernoulli}(P^u), \tag{I.5}$$

where p_{dl}^u, p_{ul}^u denote 0 or 1 to determine if a packet was generated for the user u in downlink and uplink for a time slot k.

2.4 Optimization Problem

We perform the allocation of slots for a specific user u with the goal of minimizing the average PER, defined as:

$$PER = \frac{LostPackets}{TotalPackets} \tag{I.6}$$

within an episode $\mathcal{K}$. The optimization problem is formulated as follows:

$$\begin{aligned} \min_{k \in \mathcal{K}} \quad & \frac{1}{UK} \sum_{u=1}^{U} \sum_{k=1}^{K} PER_{uk} \\ \text{s. t.} \quad & \sum_{u=1}^{U} s_{uk} + \sum_{u=1}^{U} d_{uk} + \sum_{u=1}^{U} u_{uk} = 1, \quad \forall k = 1, 2, ..., K \\ & s_{uk}, d_{uk}, u_{uk} \in \{0, 1\}, \quad \forall u = 1, 2, ..., U, \ k = 1, 2, ..., K \end{aligned} \tag{I.7}$$

where the constraints denote that just one user can be allocated in every slot k for sensing s_{uk}, downlink d_{uk}or uplink u_{uk}, respectively.

3 PPO Fundamentals

PPO is an actor-critic algorithm [8] that models both the policy and value functions as a neural network, with parameters denoted as θ, and ϕ. It aims to learn a policy $\pi_\theta(a|s)$ that maximizes the expected cumulative reward in an environment. The algorithm utilizes a value function $V_\phi(s)$ to estimate the expected total reward from a given state s, and an advantage function A_k to measure the quality of an action in a specific state. At each time step k, the agent observes the current state s_k of the environment and samples an action a_k from the policy distribution $\pi_\theta(a_k|s_k)$. The action is then executed, resulting in a reward r_k and a new state s_{k+1}. The agent stores these experiences to update its policy and value functions after accumulating N tuples of experience for a given policy. From here, we will denote the equations for a single time step k for simplicity. The advantage function A_k measures the advantage of taking action a_k in state s_k compared to the expected value from the current state. It is computed as the sum of the discounted future rewards minus the value function at the current state:

$$A_k = \sum_{i=k}^{N} \gamma^{i-k} r_i - V_\phi(s_k), \tag{I.8}$$

where N is the maximum number of time steps per experience memory. Next, the value function $V_\phi(s_k)$ estimates the expected cumulative reward from the current state s_k onwards. It is updated by minimizing the mean squared error between the estimated value and the target value:

$$L^{(k)}_{\text{critic}}(\phi) = \frac{1}{2} \left(V_\phi(s_k) - (r_k + \gamma V_\phi(s_{k+1})) \right)^2, \tag{I.9}$$

where γ is the discount factor that balances immediate and future rewards.

The policy is updated using the PPO objective, which aims to maximize the expected advantage while avoiding large policy changes. The PPO loss function for a single time step is given by:

$$L^{(k)}_{\text{PPO}}(\theta) = \min \left(r_k(\theta) A_k, \text{clip}(r_k(\theta), 1 - \epsilon, 1 + \epsilon) A_k \right), \tag{I.10}$$

where the clip operation restrict the values from $1 - \epsilon$ to $1 + \epsilon$, $r_k(\theta) = \frac{\pi_\theta(a_k|s_k)}{\pi_{\text{old}}(a_k|s_k)}$ is the ratio of the updated policy probabilities to the old policy probabilities, and ϵ is a hyperparameter that controls the magnitude of the policy change. To further improve the stability of training, an entropy regularization term is included. The entropy $S[\pi_\theta(a_k|s_k)]$ measures the uncertainty or randomness in the policy distribution:

$$S[\pi_\theta(a_k|s_k)] = -\sum_a \pi_\theta(a_k|s_k) \log \pi_\theta(a_k|s_k). \tag{I.11}$$

This term encourages exploration by discouraging overly deterministic policies. Finally, the total loss function for PPO combines the policy loss, value function loss, and entropy regularization over the entire N experiences:

$$L_{\text{total}}(\theta, \phi) = \frac{1}{N} \sum_{k=1}^{N} \left(L_{\text{PPO}}^{(k)}(\theta) - c_1 L_{\text{critic}}^{(k)}(\phi) + c_2 S[\pi_\theta(a_k|s_k)] \right) \tag{I.12}$$

where c_1 and c_2 are hyperparameters that control the trade-off between the value function loss and the entropy regularization. The policy and value networks are then updated jointly by minimizing the total loss.

4 PPO for joint beam alignment and resource allocation

To solve the problem in equation (7), the PPO framework is used, with the BS as the agent. To do so, the environment, state space, action space, and reward signal are defined. Algorithm 1 provides a summary of the training procedure.

4.1 State space

The state space of this environment is characterized by four parameters:

- Current beam indexes $i^u \in \mathcal{I}$ for the $u \in U$ users.
- Current amount of packets of downlink buffer P_{DL}^u for the $u \in U$ users.
- Current amount of packets of uplink buffers P_{UL}^u for the $u \in U$ users.
- Current received power for every selected beam for the $u \in U$ users, R_p^u.

In this way, the state in time slot k can be represented as

$$\begin{aligned} \mathcal{S}_k = \{&\{i^1, \ldots, i^U\}, \{P_{DL}^1, \ldots, P_{DL}^U\}, \\ &\{P_{UL}^1, \ldots, P_{UL}^U\}, \{R_p^1, \ldots, R_p^U\}\}, \end{aligned} \tag{I.13}$$

We normalize the beam indexes and buffer values dividing by the number of total beams in the codebook and the buffer sizes, respectively.

4.2 Action space

The action space is the set of all possible actions that the BS agent can choose from at each time slot. While the method presented here can be applied to the selection of any wireless resource, we consider the allocation of time slots. In this way, the possible actions are a discrete variable with $U \times 3$ possible values

$$a_k \in \mathcal{A} = \{(a, u), a = 0, 1, 2, u = 1, 2, ..., U\}, \tag{I.14}$$

with the first element a denoting the action (0 for sensing, 1 for UL transmission, 2 for DL transmission) and the second element u denoting the target user. Once in a sensing slot s_{uk}, we assume we select the best neighboring beam (beam tracking) such that

$$\underset{\mathbf{f}_i^u \in \{\mathbf{f}_{i-1}^u, \mathbf{f}_i^u, \mathbf{f}_{i+1}^u\}}{\arg\max} |\sqrt{P_t}\mathbf{h}_u^T \mathbf{f}_i^u s + \mathbf{n}_u|^2. \tag{I.15}$$

If the sensing variable s_{uk} is active (i.e., $s_{uk} = 1$), the beam selection for either downlink transmission (d_{uk}) or uplink transmission (u_{uk}) in slots will be based on the beam selected during the last sensing slot.

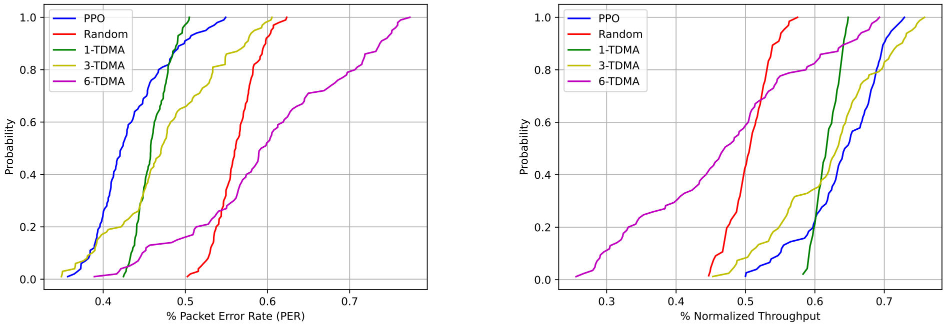
4.3 Reward signal

The objective stated in equation (7) is to minimize the average PER. In our problem, packets can be dropped due to different reasons: either encountering a full buffer ($P_{DL}^u > B_{DL}^u$ or $P_{UL}^u > B_{UL}^u$), or transmitting with a poor quality beam (i.e., using an outdated beam $\mathbf{f}_i^u$ obtained from equation (I.15) during s_{uk}). A beam is considered outdated, if $\mathbf{f}_i^u$ is no longer the optimal beam. To encourage the reduction of these packet drops, we assign a reward in each time slot k based on the selected action. To understand our reward, let's first define $G_b^u(k)$ as an indicator that equals 1 if, at the end of time step k, the buffer size P_b^U for user u (where b can be DL or UL) is such that $P_b^U < B_b^u$. This indicates that the buffer is not full. Additionally, we use the factor $\rho(k)$ to represent the impact of beam tracking on the beam. When beam tracking is performed and it results in a change to the beam, $\rho(k)$ is greater than zero ($\rho(k) > 0$). Conversely, when beam tracking is unnecessary and the beam remains unchanged, $\rho(k)$ is less than zero ($\rho(k) < 0$). The purpose of $\rho(k)$ is to encourage the use of sensing to recover the best beam when it is beneficial, and to penalize unnecessary sensing. If it is not a sensing slot, $\rho(k)$ is set to zero ($\rho(k) = 0$). Finally, we denote $D(k)$ as the amount of packets dropped in a time slot k. Furthermore, the reward r_k is defined as

$$r_k = \frac{1}{U \times 2} \sum_{u=1}^{U} \sum_{b \in \{DL, UL\}} G_b^u(k) + \rho(k) - D(k). \tag{I.16}$$

5 Simulation environment and evaluation description

The algorithm is trained for a fixed number of episodes E_{train}, with some randomly selected initial positions for the UEs with varying speeds, small variations, and arrived packet probabilities for each user. The results are evaluated on E_{test} randomly initialized episodes to



(a) ECDF for PER **(b)** ECDF for Throughput

Fig. I.2: ECDFs over E_{test} episodes

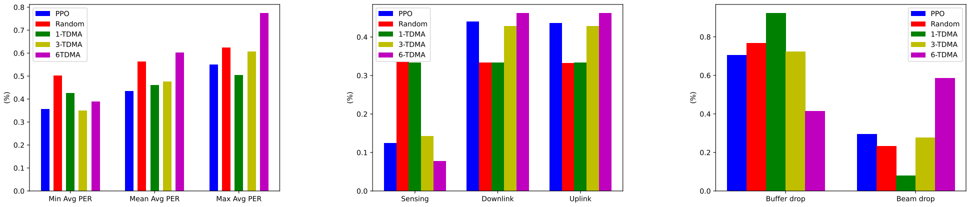
demonstrate generalization capability. Simulation and PPO parameters are listed in Tables I and II, respectively. The baselines used to evaluate the results are described below:

1. Random: The policy of the agent is to allocate slots uniformly at random among all possible actions.
2. X-TDMA (X Time Division Multiple Access) is a slot allocation policy where slots are divided into sensing (S), uplink (UL), and downlink (DL) categories. The value of X represents the number of consecutive UL/DL slots before transitioning to the next sensing slot. Specifically:
   (a) For X = 1: The pattern is S-UL-DL, repeating cyclically for each user.
   (b) For X = 3: The pattern is S-UL-DL-UL-DL-UL-DL, repeating cyclically for each user
   (c) For X = 6: The pattern is S-UL-DL-UL-DL-UL-DL-UL-DL-UL-DL-UL-DL, repeating cyclically for each user.

   In each case, X-TDMA follows TDMA principles and allows users to take turns using the channel resources for uplink and downlink communication. These patterns are allocated for every user u.

6 Results

We compare our proposed solution with the baselines in terms of average PER and normalized throughput across the evaluation episodes (E_{test}). Figure I.2 presents the empirical cumulative distribution function (ECDF) of the average PER and normalized throughput. Our method demonstrates effective policy learning compared to random allocation. It outperforms the



(a) PER statistics over E_{test} episodes

(b) Slots distribution over E_{test} episodes

(c) Drops distribution over E_{test} episodes

baselines in terms of PER, even though 1-TDMA is the safest option on average, presenting a lower variance in the results. When reducing the number of sensing slots (3/6-TDMA), the variance of results increases significantly. Moreover, our method exhibits improved throughput performance. To provide a better understanding of our solution's effectiveness, Figure I.3a illustrates the maximum, minimum, and average PER. The allocation policy of our method efficiently reduces the number of required sensing slots, as depicted in Figure I.3b. Despite this reduction, our method achieves lower PER compared to the baselines, as observed in Figures I.2 and I.3a. Furthermore, we analyze the efficiency in Figure I.3c, which reveals that our dynamic allocation policy prioritizes transmitting packets to reduce the full buffer drop rate, rather than increasing sensing to minimize total PER but not as much as 6-TDMA does. In comparison to 1/3-TDMA, the dynamic allocation of transmissions provides flexibility to improve overall performance, as evidenced by the average PER results.

7 Conclusion

The proposed approach for joint resource allocation and beam alignment in mmWave wireless communication networks performs significantly better than baseline methods in terms of average packet error rate (PER). It provides an effective solution to the resource allocation problem and exhibits great generalization performance, regardless of system stochasticity. The approach is promising for integrated sensing and communications networks and we plan to explore a learned protocol design in future work.

8 Acknowledgements

This work was supported by the European Union's Horizon EUROPE research and innovation program under grant agreement No. 101037090 - project CENTRIC.

References

[1] M. Giordani, M. Polese, A. Roy, D. Castor, and M. Zorzi, *A tutorial on beam management for 3GPP NR at mmWave frequencies*, *IEEE Communications Surveys & Tutorials*, vol. 21, no. 1, pp. 173–196, 2018.

[2] P. Susarla, Y. Deng, M. Juntti, and O. Sílven, *Hierarchial-DQN Position-Aided Beamforming for Uplink mmWave Cellular-Connected UAVs*, *GLOBECOM 2022-2022 IEEE Global Communications Conference*, pp. 1308–1313, 2022.

[3] V. Raj, N. Nayak, and S. Kalyani, *Deep reinforcement learning based blind mmwave MIMO beam alignment*, *IEEE Transactions on Wireless Communications*, vol. 21, no. 10, pp. 8772–8785, 2022.

[4] W. Ma, C. Qi, and G. Y. Li, *Machine learning for beam alignment in millimeter wave massive MIMO*, *IEEE Wireless Communications Letters*, vol. 9, no. 6, pp. 875–878, 2020.

[5] Y. Wang, A. Klautau, M. Ribero, A. C. K. Soong, and R. W. Heath, *MmWave vehicular beam selection with situational awareness using machine learning*, *IEEE Access*, vol. 7, pp. 87479–87493, 2019.

[6] L. Liang, H. Ye, G. Yu, and G. Ye Li, *Deep-learning-based wireless resource allocation with application to vehicular networks*, *Proceedings of the IEEE*, vol. 108, no. 2, pp. 341–356, 2019.

[7] K. Sridhara, A. Chandra, and P. S. M. Tripathi, *Spectrum challenges and solutions by cognitive radio: An overview*, *Wireless Personal Communications*, vol. 45, pp. 281–291, 2008.

[8] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, *Proximal policy optimization algorithms*, *arXiv preprint arXiv:1707.06347*, 2017.

[9] S. Rezaie, C. N. Manchón, and E. De Carvalho, *Location-and orientation-aided millimeter wave beam selection using deep learning*, *ICC 2020-2020 IEEE International Conference on Communications (ICC)*, pp. 1–6, 2020.

Algorithm 3: PPO Training Procedure

```
Input: The policy network $\pi_\theta(a, s)$, the value network $V_\phi(s)$ with weights $\theta$
       and $\phi$, $episodes$, learning rate $\alpha$, discount factor $\gamma$, weighting factor $\lambda$,
       number of time steps per episode $K$, environment $env$, clipping
       parameter $\epsilon$
Output: Policy function $\pi_\theta$, value function $V_\phi$
Initialize $\theta$ and $\phi$ weights
for $i \leftarrow 1$ to $episodes$ do
    Initialize environment: $s_1 \leftarrow env.reset()$
    Initialize empty lists for log probabilities, values, rewards, and old policy
      probabilities
    Initialize counter $c \leftarrow 0$
    while $c < K$ do
        for $k \leftarrow (1 + c)$ to $(N + c)$ do
            Compute action probabilities $\pi_\theta(a_k|s_k)$ and value estimate $V_\phi(s_k)$
              from neural network
            Sample action $a_k \sim \pi_\theta(a_k|s_k)$ and compute log probability
              $\log \pi_\theta(a_k|s_k)$
            Compute old policy probability $\pi_{\text{old}}(a_k|s_k)$
            Take action $a_k$ and observe reward $r_k$ and next state $s_{k+1}$
            Append log probability $\log \pi_\theta(a_k|s_k)$, value estimate $V_\phi(s_k)$, reward
              $r_k$, and old policy probability $\pi_{\text{old}}(a_k|s_k)$ to their respective lists
            Update state to $s_{k+1}$
        end
        for $k \leftarrow (1 + c)$ to $(N + c)$ do
            Compute advantages $A_k$
            Compute ratio $r_t(\theta) = \frac{\pi_\theta(a_k|s_k)}{\pi_{\text{old}}(a_k|s_k)}$
            Compute surrogate objective
              $\hat{L}_{\text{PPO}}(\theta) = \min\left(r_t(\theta)A_k, \text{clip}(r_k(\theta), 1 - \epsilon, 1 + \epsilon)A_k\right)$
            Compute value function loss $L^{(k)}_{\text{critic}} = \frac{1}{N}\sum_{k=1+c}^{N+c}(G_k - V_\phi(s_k))^2$
            Compute entropy regularization term $S[\pi_\theta(a_k|s_k)]$
            Compute total loss
              $L_{\text{total}}(\theta, \phi) = \frac{1}{N}\sum_{k=1+c}^{N+c} L^{(k)}_{\text{PPO}}(\theta) - c_1 L^{(k)}_{\text{critic}}(\phi) + c_2 S[\pi_\theta(a_k|s_k)]$
            Update neural network parameters by minimizing $L_{\text{total}}$ using the
              optimizer
        end
        $c \leftarrow (N + c)$
    end
end
return $\pi_\theta$, $V_\phi$
```

Table I.1: Simulation Parameters

Parameter	**Symbol**	**Value**
BS antennas	M_t	32
Frequency	f_c	28 GHz
Tx Power	P_t	5 W
Signal-to-Noise Ratio	SNR	20 dB
Scenario dimensions	SD	[100 x 100]
Number of UEs	U	3
Packet arrival probability	P^u	[0.6, 0.4, 0.3]
Packet dl/ul probability	$[p^u_{DL}, p^u_{UL}]$	[0.5, 0.5]
UEs initial positions [(x, y) permute]	$[[x^u_0, y^u_0]]$	[[0, 80], [0, 40], [0, 27]]
Size of downlink buffers	B^u_{DL}	5
Size of uplink buffers	B^u_{UL}	5
Number of time slots	K	1000
Number of training episodes	E_{train}	3000000
Number of testing episodes	E_{test}	6000

Table I.2: Training Algorithm Parameters

Parameter	**Value**
Number of layers for actor and critic	3
Neurons of actor layers	$[\{state\_size, 64\}, \{64, 64\}, \{64, action\_size\}]$
Neurons of critic layers	$[\{state\_size, 64\}, \{64, 64\}, \{64, 1\}]$
Activation function all layers	tanh
Actor probability mapping	softmax
Memory size N	80
Learning rate actor α_a	0.0003
Learning rate critic α_a	0.001
Discount factor γ	0.99
Beam tracking factor $\rho(k)$	[3, 0, -1]
Loss weighing c_1	0.5
Loss weighing c_2	0.01
Clipping parameter ϵ	0.2
Optimizer	Adam

ISSN (online): 2446-1628
ISBN (online): 978-87-7573-650-8

AALBORG UNIVERSITY PRESS