

6G BRAINS Deliverable D6.3

3D Measurement results and evaluation of Location Functions

Editor:	John Cosmas and Kareem Ali
Deliverable nature:	Report (R)
Dissemination level: (Confidentiality)	Public (PU)
Contractual delivery date:	30 June 2023
Actual delivery date:	14 July 2023
Suggested readers:	5G/6G System Developers, Industry, Manufacturers, Product Developers, Application Designers
Version:	1.0
Total number of pages:	59
Keywords:	6G, Time Difference of Arrival Localisation, Angle of Arrival Localisation,

Abstract

This deliverable presents the results of three localisation experiments performed in the 6G BRAINS project. The first hardware realisation experiment aimed to demonstrate RunEL's capability functionally measuring the location of 5G/6G Mobile User Equipment Terminals (UE) within centimetre accuracy using Time Difference of Arrival (TDoA). The second software simulation experiment aimed to demonstrate how Deep Learning Integrated Reinforcement Learning (DLIRL) algorithm can be used to perform intelligent beamsteering by estimating the Angle of Arrival between gNB smart base station and UE in Beyond fifth Generation (B5G) networks. The third experiment applied a realistic visible light positioning testing environment for analysing and demonstrating that an innovative and data/energy/cost efficient communication system can be used to measure the 3D position by calculating the height of the receiver as well as its x-y position.

Disclaimer

This document contains material, which is the copyright of certain 6G BRAINS consortium parties, and may not be reproduced or copied without permission.

In case of Public (PU):

All 6G BRAINS consortium parties have agreed to full publication of this document.

In case of Restricted to Programme (PP):

All 6G BRAINS consortium parties have agreed to make this document available on request to other framework programme participants.

In case of Restricted to Group (RE):

All 6G BRAINS consortium parties have agreed to full publication of this document. However this document is written for being used by <organisation / other project / company etc.> as <a contribution to standardisation / material for consideration in product development etc.>.

In case of Consortium confidential (CO):

The information contained in this document is the proprietary confidential information of the 6G BRAINS consortium and may not be disclosed except in accordance with the consortium agreement.

The commercial use of any information contained in this document may require a license from the proprietor of that information.

Neither the 6G BRAINS consortium as a whole, nor a certain part of the 6G BRAINS consortium warrant that the information contained in this document is capable of use, nor that use of the information is free from risk, accepting no liability for loss or damage suffered by any person using this information.

The EC flag in this document is owned by the European Commission and the 5G PPP logo is owned by the 5G PPP initiative. The use of the EC flag and the 5G PPP logo reflects that 6G BRAINS receives funding from the European Commission, integrated in its 5G PPP initiative. Apart from this, the European Commission and the 5G PPP initiative have no responsibility for the content of this document.

The research leading to these results has received funding from the European Union Horizon 2020 Programme under grant agreement number 101017226 – 6G BRAINS – H2020-ICT-2020-2. The content of this document reflects only the author's view and the Commission is not responsible for any use that may be made of the information it contains.

Impressum

[Full project title] Bringing Reinforcement-learning Into Radio Light Network for Massive Connections

[Short project title] 6G BRAINS

[Number and title of work-package] WP6, Hybrid Radio-Optical 3D Localisation System for Ultra-dense Machine Type Communications

[Number and title of task]

T6.1: High Resolution Radio-Light 3D Simultaneous Localization and Mapping (SLAM) and Fusion for Dynamic Ultra-dense Network

Document title: 3D Measurement results and evaluation of Location Functions

Editor: John Cosmas and Kareem Ali, Brunel University

Work-package leader: John Cosmas, Brunel University

Estimation of PM spent on the Deliverable: 25

Copyright notice © 2023 Participants in 6G BRAINS project

Executive summary

This deliverable presents the results of three localisation experiments performed in the 6G BRAINS project.

The first hardware realisation experiment aimed to demonstrate RunEL's capability functionally measuring the location of 5G/6G Mobile User equipment Terminals (UE) within centimeter accuracy. This capacity is accomplished by measuring the Time Difference of Arrival (TDoA) of the terminal transmissions received and processed by several antennae elements connected to access points of the base-station. This provided a good indication of the degree of accuracy that can be obtained using this technology.

The second software simulation experiment aimed to demonstrate how Deep Learning Integrated Reinforcement Learning (DLIRL) algorithm can be used to perform intelligent beamsteering by estimating the Angle of Arrival between gNB smart base station and UE in Beyond Fifth Generation (B5G) networks. The smart base station in B5G networks steers the beam towards appropriate user equipment based by Neural Network analysis of isotropic transmissions of UE. The beam direction is then optimised through reinforcement learning that delivers significant improvement in signal to noise ratio (SNR). This experiment illustrates how localisation of user equipment can potentially be obtained from one access point by combining angle of arrival measurements with Time Difference of Arrival measurements.

The third experiment applied a realistic visible light positioning testing environment for analysing and demonstrating that an innovative and data/energy/cost efficient communication system can be used to measure the 3D position by calculating the height of the receiver as well as its x-y position.

List of authors

Company	Author	Contribution
Brunel University	Geoffrey Eappen, John Cosmas, Raj Nilavalan	Section 3 AoA from OFDM Transmissions Measurement Campaign
RunEL	Baruch Globen, Israel Koffman	Section 2 - Accurate UE location measurement
ISEP	Xun Zhang	Section 4 - OWC measurement campaigns

Table of Contents

Contents

1	Introduction.....	11
2	Sub 6GHz Functional Test Measurement Campaign.....	12
2.1	Aim of Experiment	12
2.2	Background Theory.....	12
2.3	Experimental Setup	13
2.4	Experimental Procedure	14
2.5	Experimental Results and Analysis	18
3	AoA from OFDM Transmissions Simulation Measurement Campaign	19
3.1	Aim of Experiment	19
3.2	Background Theory.....	19
3.2.1	Literary Survey.....	19
3.2.2	Research Gap.....	21
3.3	Experimental Setup	21
3.3.1	System Model.....	21
3.3.2	Channel modelling.....	23
3.3.3	The Problem Formulation	23
3.3.4	DLIRL-Based Beamforming	24
3.4	Experimental Procedure	29
3.5	Experimental Results and Analysis	30
4	OWC measurement campaigns.....	38
4.1	Aim of Experiment.....	38
4.2	Background Theory:.....	38
4.2.1	impact on experimenters	38
4.2.2	Introduction of 3D VLP algorithm	39
4.3	Simulation set up and results	45
4.3.1	Set up of the simulation	45
4.3.2	Simulation results of height estimation	46
4.3.3	3D positioning results.....	50
5	Conclusions.....	56

List of figures and tables

List of figures:

Figure 2-1: Hyperbola pattern crossing at the target UE position.	13
Figure 2-2: ToA Accurate Measurements Experiment diagram	14
Figure 2-3: Initial Setup Configuration	14
Figure 2-4: Tests Setup Configuration	15
Figure 2-5: Target and Reference UE Installed Antennae	16
Figure 2-6: Antenna Installations	16
Figure 2-7: RU 4 Installed Antennae diagram and photos	16
Figure 2-8: Experiment UE emulator, RU, DU and Location Server	17
Figure 2-9: Experiments Results Sample	18
Figure 3-1: B5G system model	22
Figure 3-2: Beam coherence time interval during training process	25
Figure 3-3: The DLIRL beamforming system model.....	27
Figure 3-4: Comparative analysis of SE with reference to IAB SNR	31
Figure 3-5: Comparative analysis of SE with reference to increase in number of antennas	31
Figure 3-6: Comparative analysis using BER	32
Figure 3-7: Beamform towards UE using DNN and RL separately	33
Figure 3-8: Beamform towards UE using DLIRL	34
Figure 3-9: Training validation accuracy of DLIRL, DNN and RL.....	35
Figure 3-10: (a) Demonstration of DLIRL based adaptive beamforming using MATLAB raytracing and siteviewer. (b) DLIRL based beamsteering with user movement.....	36
Figure 3-11: SE versus training samples for DLIRL, RL and DNN	37
Figure 4-1: The framework of proposed 3D environment-aware Visible light-based positioning algorithms.	39
Figure 4-2: The framework of Step 2: Receiver's height estimation.....	42
Figure 4-3: Simulation set up: placement of transmitters (red dots) and reference points of the receiver (blue dots) in (a) 3D view (b) 2D view	46
Figure 4-4: Simulation flow for four groups with different repeat time R.....	48
Figure 4-5: Percentage of successful height estimations of different groups with different repeat time R.....	49
Figure 4-6: Distribution of 3D positioning results when height of receiver was from 0 m to 2 m.....	52
Figure 4-7: CDF of the positioning results in different height of receiver by using the proposal.....	53

List of tables:

Table 3-1: Simulation Parameters.....	30
Table 3-2: SE performance evaluation	32
Table 3-3: BER performance evaluation.....	33
Table 3-4: Performance evaluation of the training algorithm	35
Table 4-1: Key configuration parameters of the simulation.....	47
Table 4-2: The height estimation results under varying receiver heights	47
Table 4-3: Simulation results of 3D position.....	51
Table 4-4: Average time taken to perform a positioning for one reference point in different height of receiver.	54

Abbreviations

A list of abbreviations is strongly recommended

5G	Fifth Generation (mobile/cellular networks)
5G PPP	5G Infrastructure Public Private Partnership
6G BRAINS	6G Bringing Reinforcement-learning Resource Allocation of Radio Light Network for Massive Connections
ACO	Ant Colony Optimization
AI	Artificial Intelligence
AoA	Angle of Arrival
APD	Avalanche Photodiode
B5G	Beyond 5G
BLE	Bluetooth Low Energy
CAPEX	Capital Expenditure
CDF	Complementary Distribution Function
CIR	Channel impulse response
CSI	Channel State Information
DL	Downlink
DLIRL	Deep Learning Integrated Reinforcement Learning
DNN	Deep Neural Network
TDoA	Time Difference of Arrival
E_b/N_0	energy per bit to noise power spectral density ratio
ET	Environmental Thumbprint
FC	Fusion Centre
FE	Front End
FFT	Fast Fourier Transform
IAB	integrated access and backhaul

IET	Institute of Engineering and Technology
IoT	Internet of Things
IRL	Integrated RL
Kph	kilometres/hour
KPI	Key Performance Indicator
kNN	k-Nearest Neighbors
LABNT	learning-based adjustable beam number training
LED	Light Emitting Diode
LOS	Line of Sight
LSTM	long short-term memory
LTE	Long Term Evolution
M2M	Machine to Machine
MIMO	multiple input multiple output
MMSE	Minimum Mean Squared Error
MSE	mean square error
NB-IoT	Narrowband Internet of Things
NLoS	Non-Line of Sight
OFDM	Orthogonal Frequency Division Multiplexing
OPEX	Operational Expenditure
OSM	Open Street Map
OWC	Optical Wireless Communications
PE	Position Error
PRS	Positioning Reference Signal
PSO	Particle Swarm Optimization
QoE	Quality of Experience
QoS	Quality of Service

R&D	Research and Development
ReLU	Rectifier Linear Unit
RF	Radio Frequency
RL	Reinforcement Learning
RS	Reference Signal
RSSI	Received Signal Strength Indication
RU	Remote Unit
SBR	Shooting and Bouncing Ray
SDN	Software Defined Networks
SE	Spectral Efficiency
SINR	Signal to Interference Noise Ratio
SNR	Signal to Noise ratio
SWOT	Strengths, Weaknesses, Opportunities, and Threats analysis
TDoA	Time Difference of Arrival
ToA	Time of Arrival
UE	User Equipment
UL	Uplink
VLP	Visible Light Positioning

1 Introduction

The presents three bodies of work:

- Sub 6GHz Functional Test Measurement Campaign – The experimental equipment that was built by RunEL to measure Time Difference of arrival with a resolution 0.33 ps was functionally tested to see if it can consistently and reliably produce measurement results. This is a necessary preliminary test before technical tests are performed to determine the accuracy of these measurements, which will be reported in Deliverable 6.4 - Integrated Simultaneous Localisation and Mapping.
- AoA from OFDM Transmissions Simulation Measurement Campaign – The simulation experiment that was built by UBRU to measure Angle of Arrival of user equipment from the radio signature of the received OFDM signal obtained from isotropic transmissions from User Equipment at the gNB Base Stations. DNN was used to interpret radio signature into angle of arrival and DRL was used to improve the accuracy of the Angle of Arrival of the beamsteered antenna based on optimising the signal strength.
- A 3D environment-aware Visible Light Positioning (VLP) system was introduced with a novel algorithm which estimates the height corresponded to the true height of the receiver

2 Sub 6GHz Functional Test Measurement Campaign

2.1 Aim of Experiment

The aim is to demonstrate RunEL's capability functionally measuring the location of 5G/6G Mobile User equipment Terminals (UE) within centimeter accuracy. This capacity is accomplished by measuring the Time Difference of Arrival (TDoA) of the terminal transmissions received and processed by several antennae elements connected to access points of the base-station.

2.2 Background Theory

Methods for positioning end terminals in a wireless network environment, are typically accomplished by measuring the intensity of the received RF signal using the RSSI (Received Signal Strength Indication or RSSI) data. The accuracy of these solutions depends on the number of nearby Access Points and can reach an accuracy of more than several meters.

Enhancing measurements accuracy can be performed by Angle of Arrival (AoA) estimation using directional beams and MIMO (Multiple Input Multiple Output) multi-elements antenna combined with triangulation.

Other methods use Bluetooth Low Energy (BLE) beacons mainly for indoor positioning at shopping centres, hospitals and similar campuses.

DToA is a localization technique where known location antennae simultaneously receive the same RF signal transmitted by the target end terminal, and by measuring the Differential Time of Arrival between pairs of the base-station receiving antennae set.

The TDoA measurements are reported to the wireless network Core Positioning Function unit which carries out a triangulation-based calculation for locating the target end terminal position. The triangulation is based on the differential time of arrival measurements, known location of the receiving antenna elements and the speed of light. The triangulation can be done by conventional methods solving a set of hyperbolic equations (hyperbola is the geometric line over which the differential between the distances from each point on the line to a measuring antenna pair is constant).

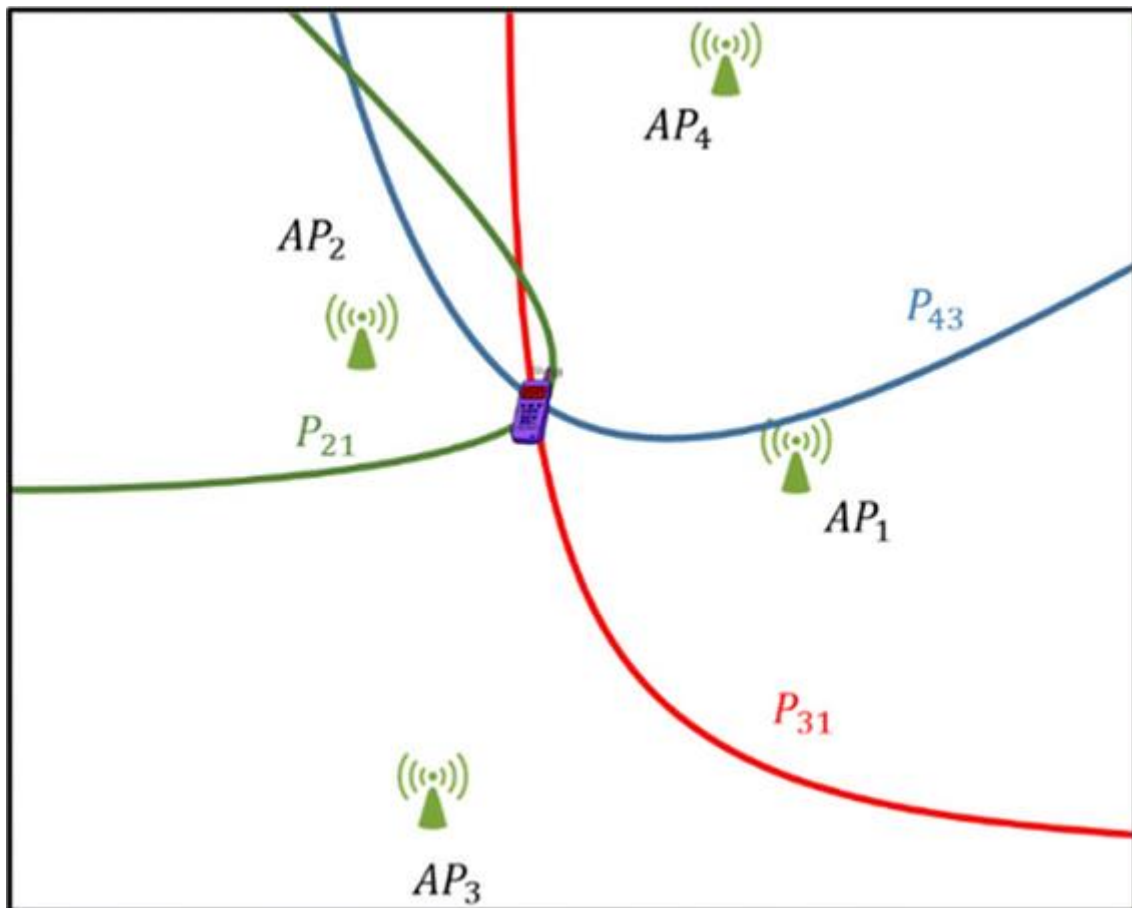


Figure 2-1: Hyperbola pattern crossing at the target UE position.

The above drawing depicted are three hyperbola patterns. For example, the blue pattern is the outcome of the TDoA measured by the 3rd and 4th antennae elements access points pair.

The differential of the distances from each point on this pattern to the antenna elements position divided by the speed of light yields the measured TDoA for this pair.

The problem left after understanding the positioning method is how to measure accurately the TDoA.

The following sections describe the accurate measurements carried out at RunEL lab. This measurements technique is based on RunEL's patent pending method.

2.3 Experimental Setup

The test was carried out at RunEL lab.

The following are the basic principles behind the tests.

- Target UE localization is based on a triangulation method by measuring its distance from at least 3 Base Stations or 3 or more antennae element connected to the same RU.
- Measured is the TDoA of the same signal replica simultaneously received at each antenna element. The speed of light, the known location of the receiving antenna elements and the measured TDoA are the parameters used by the core localization server to calculate the target's position.

- A known location UE is used for calibration and for enhancing the measurements quality.
- The Accuracy of the time measurements is based on a RunEL patent pending algorithm that allows measuring the signal reception time in the range of ten picosecond accuracy (33 Picoseconds = 1 cm).
- In the following drawing depicted is the experiment measurements scenario, including 4 access points (an antenna element connected to an RF Front End (FE) and a processor) receiving signals transmitted by the target UE. The hyperbola patterns, related to the pairs TDoA, are generated by the Core Localization Function.

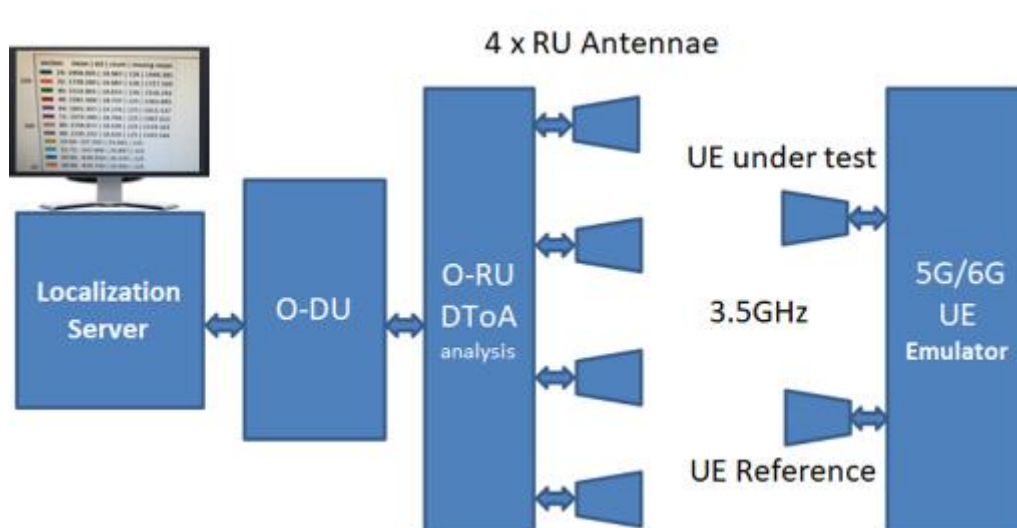


Figure 2-2: ToA Accurate Measurements Experiment diagram

2.4 Experimental Procedure

The setup is based on RunEL RU and DU with enhancements enabling to accomplish the DToA measurements.

A simple setup is used initially. This simple configuration is using an RU emulating a transmitting UE.

This configuration is depicted in the following diagram.

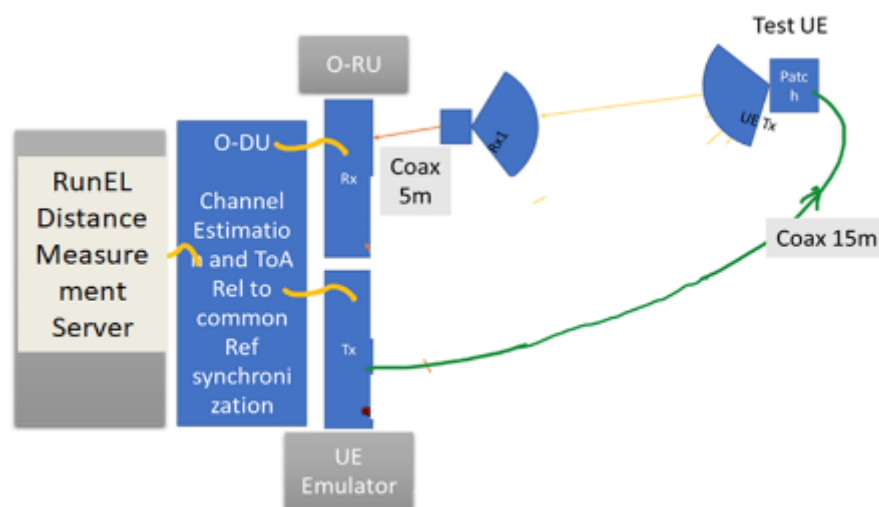


Figure 2-3: Initial Setup Configuration

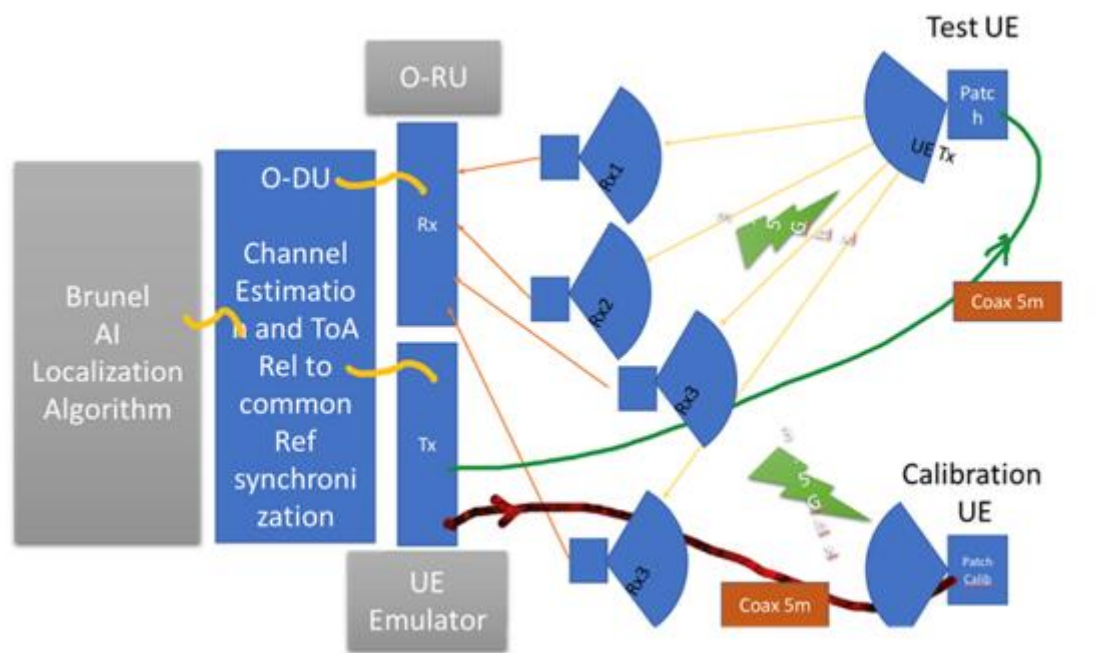


Figure 2-4: Tests Setup Configuration

The above drawing shows the tests setup configuration. The UE emulator (implemented by an RU) transmits a 5G SRS signal, or other RS (Reference Signal), at 3.5GHz which is simultaneously received by the four RU antennae. Another part of the RU UE emulator is used to generate the reference UE signals.

At the RU, after detection, sampling, and analog to digital conversion. The digitized signal is FFT converted. Following are the time measurements. Eight time measurements are carried out (each RU antenna element measures time delay related to the target and the reference UE). The process includes calibration with the reference UE to accommodate bias, connecting cables impact and deviation errors. The TDoA differential processing inherently cancels part of measurements errors.

On the drawing right side is the UE emulator unit, which emulates the target UE and the reference UE. These emulator UL RF signals transmissions are received by the RU antenna elements.

The RU in the above drawing is connected to 4 antenna elements. These antennae are spread over the lab hauls. The antennae distance spread is about 10 meters, enabling to acquire adequate TDoA range.

The Location Measurement Server at the drawing left side calculates the 3D location of the target UE. The antenna pairs TDoA measurements every 500 microseconds enable to accurately, accuracy of 1 cm, finding the target UE position.



Figure 2-5: Target and Reference UE Installed Antennae

Above drawing is a picture of the UE emulator antennae.

The next drawings exhibit the 4 RU antennae set installation.



Figure 2-6: Antenna Installations

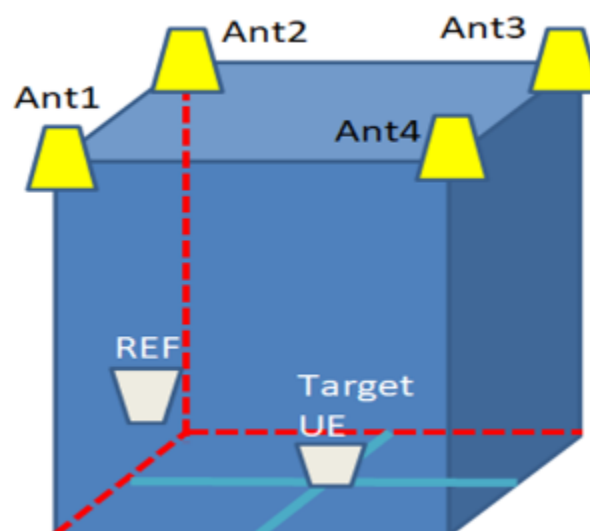


Figure 2-7: RU 4 Installed Antennae diagram and photos

The UE emulator, the RU, the DU and the location measurements server are shown in the next drawing.

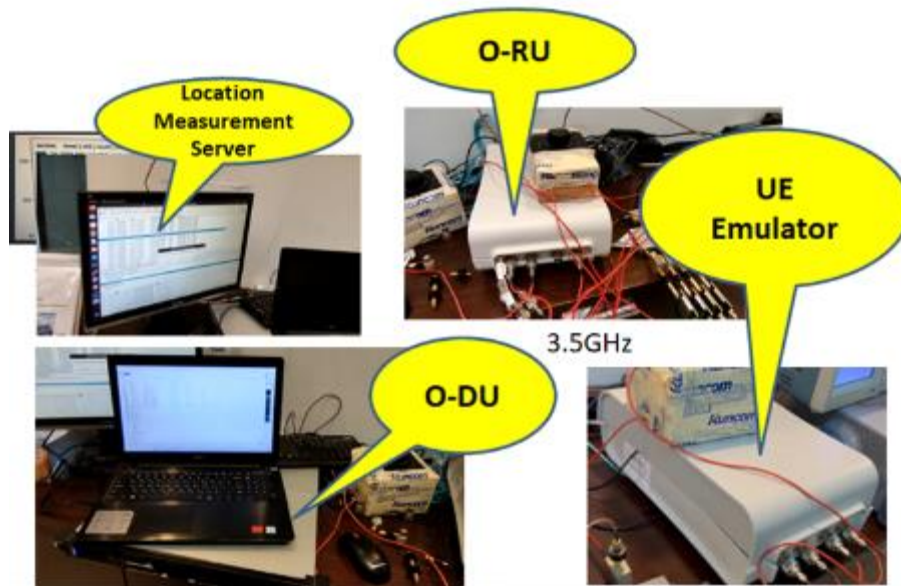


Figure 2-8: Experiment UE emulator, RU, DU and Location Server

The transmitted signals are received, digitized, and analyzed. These includes 8 measurements (4 for the reference UE and 4 for the target UE).

TDoA results for antennae pairs are delivered to the core server to analyze the target location. Result samples are presented in the next section.

2.5 Experimental Results and Analysis

In the following drawings presented is a sample of the experimental results.

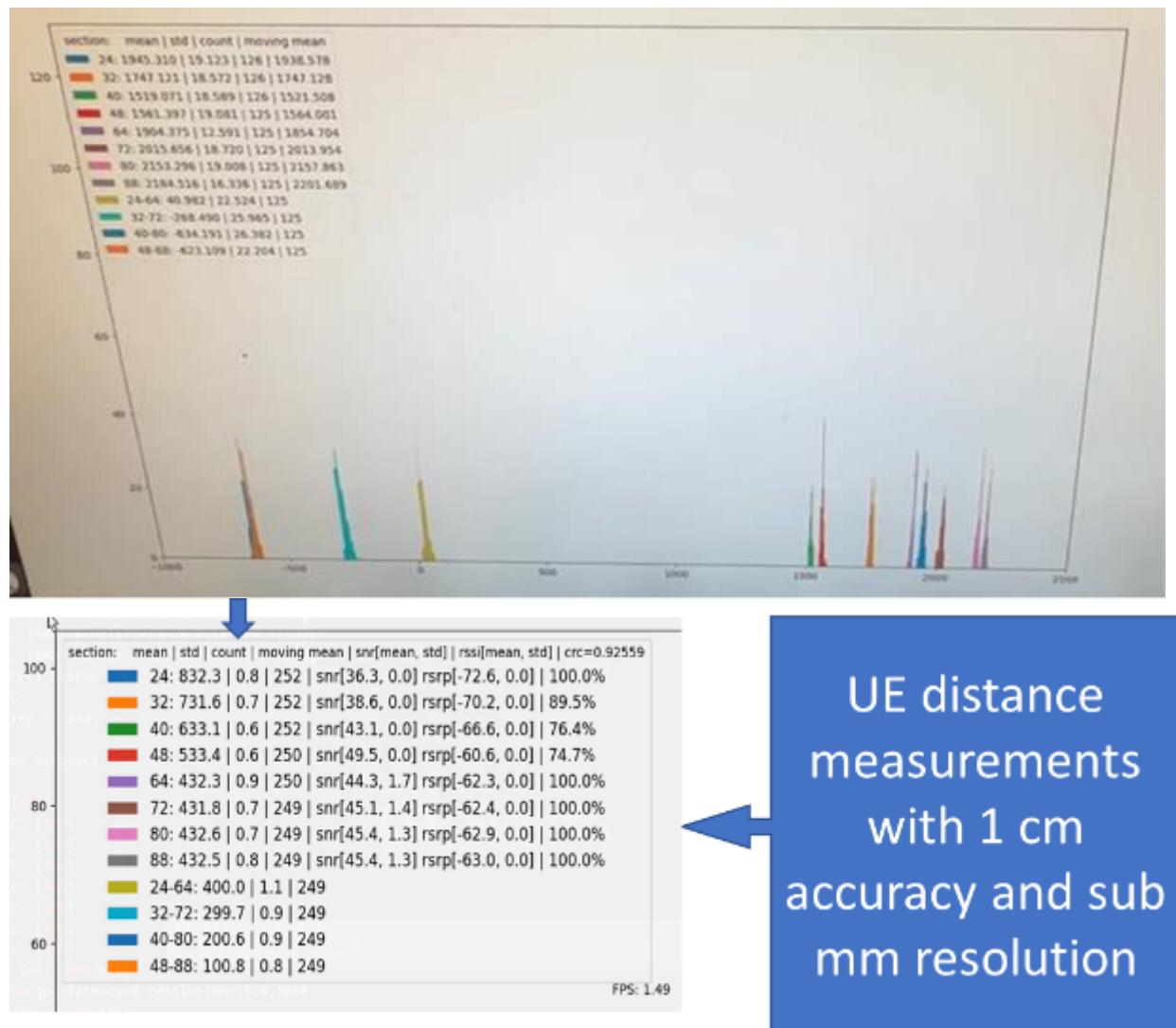


Figure 2-9: Experiments Results Sample

The upper drawing includes a graphical and a numerical view of the measurements. The horizontal axis is the distance in centimeters.

The lower drawing includes a closer look to the numerical results.

We get 8 Time of Arrival (ToA) measurement results (each RU antenna element measures the ToA related to the target and the reference UE).

The lower part includes the TDoA with the reference.

The results include averaging in a short and a long averaging window.

These tests demonstrated that the UE position can be measured using 5G/6G network signals in the range of centimeter accuracy.

3 AoA from OFDM Transmissions Simulation Measurement Campaign

3.1 Aim of Experiment

A Deep Learning Integrated Reinforcement Learning (DLIRL) algorithm is proposed for comprehending intelligent beamsteering in Beyond Fifth Generation (B5G) networks. The smart base station in B5G networks steers the beam towards appropriate user equipment based on the acquaintance of isotropic transmissions. The foremost methodology is to optimize beam direction through reinforcement learning that delivers significant improvement in signal to noise ratio (SNR). This includes alternate path finding during path obstruction and steering the beam appropriately between the smart base station and user equipment. The DLIRL is realized through supervised learning with deep neural networks and deep Q-learning schemes. The proposed algorithm comprises of an online learning phase for training the weights and a working phase for carrying out the prediction. Results confirm that the performance of the B5G system is improved considerably as compared to its counterparts with a spectral efficiency of 11 bps/Hz at SNR=10 dB for a bit error rate performance of 10^{-5} . As compared to reinforced learning and deep neural network with a deviation of $\pm 3^\circ$ and $\pm 5^\circ$, respectively, the DLIRL beamforming displays a deviation of $\pm 2^\circ$. Moreover, the DLIRL can track the user equipment and steer the beam in its direction with an accuracy of 92%. This research, which was performed on behalf of 6G BRAINS project, has also been originally published in an IET open-source publication [36].

3.2 Background Theory

3.2.1 Literary Survey

The increased usage of wireless communication-based applications and the huge demand for data rates have resulted in the development of B5G networks. These future networks are required to meet the demands of high data rates and enhanced user experience. In this paper, Artificial Intelligence (AI) is incorporated with beamforming and millimetre Wave (mmWave) enabling intelligent beamsteering based on Channel State Information (CSI), thus enabling a high data rate and better user experience. The novel deep learning integrated reinforcement learning (DLIRL) algorithm is proposed for the beamforming solution to overcome the problems associated with mmWave like blockage impacting the coverage, reliability of highly mobile links, latency overheads associated with high-speed mobile devices in dense mmWave scenarios that require frequent hand-offs [1, 2]. From the research gap, it can be visualized that most of the works for beamforming for B5G networks were carried out employing the conventional schemes, but the employed schemes are not effective enough to steer the beam based on the user locations [12]. Also, the works in [20, 21] for beam direction estimation based on vehicle motion were carried out on a single domain considering only the vertical direction. But implementing the B5G networks for the real time scenario, it is important to consider beam direction from 3D point of view [11]. To have an enhanced beamforming for the real time scenario based B5G networks, we are proposing in this paper an AI-based novel DLIRL efficient beamforming scheme. The proposed scheme is trained via pilot symbols received at the coordinated integrated access and backhaul (IAB) nodes with negligible training overheads. These symbols represent the foot-prints of pilot information communication with the nearby scenarios. Utilizing this information for training the DLIRL to

produce the efficient beamforming is the idea behind this work. In this implemented coordinated beamforming system, the UE performs a transmission of an upstream pilot arrangement, which is cooperatively acknowledged by the organizing IAB nodes. The acknowledged information carries valuable statistics regarding the nearby scenarios because of their communication to the nearby scenarios. Through deep learning, the prototype studies enhance the beamforming vector at IABs by means of the statistics so as to improve the signal to interference noise ratio at user equipment. The DLIRL constructed beamforming acquires the use of principal patterns to envisage the ideal beamforming vectors in IABs. Here, the beamforming vector through DNN exploitation of isotropic communication from UE devices is arranged in a lattice over the exposed region. The estimated impulse response at each of the IAB receiver functions acts as input data to a DNN model and acquires state space for reinforcement learning (RL). The RL steers accurate and adaptive beams through its experienced learning. This projected method affords an inclusive solution for high mobile B5G scenarios with improved network connectivity, reduced latency, and minimized training overheads. The long short-term memory (LSTM) based hotspot prediction in small cell 5G networks has been discussed in [25]. Here, adaptive beamforming is carried out to adjust the beam towards the small cell for traffic aggregation between the small cell and macro-cell. The requirement to adjust narrow beams in mmWave communication by leveraging machine learning and using concurrent and multiarmed bandit techniques to establish robust links has been explored in [26]. The requirement of deep learning techniques for 5G systems at the physical layer, network layer and applications layer to process information and automate decisions has been detailed in [27]. The requirement of deep learning in channel estimation for cooperative beam-forming in massive multiple input multiple output (MIMO) 5G systems has been discussed in [28]. An adaptive hybrid beamforming for 5G MIMO mmWave networks has been detailed in [29]. Here, to reduce the signalling overhead and blocking probability, analogue-digital beamforming approach is used to generate beams based on the traffic demands. The exploration of narrow beamwidths and adaptive steering of signals to reduce inter reference and energy consumption has been discussed in [30]. The importance of beamforming for enhanced signal quality, improved network capacity, frequency reuse, mitigation of multipath effects, estimation of angle of arrival and tracking of mobile devices have been explored [31]. The requirement of machine learning algorithms to track communication scenarios and handle big data in 5G MIMO systems has been discussed in [32]. Here, machine learning based classification models have been developed for beam selection. The deep learning framework to allocate resources for TV multimedia service in 5G scenario has been detailed in [33]. Here, an LSTM-based deep learning model has been developed to model the traffic pattern for resource allocation under Quality of Service (QoS) requirements. A learning-based adjustable beam number training (LABNT) algorithm has been developed in [34] for optimal beam direction and reduced training overhead. The trade-off between beam alignment accuracy and spectral efficiency in beamforming training for non-line-of-sight mmWave systems has been demonstrated. The concept of deep transfer learning to explore the beamforming vector in massive MIMO systems has been explored in [35]. Here, the trade-off between the number of training data and uncertainty of real-time channel has been discussed. Further, the requirement of an effective deep learning model to train 5G system with less overhead and latency has been highlighted.

3.2.2 Research Gap

From the literature, it can be inferred that effective schemes have been developed for beamforming [1–3]. The proposed algorithm in [1] is an initial work on adaptive beamforming. In the proposed scheme, several sensors are considered to obtain correlation. However, it is not practical to mount too many sensors on the user equipment. In [2], user location-based transmit and receive beamforming is proposed considering only the Line of Sight (LoS) condition, which is not always feasible for a real-time operation in real environments where the LoS path may be blocked due to human activity. Moreover, the proposed work is considered only for the known user locations. In addition, ref. [4] proposed robust beamforming with an assumption that Angle of Arrival (AoA) and CSI is known. Nevertheless, considering known AoA and CSI is complicated and often unrealizable [3]. In [5], hybrid beamforming with gradient iterative algorithm is proposed considering instantaneous CSI. However, the proposed gradient algorithm is prone to get stuck at local optima solutions instead of global optima [10]. Beamforming considering unknown user location and instantaneous CSI is proposed in [6]. The major challenge in [6] is the huge training overhead associated with large array beamforming vectors. To tackle the problem of training overheads, authors in [7] proposed deep learning-based beamforming. Deep learning-based predictive beamforming is also proposed in [8] considering location awareness. Employing recurrent Long Short-Term Memory (LSTM) has inefficient exploration hampering its prediction abilities [9]. The work in [10] proposed a promising RL with Q learning for a joint optimized beamforming scheme. However, the work considered known optimized beamform vectors and user locations, which are unattainable in a real scenario. The deep learning-based beamforming scheme is an effective way of achieving efficient beamforming [11]. The DL scheme is also a fast way of beamforming for the mmWave channels [12]. However, the existing DL network requires a huge amount of training data to achieve a good performance via beamforming. Moreover, this DL network is based on mere learning and thus it is not easy to comprehend the output [13]. Therefore, in this work, we propose a DL plus RL network for beamforming capable of getting trained with fewer data. The proposed scheme is not merely based on the training of the DL network but also on the experience provided by the RL network in improving the beamforming performance. Recently, RL for multimedia data segregation has been investigated for health-care sector using fuzzy algorithms to improve the quality of service in fog computing [22]. Alternatively, multimedia data segregation using k-fold random forest has been developed for reducing the latency from Internet of Things (IoT) devices in healthcare environments [23]. Further, Ant Colony Optimization (ACO) algorithm has been utilized to offload data in resource constrained IoT scenarios [24].

3.3 Experimental Setup

3.3.1 System Model

The considered system model comprises of I number of Integrated Access and Backhaul nodes termed as IABs serving a UE fitted with the single isotropic antenna as shown in Figure 3-1. The IAB node is connected to the 5G core via an IAB donor. The 5G core carries the brain for the intelligent and adaptive beamforming in the form of a DLIRL network. Each IABs are equipped with N^T antennas communicating information symbol $s^k \in \mathbb{C}$ for the K^{th} subcarrier here $K = 1, \dots, K$. Each IAB has baseband precoder vector for each K^{th} subcarrier $\mathbf{f}_{BB}^k \in \mathbb{C}^{I \times 1}$, and for RF precoder $\mathbf{f}_{RF} \in \mathbb{C}^{N^T \times 1}$. The RF precoder for the l^{th} IAB at m^{th} antenna element can be

modelled as a phase shifter network which is mathematically represented as $\mathbf{f}_{\text{RF}}^{l,m} = 1/\sqrt{N^T} e^{j\phi^{l,m}}$, here $\phi^{l,m}$ is quantized angle [15]. The downlink transmission for the transmitted data symbol can be represented as $\mathbf{y} = \mathbf{f}_{\text{RF}}^l \mathbf{f}_{\text{BB}}^K s^K$. Here $E[s^K(s^K)^H] = P^K/k$, P^K is the power associated with the K^{th} subcarrier and k is the total number of subcarriers. The constraint in the total transmission power of IAB should be

$$\|\mathbf{F}_{\text{RF}} \mathbf{f}_{\text{BB}}^K\|_F^2 = 1, K = 1, 2, \dots, k.$$

Here,

$$\mathbf{F}_{\text{RF}} = \text{blkdiag}(\mathbf{f}_{\text{RF}}^1 \dots \dots, \mathbf{f}_{\text{RF}}^I) \in \mathbb{C}^{IN^T \times I}$$

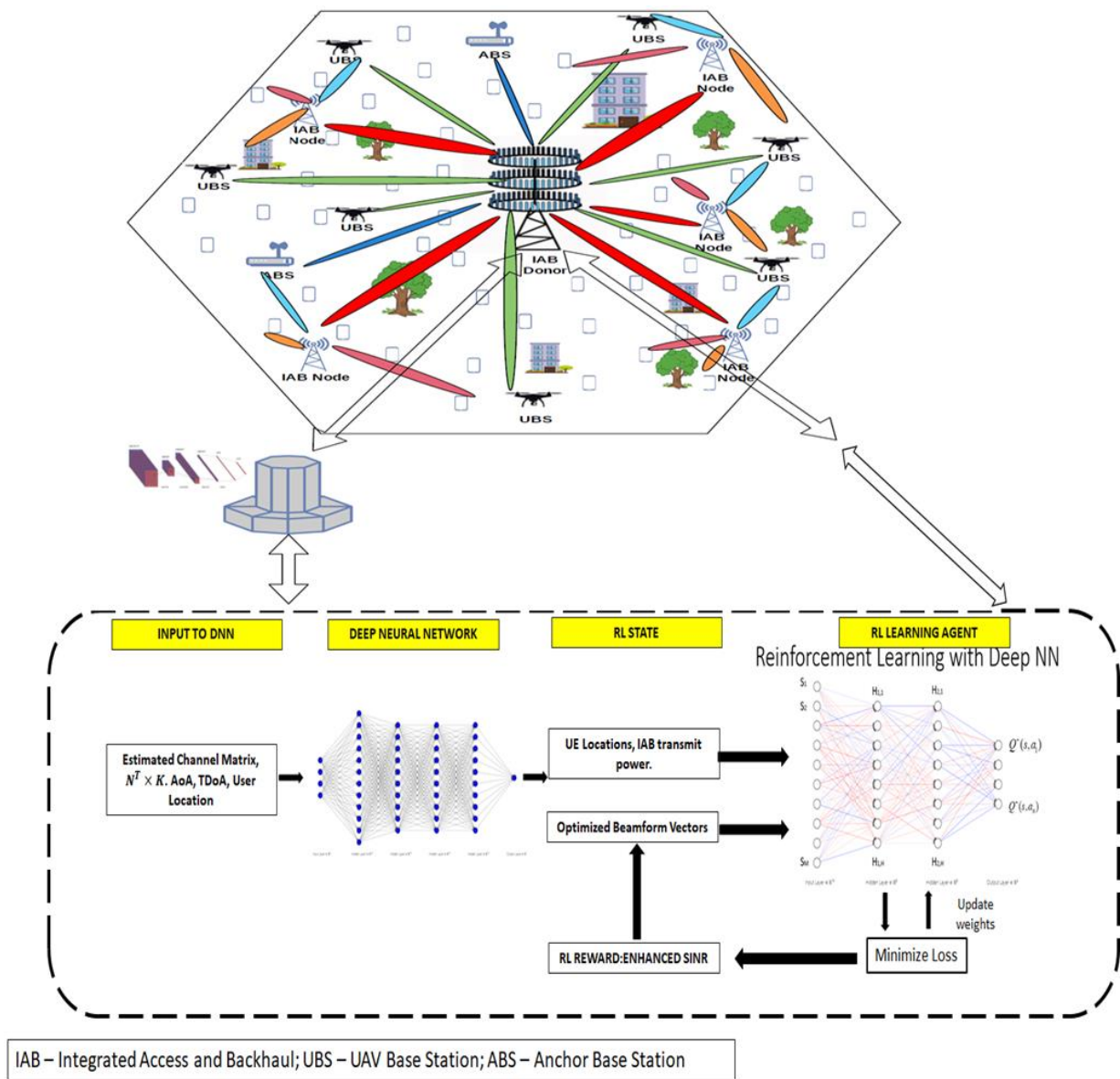


Figure 3-1: B5G system model

The channel vector between the l th IAB and the UE for the K th subcarrier is denoted as $\mathbf{h}_{l,K} \in \mathbb{C}^{N^T \times 1}$.

Then the received data stream $\mathbf{x}_{l,K}$ by the UE from the l th IAB during the downlink at K th subcarrier can be written as in Equation (1) [14]:

$$\mathbf{x}_{l,K} = \sqrt{P_{avg}} \mathbf{h}_{l,K}^H \mathbf{f}_{\text{RF}}^{l,K} f_{BB}^{l,K} s^K + n_{K,l}, \quad (1)$$

where P_{avg} is the average received power at the UE. The $n_{K,l} \sim \mathcal{N}_c(0, \sigma^2)$ is the K th subcarrier noise at the UE. The symbol notation involved in the system modelling is represented in the Nomenclature list.

3.3.2 Channel modelling

The channel between IABs and the UE is considered as a wideband mmWave channel with $c = 1, \dots, C$ clusters contributing to one ray of time delay as τ_c , AoA as ϕ_c and ϑ_c for azimuth and elevation angles, respectively. The channel path loss between UE and the l th IAB over K th is represented as $L_{l,K}$. The antenna array response vector between l th IAB and the UE for the K th subcarrier is represented as $\mathbf{a}_{l,K}(\theta_c, \phi_c)$. Mathematically the channel can be modelled for the K th subcarrier as shown in Equation (2) [14]:

$$\mathbf{h}_{l,K} = \sqrt{\frac{N^T}{L_{l,K}}} \sum_{c=1}^C \beta_c \mathbf{a}_{l,K}(\theta_c, \phi_c), \quad (2)$$

Here β_c is the complex gain associated with the resolvable path c . The considered channel model is assumed to remain constant over coherence time T_c as it is a block fading channel.

3.3.3 The Problem Formulation

The challenges associated with 5G standards are to reach the goals of higher data rate, lower latency, better coverage, and high mobility. To achieve this, it is important to have the most flexible and controlled beamforming scheme. The existing beamforming techniques [1–6] tried to achieve flexibility and control via dedicated transmit/receive for each element. Considering massive MIMO-based wireless communication systems, building this type of architecture is highly difficult due to extensive cost, power, and space-based limitations [12], thus, hindering the design budget. With this motivation, this work is dedicated to utilising an effective AI training scheme based on DL and RL combinedly termed DLIRL for the beamforming strategy.

To quantify the goals achieved by the proposed scheme, the performance is evaluated by considering the beamforming system that can maximize effective spectral efficiency for mmWave-based wireless communication applications at UE. For that purpose, the achievable rate at UE for the considered hybrid beamformer, system, and channel model are evaluated as per Equation (3):

$$S_e = \frac{1}{k} \sum_{K=1}^k \log_2 \left(1 + \text{SNR} \left| \sum_{l=1}^I \mathbf{h}_{l,K}^H \mathbf{f}_{\text{RF}}^l \mathbf{f}_{\text{BB}}^{l,K} \right|^2 \right), \quad (3)$$

Here, I is the total IABs considered. SNR represents the signal to noise ratio at UE denoted as $(P_{\text{avg}}/k\sigma^2)$. The objective of this work is to create an effective beamforming design and channel training so as to maximize the achievable rate at UE. The final optimization problem can be deduced as:

$$S_e(\mathbf{f}_{\text{RF}}^l, \mathbf{f}_{\text{BB}}^{l,K}) = \arg \max \frac{1}{k} \sum_{K=1}^k \times \log_2 \left(1 + \text{SNR} \left| \sum_{l=1}^I \mathbf{h}_{l,K}^H \mathbf{f}_{\text{RF}}^l \mathbf{f}_{\text{BB}}^{l,K} \right|^2 \right), \text{ s.t. } \left\| \mathbf{f}_{\text{RF}}^l \mathbf{f}_{\text{BB}}^{l,K} \right\|^2 = 1, \quad \forall l, \quad (4)$$

3.3.4 DLIRL-Based Beamforming

The deep learning integrated with reinforcement learning scheme for the beamforming design employs techniques that make use of the learning strategy by performing mapping between beamforming weights and the environmental setup, channel estimates, AoA, and Time Difference of Arrival (TDoA). The proposed DLIRL employs a pilot signal transmitted from UE to the IAB's to learn about the channel condition and predict the optimal beamform weights. The pilot signals received at IABs are the result of the interaction of the signal with the environment during its propagation. These reflected and diffracted waves are jointly received at different IABs and carries the thumbprint of the environmental factors and channel conditions. The Environmental Thumbprint (ET) carried by these pilot signals is employed for training. The DLIRL has two periods: working and training. The DLIRL initiates with the training period. During this period, DLIRL receives pilot sequences transmitted from the predefined UEs positions. The UE transmission is omnidirectional and it carries ET. The DLIRL then maps the received sequence with the training process and learns it.

The working period marks the prediction scheme of the DLIRL and performs prediction of the optimal beamform without any need for additional training. Multiple advantages achieved via the proposed scheme are that it does not require additional resources for learning during the working period. Moreover, the proposed scheme employs an omnidirectional UE pilot sequence for the training period. Therefore, it has minimal overheads during the training period. Also, the proposed scheme can be trained for any environment including LoS and Non-Line of Sight (NLoS).

3.3.4.1 Training Period

In this period DNN observes the environment and trains the deep neural network. We consider a beam coherence time T_c , a concept introduced in [15] for mmWave-based wireless communication systems also shown in Figure 3-2. It can be defined as the period over which the beams are unchanging. Considering T_{tr} as the channel training period of the first T_{tr} time instants of the T_c , then Equation (4) can be re-written as in Equation (5).

$$S_e(\mathbf{f}_{\text{RF}}^l, \mathbf{f}_{\text{BB}}^{l,K}) = \left(1 - \frac{T^{tr}}{T_c}\right) \arg \max \frac{1}{k} \sum_{K=1}^k \log_2 \times \left(1 + SNR \left| \sum_{l=1}^I \mathbf{h}_{l,K}^H \mathbf{f}_{\text{RF}}^l \mathbf{f}_{\text{BB}}^{l,K} \right|^2\right), \quad (5)$$

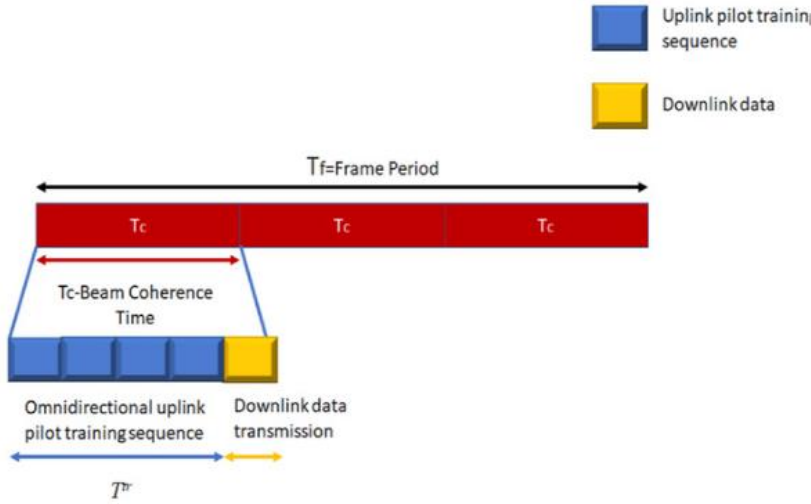


Figure 3-2: Beam coherence time interval during training process

Figure 3-2 depicts the training period for deep neural network-based learning. In each T_c UE transmits pilot matrix $\mathbf{S}_p^K \in \mathbb{C}^{N \times L}$ repeatedly, here $K = 1, 2, 3, \dots, k$. The received pilot sequence at l^{th} IAB is as in Equation (6):

$$\mathbf{x}_{l,K}^p = \mathbf{h}_{l,K} \mathbf{S}_p^K + \mathbf{n}_{K,l}, \quad (6)$$

The combined beamforming strategy starts with feeding the received pilot signals from all IABs to the Fusion Centre (FC) comprising DLIRL. The FC first selects the RF beamform vectors for the IABs downlink as:

$$\mathbf{f}_{\text{RF}}^l = \arg \max_{\mathbf{f}_{\text{RF}}^l \in \kappa_{\text{RF}}, \forall l} \sum_{K=1}^k \log_2 \left(1 + SNR \left| \sum_{l=1}^I \mathbf{h}_{l,K}^H \right|^2\right), \quad (7)$$

Here, κ_{RF} is the RF beamform codebook. The FC then applies baseband precoder calculated as [14]:

$$\mathbf{f}_{\text{BB}}^* = \frac{(\mathbf{h}_K^H \mathbf{F}_{\text{RF}})^H}{\|\mathbf{h}_K^H \mathbf{F}_{\text{RF}}\|}, \quad \forall K \quad (8)$$

The DNN is fed with the pilot symbols $\mathbf{x}_{l,K}^p$ as the input for the training model. The spectral efficiency for every RF and baseband beamform vector acts as the output shown in Equation (5). The DNN is trained during this period and learns the correlation between received pilot

symbols at each IABs and the ET. Post-training the DNN moves to the working model where it performs the prediction. The detailed DNN architecture is explained in Section 3.3.4.2.

3.3.4.2 DNN architecture

A multilayer DNN structure is used for training the proposed model and getting adapted to the scenario. The inputs to the model are the channel impulse response (CIR), AoA, and TDoA. The CIR, AoA, and TDoA are obtained via received pilot symbols at I IABs. The DNN structure comprises four hidden layers. The first layer has 12 nodes and the remaining 3 layers have 10 nodes each. The DNN is trained end to end as a supervised learning class [16]. The DNN input is normalized based on the training dataset. The DNN architecture employed Rectifier Linear Unit (ReLU) as the activation function [17, 18].

3.3.4.3 Working period

Based on the estimated channel, the RF and the Baseband precoding beamform codebook $\mathcal{K}_{\text{RF}}, \mathcal{K}_{\text{BB}}$ is respectively formed using Equations (7) and (8). The objective of the DNN is to maximize the $S_e(\mathbf{f}_{\text{RF}}^l, \mathbf{f}_{\text{BB}}^l)$ for each IABs. The regression-based learning model is adapted such that for each l^{th} IAB, the error difference between the DNN's predicted output $S_e^{\text{pred}, p, l}$ and the desired $S_e^{\text{des}, p, l}$ output is minimum. Here, $p = 1, 2, \dots, T_{\text{rain}}$, the T_{rain} is the number of RF and the baseband beamforming vectors. Mathematically DNN is trained to minimize the error function $e(w)$ for the set of different weight values of the DNN. The $e(w)$ for the l^{th} IAB can be written as in Equation (9):

$$e_l(\theta) = \sum_{p=1}^{T_{\text{rain}}} M_{se}(S_e^{\text{des}, p, l}, S_e^{\text{pred}, p, l}), \quad (9)$$

Here $M_{se}(S_e^{\text{des}, p, l}, S_e^{\text{pred}, p, l})$ is the mean square error (MSE) between the predicted output $S_e^{\text{pred}, p, l}$ and the desired $S_e^{\text{des}, p, l}$ output. The optimal RF and the baseband precoding vectors get updated in the beamform codebook $\mathcal{K}_{\text{RF}}, \mathcal{K}_{\text{BB}}$ respectively. From this stage onwards, the work of the Integrated RL (IRL) begins. The IRL employs the deep Q network for fine-tuning the beamform. The IRL goes through the optimized RF and baseband beamform codebook to further fine-tune the beamform vectors and steer the beam more precisely to the user location. The deep RL network employed in this work is shown in Figure 3-3.

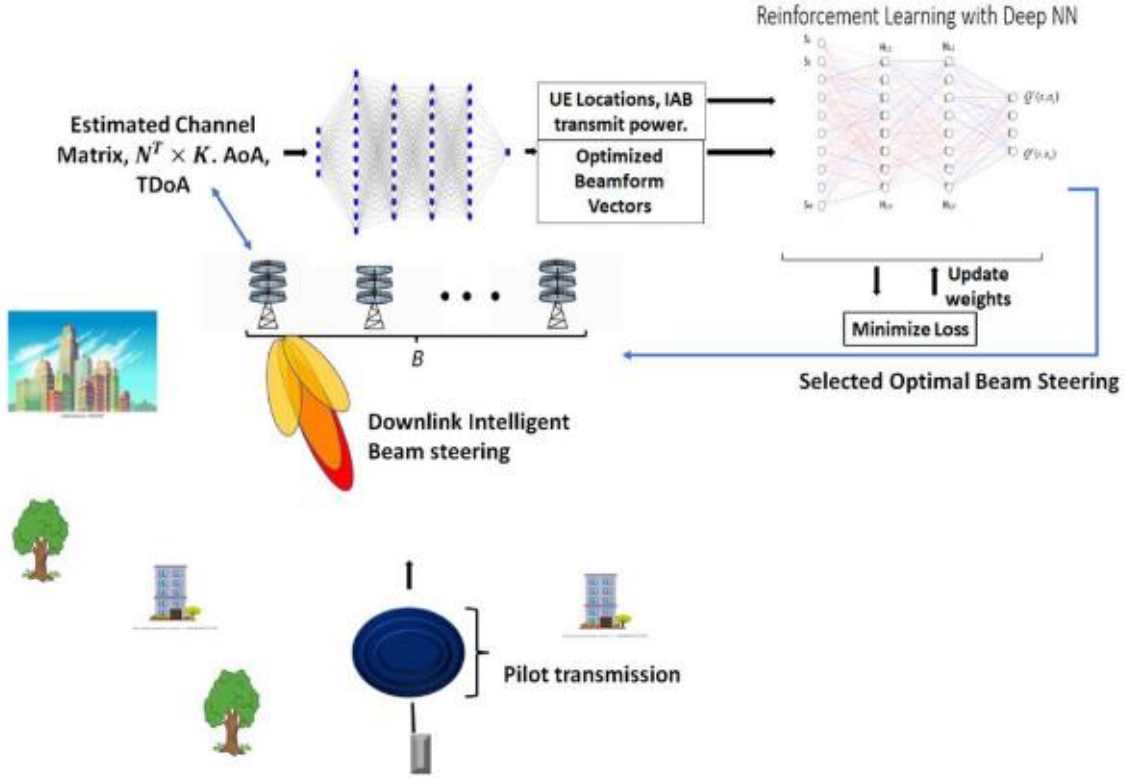


Figure 3-3: The DLIRL beamforming system model

The objective of the IRL network is to maximize the received UE's SNR for the given beamform vectors. The received power at UE for the time instant t from the l^{th} IAB can be denoted as:

$$P_{w_{UE}}^l(t) = P_{w_{IAB}}^l(t) \left| \mathbf{h}_{l,K}^H(t) \mathbf{f}_{RF}^l(t) \mathbf{f}_{BB}^l(t) \right|^2, \quad (10)$$

Here $P_{w_{IAB}}^l(t)$ is the transmitted power associated with the l^{th} IAB at time instant t . Now the received UE's SNR for the signal from the l^{th} IAB can be written as:

$$SNR^l(t) = \frac{P_{w_{IAB}}^l(t) \left| \mathbf{h}_{l,K}^H(t) \mathbf{f}_{RF}^l(t) \mathbf{f}_{BB}^l(t) \right|^2}{\sigma_n^2}, \quad (11)$$

The maximization problem for the IRL is as shown in Equation (12) such that the transmission power of each IAB should be within the total permissible power of each IAB i.e. *Total power*.

$$\begin{aligned} & \underset{\substack{P_{w_{IAB}}^l(t) \\ \mathbf{f}_{RF}^l(t) \mathbf{f}_{BB}^l(t)}}}{\text{maximize}} \sum_{l=1,2,3..N} SNR^l(t), \\ & \text{s.t. } P_{w_{IAB}}^l(t) \in \text{Total power}, \\ & \mathbf{f}_{RF}^l(t) \mathbf{f}_{BB}^l(t) \in \kappa_{RF}, \kappa_{BB}, \end{aligned} \quad (12)$$

An effort is made to fine-tune beam steering via DRL fed with the DNN optimized beamform vectors, UE location, and IAB transmit power to jointly control them to maximize Equation (12). The IRL is an efficient deep learning Q network. The state-space τ associated with the IRL are state₁ indicating the transmission power of the l th IAB, state₂, which is the beamform vectors associated with the l th IAB. The state₃ indicates the deployed UE locations employed in the training. Action space A involves regulation in the transmission power and beamform vectors from the codebook K_{RF}, K_{BB} of the respective IABs.

3.3.4.4 Deep IRL model

The deep IRL involves training of the Q Neural Network (NN) as shown in Figure 3-3. For the policy po the value of the state s and action a is given as $Q_{po}(s, a)$. For converging to the optimal state-action value $Q_{po}^*(s, a)$ we employed the NN architecture. The NN is defined by its weight values as \mathbf{we}_t for each time step t . The $\text{vec}(\mathbf{we}_t)$ is represented as \mathbf{we}_t . The state action value in terms of NN weights can be represented as $Q_{po}(s, a : \mathbf{we}_t)$ as shown in Algorithm 1.

ALGORITHM 1: Deep learning integrated reinforcement learning

1. Parameter Initialization: T, j, s, a
2. Input: UEs' SNR
3. Output: Beamform steering weights
4. for ($j=1; j<T; j++$)
5. Current s observation
6. Choose exploitation (exi) or exploration (exo) based on s
7. if exo
8. Select an a randomly from the set of a_s
9. else
10. Obtain $a = \text{argmax}_{a'} Q_{po}(s, a' : \mathbf{we}_t)$
11. end
12. Compute Reward
13. Obtain SINR
14. if obtained SINR < threshold
15. Abort
16. end
17. Next s is observed
18. Estimate
19. $o_t = \epsilon [\text{reward}_{s,s',a} + \text{discount} * \max_{a'} Q_{po}(s', a' : \mathbf{we}_{t-1}) | s_t, a_t]$
20. Perform DL training
21. Estimate and update weights of the DLIRL
22. Estimate the Mean Square Error:
23. $\min_{\mathbf{we}_t} \text{Err}(\mathbf{we}_t) = \epsilon_{s,a} [(o_t - Q_{po}(s, a : \mathbf{we}_t))^2]$
24. Calculate SINR
25. Estimate reward based on the SINR
26. end

The NN architecture of the deep IRL has the activator in the form of sigmoid function as $y \rightarrow \frac{1}{1+\exp^{-y}}$ [10]. Here the objective function for the NN of the IRL is to minimize the mean square error (MSE) represented as:

$$\min_{\mathbf{we}_t} \text{Err}(\mathbf{we}_t) = \epsilon_{s,a} [(o_t - Q_{p0}(s, a : \mathbf{we}_t))^2], \quad (13)$$

Here,

$$o_t = \epsilon [\text{reward}_{s,s',a} + \text{discount} * \max_{a'} Q_{p0}(s', a' : \mathbf{we}_{t-1}) | s_t, a_t], \quad (14)$$

The $\text{reward}_{s,s',a}$ is the reward for the agent post taking the action a and moving from state s at time t to state s' . The a' represents the next action to be taken. The weights of the NN are updated based on the gradient descent as:

$$\mathbf{we}_{t+1} = \mathbf{we}_t - \text{step} \nabla \text{Err}(\mathbf{we}_t), \quad (15)$$

Here, *step* indicates the step size has a value between 0 and 1. In the proposed IRL, the value $Q_{p0}^*(s, a)$ is estimated based on the approximation $Q_{p0}(s, a : \mathbf{we}_t)$ that minimizes the MSE $\text{Err}(\mathbf{we}_t)$. The proposed scheme is implemented corresponding to the downlink scenario. Here, IABs are separated at an estimated distance, and user equipment is positioned at a specific geographical location within the coverage of IABs. Also, these UEs move at a particular velocity. In this work, it is fixed at 20 mph. The reward function for the IRL is estimated based on the performance of an action on meeting the threshold *SINR*. The maximum reward at a unit time step is assigned to the agent having the best performance.

3.4 Experimental Procedure

The beamforming is implemented for the downlink scenario for the data transmission from IAB to UE. The simulation environment is first configured and then the proposed DLIRL beamforming is executed. The simulation parameters employed in this work are as shown in Table 3-1. The data set of the channel model and the channel parameters for the simulation is generated via MATLAB 2021 site viewer-based ray tracing. The Shooting and Bouncing Ray (SBR) based ray-tracing model is employed for the LoS and NLoS communication. The orthogonal frequency division multiplexing (OFDM) system is employed for symbol transmission. The considered OFDM size is 1024.

Table 3-1: Simulation Parameters

S. no.	Parameter	Specification
1	IABs/IAB count (N)	4
2	IAB antenna array	Uniform planar array
3	IAB Antenna Specification	32×8
4	User equipment (UE) setup	Deployed in a rectangular grid of dimension $40 \text{ m} \times 60 \text{ m}$, resolution 0.1 m
5	DNN activation unit	ReLU (Rectified Linear Unit)
6	DNN dropout rate	0.5%
7	DNN batch size	100
8	Python Libraries	Keras with Tensorflow backend
9	System bandwidth	0.5 GHz
10	OFDM subcarriers	1024
11	Sampling Factor	1
12	Multipaths	7

The DNN architecture has a total of 6 interconnected layers including 4 hidden layers and 1 input and an output layer. The DNN has a total $l \times k$ number of inputs and T_{train} number of outputs. The considered data has a set size of two hundred thousand samples and a batch size of two hundred.

3.5 Experimental Results and Analysis

To have a comparative analysis of the proposed algorithm with the existing conventional beamforming techniques, we have used SE and the BER as the metrics. Figure 3-4 and Figure 3-5 show the Spectral Efficiency (SE) for different SNR values received at UE. The simulations were carried out for 30 runs comprising 1000 iterations each. The depicted graph values are averaged values obtained in the simulation environment. For the simulation environment, the IABs are installed on the buildings played in the x - y plane of the 3D environment. The IAB's antenna is facing the street on the y - z plane. The antenna transmit power is considered at 30 dBm. The UEs are mobile and are installed with a single antenna. For each beam coherence time, the UE locations are updated in the x - y plane. During the training period, the UE uplink transmit power is set at 30 dB m.

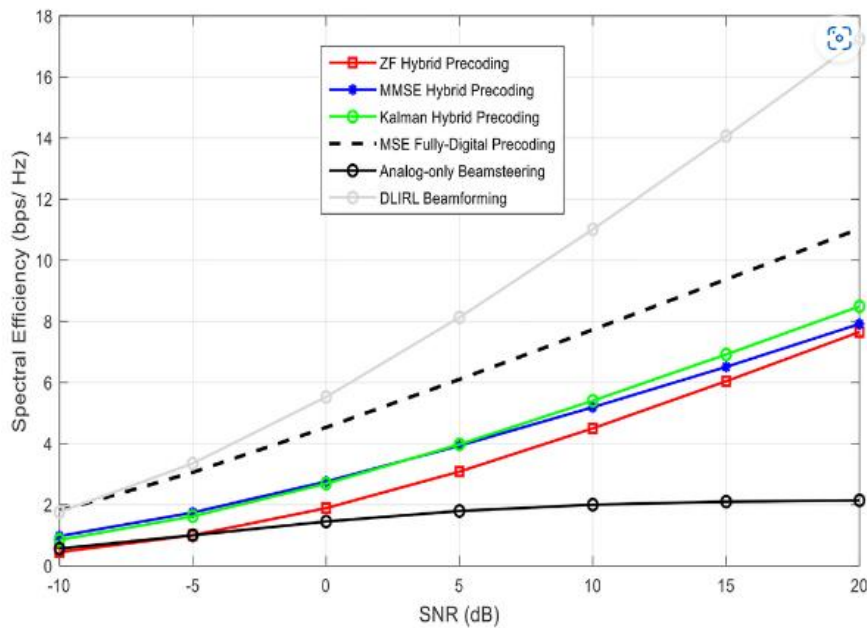


Figure 3-4: Comparative analysis of SE with reference to IAB SNR

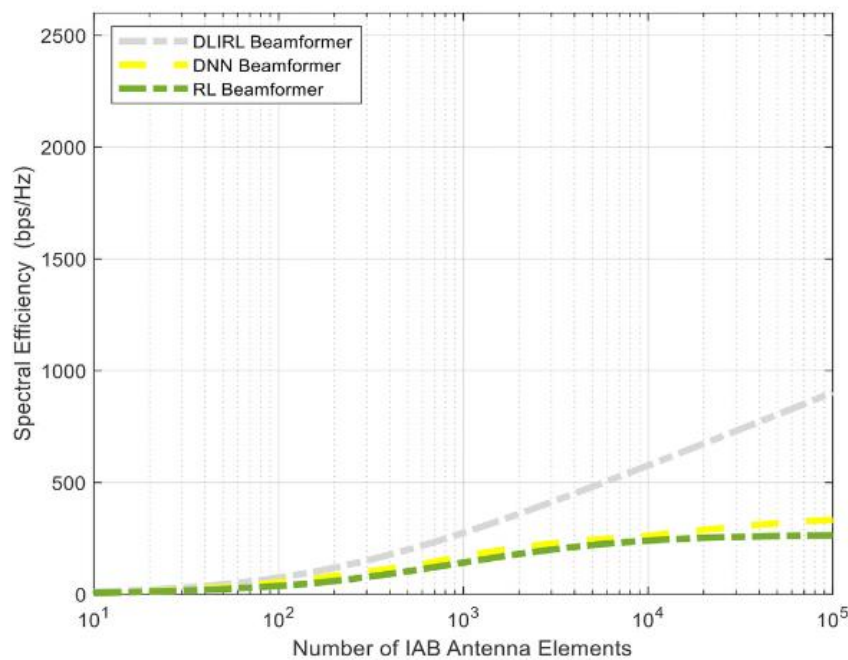


Figure 3-5: Comparative analysis of SE with reference to increase in number of antennas

From Figure 3-4, it is visualized that the DLIRL beamforming has achieved better spectral efficiency as compared to the existing conventional beamformers in [19]. As seen from the curves, the spectral efficiency with analogue beamforming is found to be around 2 bps/Hz, and close convergence is observed between ZF hybrid precoding, MMSE hybrid precoding, and Kalman hybrid precoding techniques. However, the MSE-based fully digital precoding displays improved spectral efficiency as compared to the above-mentioned precoding techniques. For an SNR of 5 dB, the DLIRL based beamforming technique displays an improvement of 77.5%, 60%, 50%, 50%, and 33.3% as compared to the analogue beamforming, ZF hybrid precoding, MMSE hybrid precoding, Kalman hybrid precoding, and MSE-based fully digital precoding techniques, respectively. The spectral efficiency is achieved for the multipath

scenario considering both LoS and NLoS, total multipath considered for evaluation of Figure 3-4 is 10 and total IAB antenna elements are 256.

Moreover, as the antenna size increases the performance of DLIRL gets better as compared to the DNN and RL beamformer. Figure 3-5 shows the comparison between DNN, RL, and DLIRL-based beamformers for different transmitting antenna elements. For instance, for the number of IAB antenna elements equal to 10^4 , the increase in spectral efficiency employing DLIRL-based beamforming is found to be 53.33% and 51.66% more efficient as compared to DNN- and RL-based beamforming techniques, respectively. The effect of BER for IAB with 4 transmit antenna elements has been displayed in Figure 3-6. Here, the performance of the system for different MIMO schemes is compared with DLIRL-based beamforming scheme. For a BER of 10^{-4} , the proposed DLIRL-based beamforming techniques require an E_b/N_0 of 7 dB. Alternatively, the system without diversity, Alamouti, and OSTBC schemes require an E_b/N_0 of 10 dB, 13.3 dB, and beyond 20 dB, respectively.

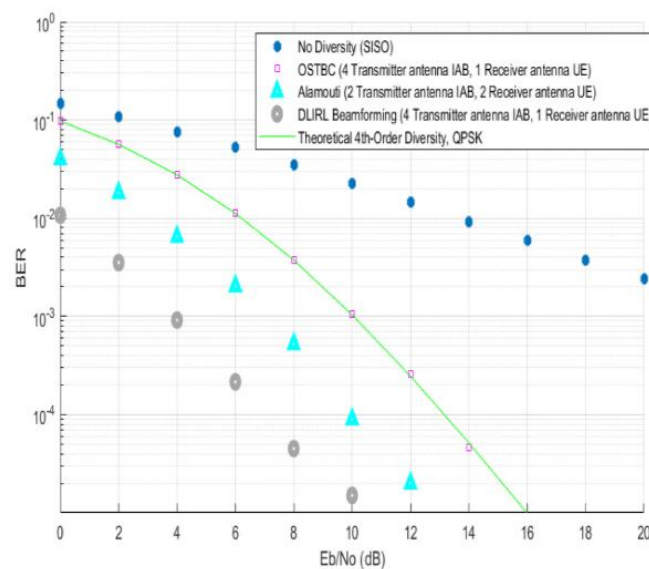


Figure 3-6: Comparative analysis using BER

The quantitative analysis of the proposed scheme concerning SE and BER is presented in Table 3-2 and Table 3-3 respectively. There is a drastic improvement in the SE and BER using DLIRL beamforming studied at SNR = 10 dB shown in both the tables.

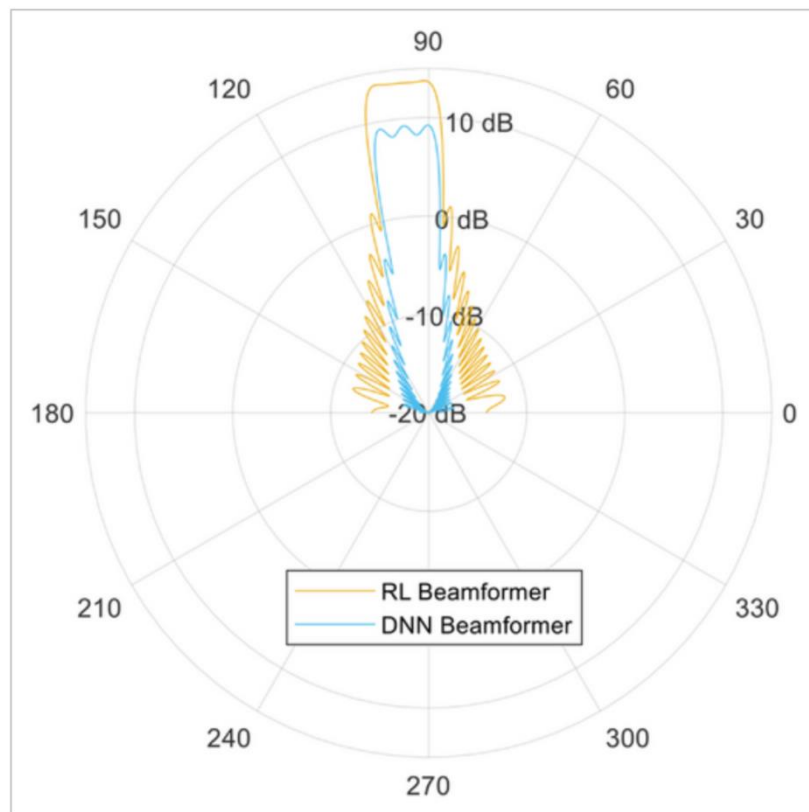
Table 3-2: SE performance evaluation

Beamforming Technique	SE (bps/Hz) at SNR = 10 dB
DLIRL beamforming	11
Analog beamforming	5
MSE digital beamforming	8
Kalman-hybrid precoding	5.7
Minimum mean squared error (MMSE) hybrid beamforming	5.1
Zero forcing hybrid beamforming	4.2

Table 3-3: BER performance evaluation

Diversity scheme	BER at SNR = 10 dB
No diversity, Single input single output (SISO)	$10^{-1.8}$
Orthogonal space time block coding (OSTBC), 1×4 MIMO transmit diversity	10^{-3}
Alamouti, 2×2 MIMO diversity	10^{-4}
DLIRL beamforming, 1×4 Transmit diversity	10^{-5}

To answer the question of whether the proposed scheme can learn the beamforming, we have simulated Figure 3-7 and Figure 3-8. The simulation graphs in Figure 3-7 and Figure 3-8 are also the report of the average value for the 30 runs with 1000 iterations each. The proposed scheme can project the beam towards the UE positioned at 100 degrees in the northwest direction of the antenna placement. From Figure 3-7 and Figure 3-8, it is estimated that the DLIRL based beamformer is better than its counterparts DNN and RL in steering the beam towards the UE placed at 101.5° normal to the antenna placement of IAB. The proposed DLIRL beamforming has Angle of Departure (AoD) towards UE location with a deviation of $\pm 2^\circ$, whereas RL has a deviation of $\pm 3^\circ$ and DNN's deviation is $\pm 5^\circ$.

**Figure 3-7: Beamform towards UE using DNN and RL separately**

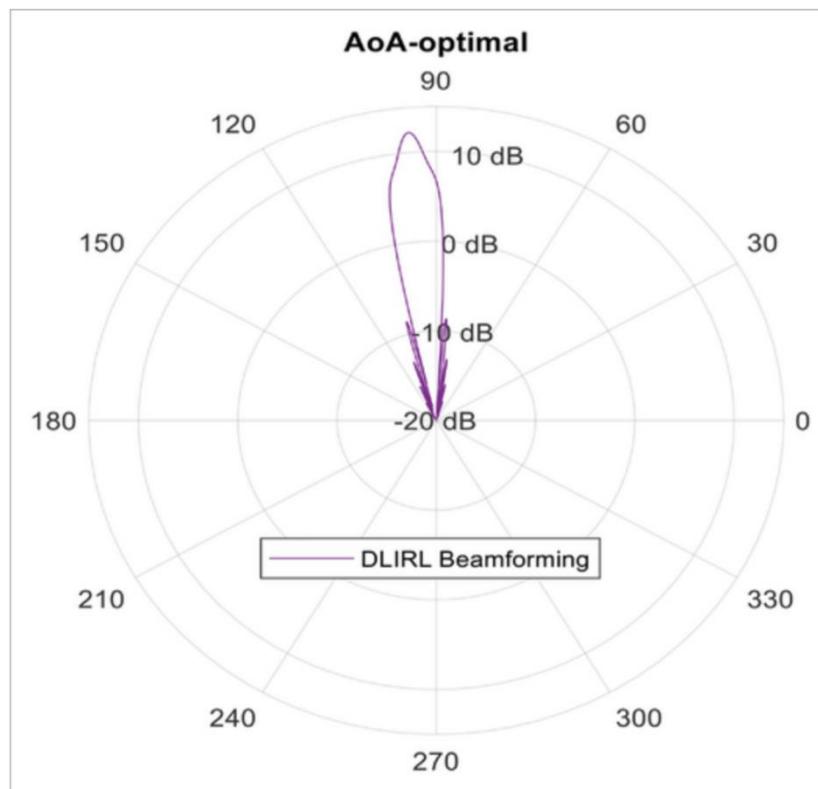


Figure 3-8: Beamform towards UE using DLIRL

The DLIRL is capable of performing efficient beamforming due to the effective training. It is vital to have comparative analysis of the DLIRL with existing DNN and RL algorithm in terms of training validation accuracy, training loss, number of iterations and epochs. Figure 3-9 sheds light on the validation accuracy of the proposed (DLIRL) and existing (DNN and RL) training algorithms. For the training we employed 20 epochs, 160 iterations, and 100 runs. Each run comprised 20 epochs and each epoch have 8 iterations. From the validation accuracy as shown in Table 3-4 and Figure 3-9: Training validation accuracy of DLIRL, DNN and RL it can be inferred that the proposed DLIRL due to its optimized amalgamation of DNN and RL has better training accuracy as compared to the DNN and RL. These training accuracy results are clearly reflected in the beamforming effectiveness as shown in Figure 3-8: Beamform towards UE using DLIRL.

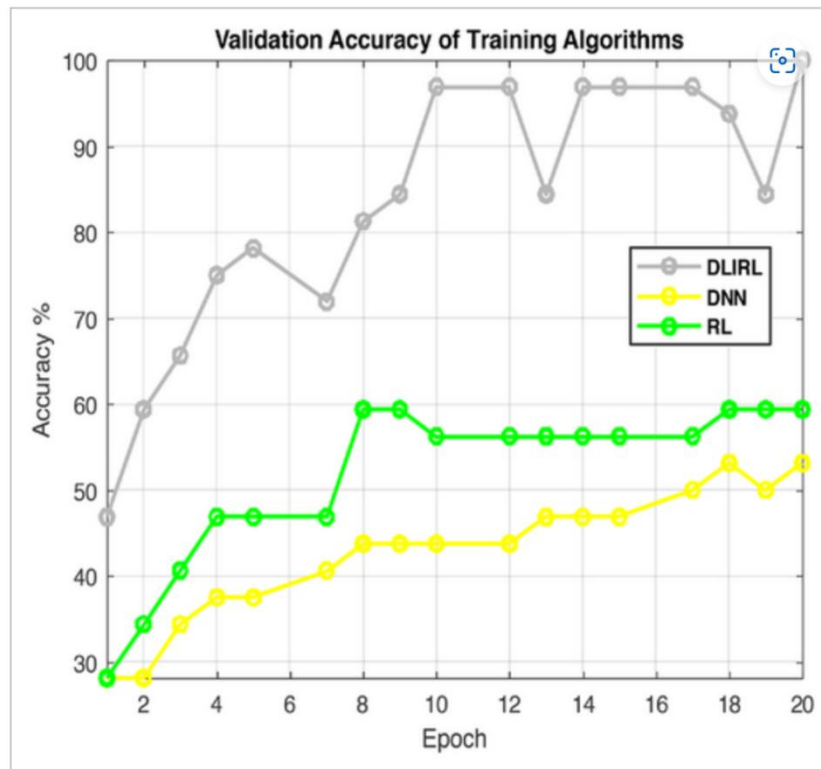


Figure 3-9: Training validation accuracy of DLIRL, DNN and RL

Table 3-4: Performance evaluation of the training algorithm

Epoch	Iteration	Validation accuracy (RL)	Validation accuracy (DNN)	Validation accuracy (DLIRL)
1	1	28.12%	28.12%	46.88%
2	10	34.38%	28.12%	59.38%
3	20	40.62%	34.38%	65.62%
4	30	46.88%	37.50%	75.00%
5	40	46.88%	37.50%	78.12%
7	50	46.88%	40.62%	71.88%
8	60	59.38%	43.75%	81.25%
9	70	59.38%	43.75%	84.38%
10	80	56.25%	43.75%	96.88%
12	90	56.25%	43.75%	96.88%
13	100	56.25%	46.88%	84.38%
14	110	56.25%	46.88%	96.88%
15	120	56.25%	46.88%	96.88%
17	130	56.25%	50.00%	96.88%
18	140	59.38%	53.12%	93.75%
19	150	59.38%	50.00%	84.38%
20	160	59.38%	53.12%	100.00%

The proposed DLIRL is effective in getting trained in less number of samples as shown in Figure 3-9. The SE of DLIRL is comparable to DNN and RL for very few samples (100) as visualized in Figure 3-11. Above the 100 training samples the performance of the DLIRL is better than its counterparts. Another important discussion associated with this paper is that the proposed beamforming scheme is adaptive beamforming based on the UE's movements. To test the efficacy of the DLIRL beamforming, simulation environment considered MATLAB 2021

employing a site viewer. For ray tracing, Hong Kong Open Street Map (OSM) with the 3D building environment is employed as shown in Figure 3-10 a, b.

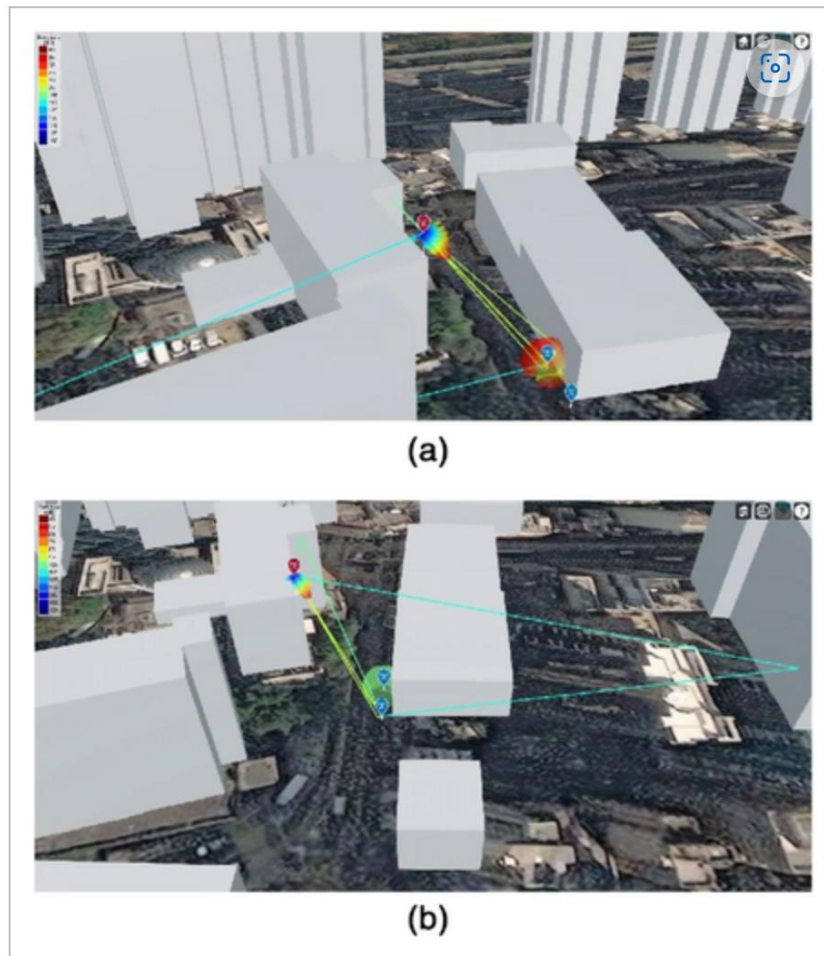


Figure 3-10: (a) Demonstration of DLIRL based adaptive beamforming using MATLAB raytracing and siteviewer. (b) DLIRL based beamsteering with user movement

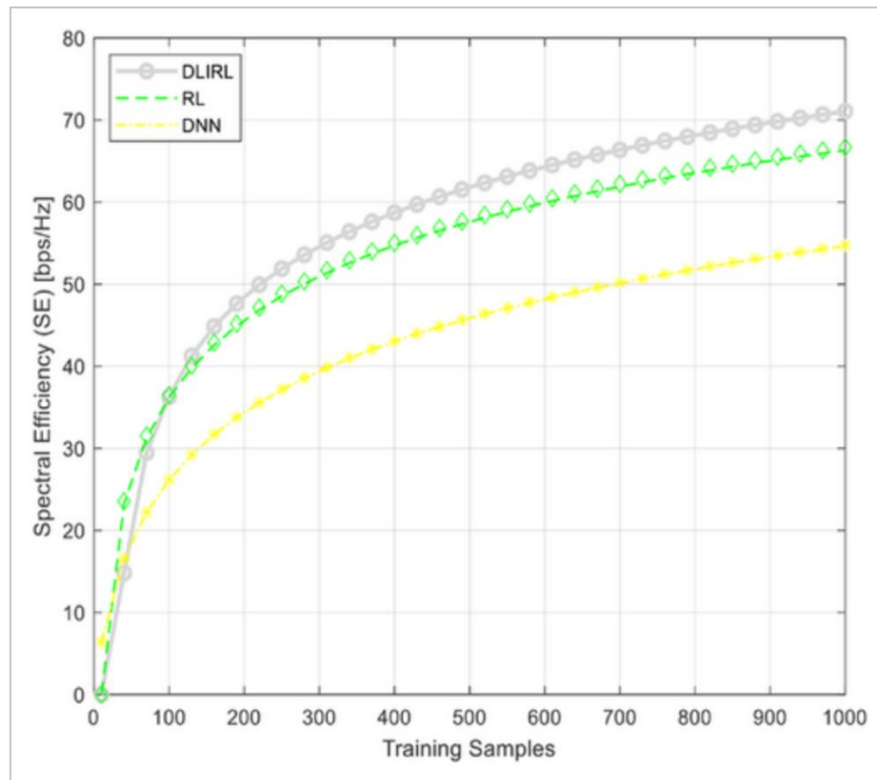


Figure 3-11: SE versus training samples for DLIRL, RL and DNN

The latitude and longitude associated with an IAB are 22.287495, 114.140706. The initial UE location is 22.287323, 114.140859 latitudes, and longitude respectively. The UE is moving at a constant speed of 28 kilometres/hour (kph) and its position changes 22.2874, 114.140859 latitude, and longitude respectively as seen in Figure 3-9. From the figure, the proposed method tracks the user equipment and steers the beam in its direction with an accuracy of 92%. Accuracy is measured based on the performance of DNN and RL in minimizing the error deviation achieved at the UE while moving to a new position from the initial position.

4 OWC measurement campaigns

4.1 Aim of Experiment

The main objective of the testbed is to provide a realistic visible light positioning testing environment for those experimenters 1) who are working on optical wireless access technology. This testbed offers an indoor visible light communication wireless access. It can be used for analysing and demonstrating that an innovative and data/energy/cost efficient communication system; 2) who are working on indoor positioning algorithm and AI based positioning algorithm. The testbed provides a realistic measurement dataset in different indoor environment.

4.2 Background Theory:

4.2.1 impact on experimenters

In April 2020, the General Office of the Ministry of Industry and Information Technology issued a notice on in-depth promotion of the comprehensive development of the mobile Internet of Things (IoT). It explicitly pointed out that we should accurately grasp the global evolution trend of technology standards and industrial pattern of mobile IoT. 2G/3G IoT networking services would be migrated, and a comprehensive mobile IOT ecosystem would be established with the coordinated development of NB-IoT (Narrowband Internet of Things), 4G (including LTE-Cat1, that is, 4G network of rate category-1) and 5G. With the rapid development of the Internet of Things technology, the indoor positioning market has gradually expanded. Meanwhile, with technical advantages such as high accuracy and low radiation, the visible light indoor positioning technology has become a powerful indoor positioning market competitor.

Due to the large number of production lines and the frequent entry and exit of materials in modern factories, it is necessary to always keep track of the quantity of personnel and materials, so as to better coordinate and allocate resources to achieve the goal of efficient production. In the past, these data were manual statistics, which were inefficient, untimely and error-prone, and it was not easy to obtain support from front-line operators. Through the use of indoor positioning technology, these data can be counted continuously, automatically and in real time, allowing engineers and experts to obtain big data in the production and operation process with unprecedented efficiency, and quickly draw operational optimization recommendations through cloud analysis tools.

In libraries, museums, and various transportation ports, it is necessary to make statistics on the current flow of people to prevent stampede incidents in public places. For large indoor places, install a visible light positioning system to count the density of people in the unit in real time, or at certain intervals, and then generate visual data after processing by the host computer, which is convenient for managers to arrange the layout, evacuation, opening hours, opening methods and other issues of indoor places.

With the development of society, people's living standards have gradually improved, and they have begun to pursue a more convenient life. Smart service markets such as smart supermarkets and smart hospitals are also booming. Indoor positioning technology can realize real-time positioning of people in indoor places. Through handheld terminals, relevant information can be sent to customers to facilitate their activities in supermarkets, hospitals

and other places. By use of visible light positioning system, people can enjoy a more convenient lifestyle.

4.2.2 Introduction of 3D VLP algorithm

When the receiver is placed in the positioning space, we need to estimate its abscissa value and its height to obtain their localization information. In general, 2D localization algorithms suffice when we know at what height the receiver is placed. However, in some cases we don't know the height information of the receiver, so we need a 3D positioning algorithm to estimate the position of the receiver in space.

We proposed a 3D environment-aware VLP algorithm based on the positioning algorithm in 2D in the previous chapter. The difference from 2D positioning is that in 3D positioning, not only the horizontal and vertical coordinates of the receiver are estimated, but also the height at which it is located.

Therefore, the idea of the algorithm we propose is: first estimate the height of the receiver, and then use the algorithm we proposed in previous chapter to estimate the horizontal and vertical coordinates of the receiver in 2D VLP algorithm.

The framework of the proposed algorithm is illustrated in Figure 4-1, which contents three steps: 1) Environment index classifier group establishment, 2) Receiver's height estimation and 3) Positioning. In the first step, we will set up a candidate group of heights and build an environmental indicator classifier at each candidate height, combining them together into a classifier group. Then in the second step, we will use an optimize algorithm to estimate the height of the receiver using the classifier group. Finally, in the third step, we will apply the 2D positioning algorithm in the height of the estimation result in the previous step.

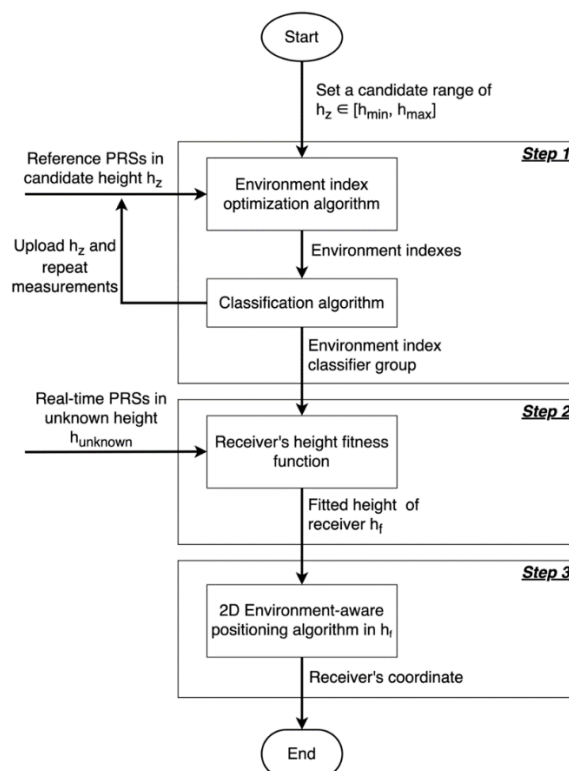


Figure 4-1: The framework of proposed 3D environment-aware Visible light-based positioning algorithms.

Step 1: Environment index classifier group establishment

The objective of the first step was to establish a group of environment index classifier in different height of receiver. Firstly, the minimum height h_{\min} and maximum height h_{\max} of the receiver should be determined depended on the available height of the positioning space. For example, if the case where the receiver is held by people is considered, it could be determined as between 0 m and 2 m according to the height of a human. Therefore, the candidate group H_C of the height of the receiver can be given as in

$$H_C = \{h_i | h_i \in [h_{\min}, h_{\max}], 1 \leq i \leq z\} \quad (16)$$

Here, h_i represents the i^{th} candidate height in the set H_C , and h_{\min} and h_{\max} represent the minimum and maximum values of candidate heights in the set, respectively. z represents the number of candidate heights in the set, which depends on the user's needs. If a large value of z is considered, the interval between every two adjacent candidate heights becomes small, therefore the algorithm may give multiple possible estimated height. In this situation, an additional strategy is necessary in which to determine the estimated height of the receiver. Such a process will bring a great algorithmic complexity. On the contrary, if the interval between adjacent candidate height is increased and few candidate heights are selected, the complexity of the algorithm can be reduced. As a consequence, the algorithm can only give few possible values for the height estimate and the estimated height of the receiver is only an approximation and cannot be accurately determined. Therefore, how to determine the value of z needs to be considered according to actual needs.

Next, in each height h_z , we determined K reference points with known coordinates to place the APD receiver. It was placed in k^{th} ($k \in [1, K]$) reference points to collect the light signals emitted from n^{th} ($n \in [1, N]$) LED. The received light signals on APD were considered as PRS P_{rnk} . In this paper, we did not focus on how to distinguish these signals, but focused on the localization algorithm. A set S_z was defined to represent these PRSs as in

$$S_z = \begin{pmatrix} P_{r11} & P_{r12} & \dots & P_{r1K} \\ P_{r21} & \dots & \dots & \\ \dots & \dots & \dots & \\ P_{rN1} & \dots & \dots & P_{rNK} \end{pmatrix} \quad (17)$$

After the measurement from all reference points in height h_z , their corresponding environment indexes λ_{nk} were optimized and resulted by using Particle Swarm Optimization (PSO) algorithm, which have been introduced in previous chapter. It consists of two parameters: m and G represent the compensation of Lambertian distribution and extra pathloss caused by random angle interference respectively as in

$$\lambda_{nk} = \{m_{nk}, G_{nk}\} \quad (18)$$

Then, Λ_z is considered as the group of all value of λ as in

$$\Lambda_z = \left\{ \begin{array}{cccc} \lambda_{11} & \lambda_{12} & \dots & \lambda_{1K} \\ \lambda_{21} & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots \\ \lambda_{N1} & \dots & \dots & \lambda_{NK} \end{array} \right\} \quad (19)$$

Here, the sets Λ in (19) and S in (17) were applied to train the classification model M_z in the height h_z as in

$$M_z : S_z \rightarrow \Lambda_z \quad (20)$$

Here, k-Nearest Neighbors (kNN), one of the most common classification methods was selected as the classification algorithm to build M_z . This classification algorithm uses the distances between data points to classify new data and assigns a class label based on the majority class of the K nearest neighbors in the training dataset. Then, it votes on these neighbors and determines the classification result. One of the reasons to use kNN is that it is efficient on large training data, robust to noisy training data, while remaining simple and transparent, which fits the classification needs of our dataset with a large number of labels Λ .

When M_z was successfully established in height h_z , h_z was uploaded in the range of $[h_{\min}, h_{\max}]$ and built its corresponding M_z by the same way. Once all candidates of $h_z \in H_c$ have been iterated a group of classifiers M_c were established as in

$$M_c = \{M_1, M_2, \dots, M_z\} \quad (21)$$

which is necessary for the next step.

Step 2: Receiver's height estimation

The objective of this step is to determine the height of the receiver.

The flow chart of this step is illustrated in Figure 4-2. When the receiver was placed on an unknown location in the positioning space, we have no idea about its height h_{unknown} . During the measurement, real-time PRSs $P_{r_{\text{unknown}}}$ were collected and sent to the environment index classifier group M_c generated in the previous step, which included all classifiers from all candidate heights.

Here, PRSs $P_{r_{\text{unknown}}}$ became the input for each classifier M_z and got the λ_z prediction results from them, grouped as in Λ_p :

$$\Lambda_p = \{\lambda_1, \lambda_2, \dots, \lambda_z\} \quad (22)$$

To determine which h_z of the candidate heights H_c was considered as the predicted height h_f of the receiver, we used $P_{r_{\text{unknown}}}$ and λ_z in each height to apply the positioning process. Then, the λ_z in h_z that leads to the smallest PE would be chosen, and its corresponding h_z was considered as the height of the receiver. Here, PE is defined as the Euclidean distance between the receiver's real coordinate and the estimated coordinate.

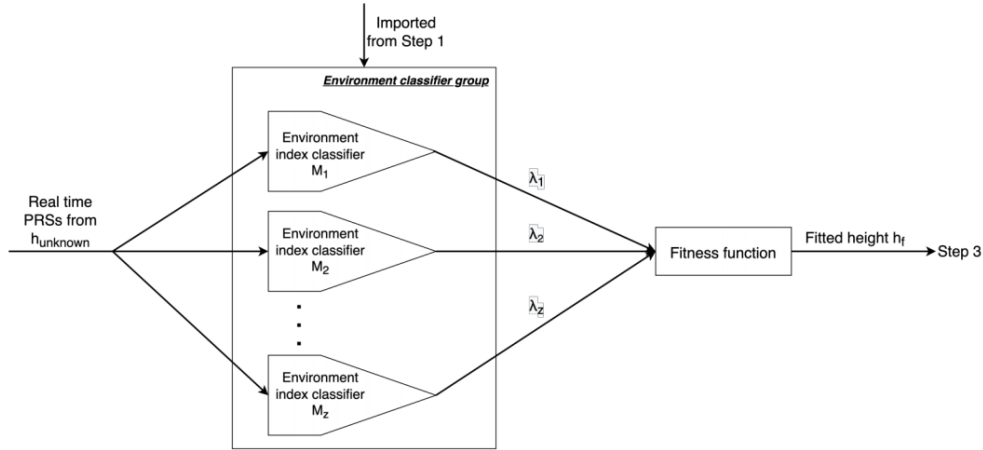


Figure 4-2: The framework of Step 2: Receiver's height estimation

The pseudo code of the fitting function is detailed in Algorithm. 1. At the beginning, all the possible heights h_z were initialized as group H_c . Coordinates of LEDs were also be required as inputs. Then the values of λ were iterated over in Λ . For each λ , we first calculated the distances \hat{D}_{nk} between the n^{th} ($n \in [1, N]$) LED and k^{th} ($k \in [1, K]$) APD's reference point based on the OWC channel characteristic. It could be calculated as in

$$\hat{D}_{nk} = G_{nk} \left(\frac{(m + m_{nk} + 1) A_{PD} P_{tx} H^{(m+m_{nk}+1)}}{2\pi P_{rnk}} \right)^{\frac{1}{m+m_{nk}+3}} \quad (23)$$

Through these distances, the coordinate of APD's $(\hat{x}, \hat{y}, \hat{z})$ could be calculated based on (24) to (29), as in

$$\begin{cases} (x_1 - \hat{x})^2 + (y_1 - \hat{y})^2 + (z_1 - h_z)^2 = \hat{D}_{1k} \\ (x_2 - \hat{x})^2 + (y_2 - \hat{y})^2 + (z_2 - h_z)^2 = \hat{D}_{2k} \\ \vdots \\ (x_n - \hat{x})^2 + (y_n - \hat{y})^2 + (z_n - h_z)^2 = \hat{D}_{nk} \end{cases} \quad (24)$$

$$AX = B \quad (25)$$

$$X = \begin{bmatrix} \hat{x} \\ \hat{y} \\ h_z \end{bmatrix} \quad (26)$$

$$A = \begin{bmatrix} 2x_2 - 2x_1 & 2y_2 - 2y_1 & 2z_2 - 2z_1 \\ 2x_3 - 2x_1 & 2y_3 - 2y_1 & 2z_3 - 2z_1 \\ \vdots & \vdots & \vdots \\ 2x_n - 2x_1 & 2y_n - 2y_1 & 2z_n - 2z_1 \end{bmatrix} \quad (27)$$

$$B = \begin{bmatrix} (\hat{D}_{1k})^2 - (\hat{D}_{2k})^2 - x_1^2 + x_2^2 - y_1^2 + y_2^2 - z_1^2 + z_2^2 \\ (\hat{D}_{1k})^2 - (\hat{D}_{3k})^2 - x_1^2 + x_3^2 - y_1^2 + y_3^2 - z_1^2 + z_3^2 \\ \vdots \\ (\hat{D}_{1k})^2 - (\hat{D}_{nk})^2 - x_1^2 + x_n^2 - y_1^2 + y_n^2 - z_1^2 + z_n^2 \end{bmatrix} \quad (28)$$

$$\begin{bmatrix} \hat{x} \\ \hat{y} \\ \hat{h}_z \end{bmatrix} = (A^T A^{-1}) A^T B \quad (29)$$

where $(\hat{x}, \hat{y}, \hat{z})$ ($n \in [1, N]$) represents the known coordinate of n^{th} LED. At the same time, we could also calculate the Euclidean distance D'_{nk} between n^{th} LED and estimated APD's coordinate $(\hat{x}, \hat{y}, \hat{z})$. Here, for n^{th} LED, two distance results were compared: \hat{D}_{nk} and D'_{nk} . The difference ΔD_{nk} between them represented the accuracy of the estimated $(\hat{x}, \hat{y}, \hat{z})$: if this estimation was precise, the value of D'_{nk} , calculated by the OWC feature and D'_{nk} from Euclidean method should be very close.

During the iteration of λ in Λ_p , the minimum ΔD_{nk} and its corresponding h_z were stored. As the result of Algorithm. 1, h_f was considered as the estimated height of the receiver. So far, the 3D localization problem has been successfully transformed into a 2D localization problem at h_z , which was the objective in the next step.

Step 3: 2D positioning in h_f

Once the height of the receiver was determined in the previous step, the positioning process was converted from 3D to 2D in the estimated height h_f . The environment index group Λ_f was also applied in this step to estimate the horizontal and vertical coordinates of the receiver.

Firstly, the distance D'_{nk} between the n^{th} ($n \in [1, N]$) LED and k^{th} ($k \in [1, K]$) APD's reference point could be calculated by using Λ_f as in 23 and 30

$$\lambda_{nk} = \{m_{nk}, G_{nk}\}, \lambda_{nk} \in \Lambda_f \quad (30)$$

Algorithm 1 Fit the height of APD h_f

1: Initialization: candidate height group $H_c = \{h_z \mid h_{\min} \leq h_z \leq h_{\max}\}$, Classification

Results $\Lambda_f = \{\lambda_1, \lambda_2, \dots, \lambda_z\}$, coordinates of LEDs, $F_{\min} = \infty$

2: for λ in Λ_p do

3: calculate distance \hat{D}_{nk} between each LED and APD based on Equation (4.8)

- 4: calculate APD's coordinate $(\hat{x}, \hat{y}, \hat{z})$
- 5: calculate the Euclidean distance \hat{D}_{nk} between each LED and $(\hat{x}, \hat{y}, \hat{z})$
- 6: end for
- 7: calculate $\Delta D_{nk} = \hat{D}_{nk} - D'_{nk}$
- 8: if $\Delta D_{nk} < F_{\min}$ then
- 9: update $F_{\min} = \Delta D_{nk}$
- 10: update $h_f = h_z$
- 11: end if

Output: fitted height h_f

Secondly, the coordinate of the receiver in k^{th} ($k \in [1, K]$) APD's reference point could be estimated as $(\hat{x}, \hat{y}, \hat{z})$ by using equations listed as in

$$\begin{cases} (x_1 - \hat{x})^2 + (y_1 - \hat{y})^2 + (z_1 - h_f)^2 = \hat{D}_{1k} \\ (x_2 - \hat{x})^2 + (y_2 - \hat{y})^2 + (z_2 - h_f)^2 = \hat{D}_{2k} \\ \vdots \\ (x_n - \hat{x})^2 + (y_n - \hat{y})^2 + (z_n - h_f)^2 = \hat{D}_{nk} \end{cases} \quad (31)$$

$$AX = B \quad (32)$$

$$X = \begin{bmatrix} \hat{x} \\ \hat{y} \\ h_f \end{bmatrix} \quad (33)$$

$$A = \begin{bmatrix} 2x_2 - 2x_1 & 2y_2 - 2y_1 & 2z_2 - 2z_1 \\ 2x_3 - 2x_1 & 2y_3 - 2y_1 & 2z_3 - 2z_1 \\ \vdots & \vdots & \vdots \\ 2x_n - 2x_1 & 2y_n - 2y_1 & 2z_n - 2z_1 \end{bmatrix} \quad (34)$$

$$B = \begin{bmatrix} (\hat{D}_{1k})^2 - (\hat{D}_{2k})^2 - x_1^2 + x_2^2 - y_1^2 + y_2^2 - z_1^2 + z_2^2 \\ (\hat{D}_{1k})^2 - (\hat{D}_{3k})^2 - x_1^2 + x_3^2 - y_1^2 + y_3^2 - z_1^2 + z_3^2 \\ \vdots \\ (\hat{D}_{1k})^2 - (\hat{D}_{nk})^2 - x_1^2 + x_n^2 - y_1^2 + y_n^2 - z_1^2 + z_n^2 \end{bmatrix} \quad (35)$$

$$\begin{bmatrix} \hat{x} \\ \hat{y} \\ h_f \end{bmatrix} = (A^T A^{-1}) A^T B \quad (36)$$

Where $(x_n, y_n, z_n)(n \in [1, N])$ represents the known coordinate of n^{th} LED.

4.3 Simulation set up and results

4.3.1 Set up of the simulation

The setup of the simulation is illustrated in Figure 4-3. It was performed within a 5 m×5m×3 m area with four LEDs and one APD. As using four LEDs on the transmitter side can give the better positioning results compared to three LEDs [37]. They were mounted as the four corners of a square and in the center of the positioning area, which is a typical configuration used in many researches. In both two figures, four red dots represent LEDs in both figures, which were placed squarely on the ceiling at a height of 3 m from the ground. For the APD, its height range was determined to be between $h_{\min} = 0\text{m}$ and $h_{\max} = 2\text{ m}$ above the ground, as we consider it to be held by a human, or by a robot at approximately equal or lower height than a human. In addition to the LoS channel, we also add Gaussian white noise. The blue dots at different heights represent the reference position of the APD.

In our daily 2D testing, we observed that the normal line of the APD deviated no more than 3 degrees from its initial position in measurement. Hence, we assume that the angle of deviation between the normal vector of the APD and the plane containing the LED is an unmeasurable random angle, with an absolute value ranging between 0 and 3 degrees. Thus, in this simulation, we consider that the normal vector of the receiver is subject to a random angle between 0 and 3 degrees. The remaining key configuration parameters are listed in Table 4-1.

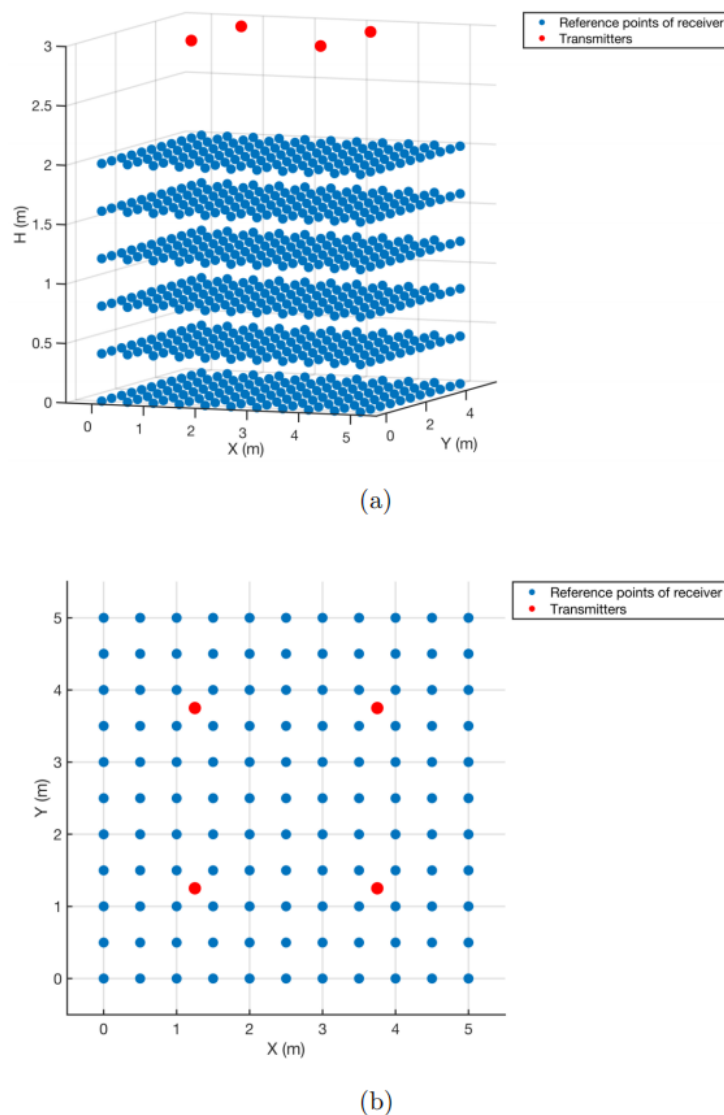


Figure 4-3: Simulation set up: placement of transmitters (red dots) and reference points of the receiver (blue dots) in (a) 3D view (b) 2D view

4.3.2 Simulation results of height estimation

As previously mentioned, successfully estimating the height of the receiver is one of the objectives in our proposed algorithm. Therefore, we first present the simulation results of receiver's height estimation.

Table 4-2 shows the height estimate results under varying receiver heights between 0m to 2m. Different rows in the table represent different height of the receiver h_{real} we put it at in simulation. The different columns represent the height estimate h_f given by the algorithm. When h_{real} is equal to h_f , it indicates that the height of the receiver has been estimated correctly. At each height, receivers were placed at 121 reference points, and the proposed algorithm estimate the height for each of them. The last column of the table calculates the proportion of heights correctly estimated by the algorithm out of the total of 121 for each height.

Table 4-1: Key configuration parameters of the simulation

Parameter	Value
Positioning space size (m^3)	$5 \times 5 \times 3$
Number of LEDs	4
Coordinate of LEDs (m)	LED1 (1.25, 1.25, 3) LED2 (1.25, 3.75, 3) LED3 (3.75, 1.25, 3) LED4 (3.75, 3.75, 3)
Transmitted power (W)	25
Number of receivers	1
Candidate heights of receiver (above the ground) (m)	{0, 0.4, 0.8, 1.2, 1.6, 2}
The semi angle at half power ($^\circ$)	70
Number of reference points of receiver at each candidate height	121
Range of angle of normal line of receiver ($^\circ$)	$[0, 3]$
Detector physical area of (m^2)	1×10^{-4}

Table 4-2: The height estimation results under varying receiver heights

		Estimated height of receiver h_f						Percentage of successful estimation
		0 m	0.4 m	0.8 m	1.2 m	1.6 m	2 m	
Real height of receiver h_{real}	0 m	108	13	0	0	0	0	89.26%
	0.4 m	18	98	4	1	0	0	80.99%
	0.8 m	9	9	103	0	0	0	85.12%
	1.2 m	3	17	4	97	0	0	80.17%
	1.6 m	2	15	2	3	99	0	81.82%
	2 m	5	3	0	3	2	108	89.26%

It could be observed that at all height, the accuracy of height estimation using the proposal was exceeding 80%. Specifically, the highest success rate of height estimation was achieved when the receiver was placed at the upper limit h_{\max} (2 m) and lower limit h_{\min} (0 m) of the candidate height above the ground. The success rate was close to 90%. One possible explanation was that the algorithm may determine the estimated height to be the boundary value when it reached the maximum or minimum candidate height during height estimation. Consequently, the proportion of correctly estimated heights increases.

Another observation is that except when the receiver was placed on the ground, almost all cases of incorrect height estimation result in an estimated height lower than the real height. In other words, these misestimation occurred when the proposed algorithm calculated the distances between the APD and LEDs as farther than the real distances. As the consequence, the height was incorrectly estimated to be lower than the real height.

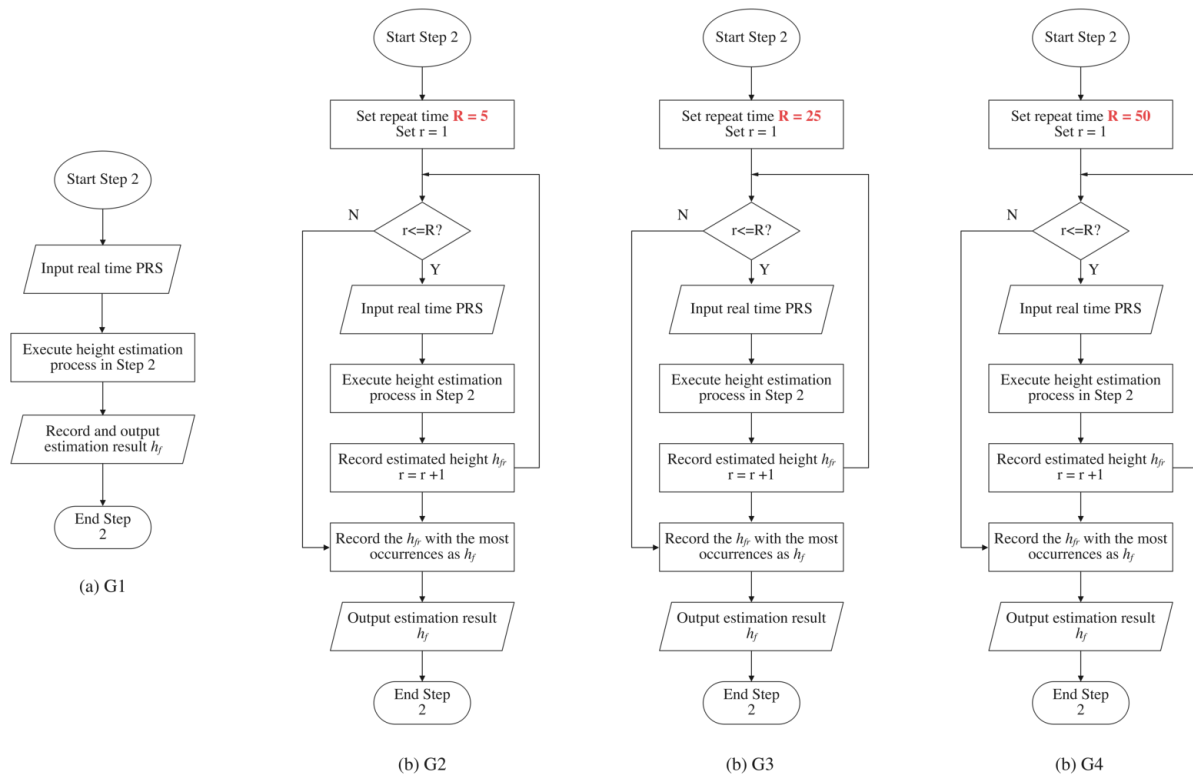


Figure 4-4: Simulation flow for four groups with different repeat time R

To reduce the count of inaccurate height estimations, further simulations were considered. We thought that in a realistic localization scenario, user would naturally consider taking multiple measurements and collecting more light signals to reduce random errors, and performing more computations to obtain accurate estimates. We aim to verify whether such an idea is applicable to our proposed algorithm. Therefore, perform multiple height estimates for each of 121 reference points and use the most frequent estimate as the final height estimate. We conduct four sets of simulation experiments, denoted as G1, G2, G3 and G4, where the height of the receiver is estimated different R times for each reference point. For G1, only one estimate is made, and the estimated value is directly used as the final decision. For G2, G3 and G4, multiple estimates were recorded and the most frequently occurring altitude was determined as the final altitude estimate. The simulation flow for these four groups and the value of R in different group are illustrated Figure 4-4.

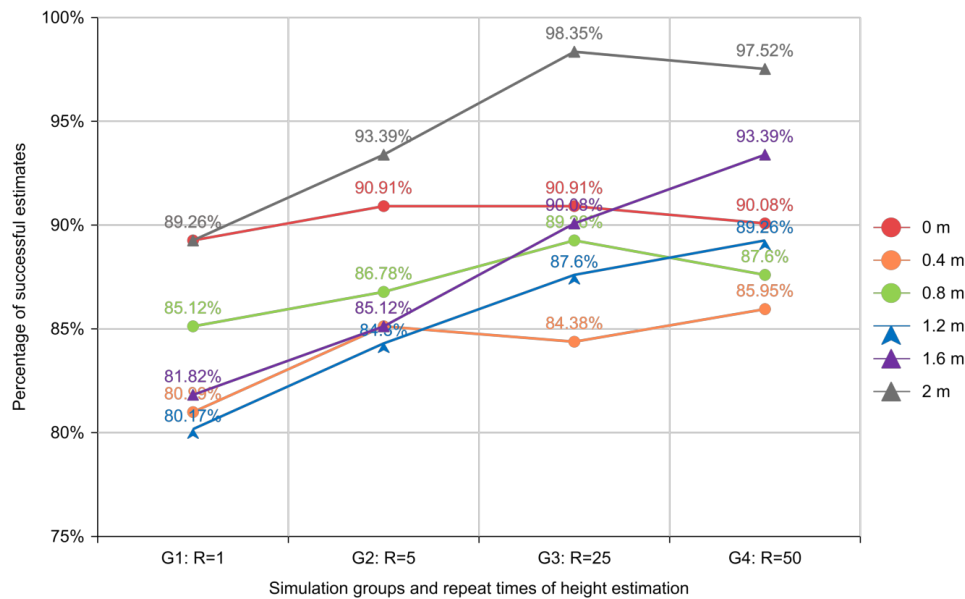


Figure 4-5: Percentage of successful height estimations of different groups with different repeat time R

The line chart Figure 4-5 illustrates the success rate of height estimation for the four groups of simulations (G1 to G4). The horizontal axis represents different groups with different repetition times for height estimation, while the vertical axis indicates the success rate of height estimation, which is the ratio of the number of successful height estimates to the total number of reference points (121) at each height. In other words, the value of vertical axis shows the percentage of the number of height estimates that match the true receiver height for each height, using the proposed algorithm.

In Figure 4-5, it could be indicated that when determining the receiver's height, a single estimation of height ($R=1$) with direct use as the final estimation value in G1 could lead the lowest success rate for estimation in every candidate height. As R increases, the proposal selected the most frequently occurring estimated height from R estimations as the final estimation value (in G2, G3, and G4), the percentage of successful height estimation was higher than G1. In other words, multiple estimations have been demonstrated to be effective in reducing random error.

In detail, as R increased from 1 to 5 (from G1 to G2), the accuracy of the height estimates significantly improved at all heights, demonstrating the efficacy of repeated measurements in reducing random errors. As R continued to increase to 25 in G2, the majority of height estimates continued to improve, indicating that random errors were still being reduced, except at two receiver heights: in 0 m and 0.4 m. At these two heights, the accuracy of height estimates was different from that at other heights and did not vary by more than 1%. In other words, when the receiver is positioned within 0.4 meters from the ground, repeating the measurement 5 times is sufficient to achieve the highest success rate and can accurately estimate the height of receiver.

Increasing the repetition number R during height estimation process in G3 could still increase the success rate of height estimation when the height of the receiver was placed above 0.8 m from the ground. Specifically, when the receiver was placed at a height of 2 m from the ground, the accuracy of height estimation reaches 98%. In other words, if the anticipated location of

the receiver is not close to the ground, repeating the measurement 25 times can achieve the best accuracy in height estimation.

When increasing the repetition times R from 25 to 50 in G4, the improvement in height estimation accuracy was only significant at two heights: 1.2 m and 1.6 m, with an increase of approximately 2 to 3 percentage points. For other heights, excessive repetition times could lead to a decrease in accuracy, such as at 0.8 m and 2 m. One possible reason was that when repeated too much times, the number of incorrect estimations also increased, causing them to have a greater weight in the 50 estimation results, and thus the algorithm adopted these incorrect estimation values as the final height estimation result. In other words, increasing the repetition times R from 25 to 50 times did not significantly improve the success rate of height estimation. Moreover, it might increase the time cost during the repetition process. Therefore, $R=50$ in G4 is not recommended.

To sum up, repeating measurements have been shown to be effective in reducing random error during receiver height estimation. As the repetition number R increased, the success rate of height estimation also increased. When the receiver is positioned within 0.4 m from the ground, repeating the measurement 5 times is sufficient to achieve the highest success rate and accurately estimate the height of the receiver. If the anticipated location of the receiver is not close to the ground, repeating the measurement 25 times can achieve the best accuracy in height estimation. However, increasing the repetition times R from 25 to 50 did not significantly improve the success rate of height estimation and might increase the time cost during the repetition process. Therefore, it is recommended to determine R as 5 or 25 based on the most likely height of the receiver. If there is no idea about the most likely height of receiver, then adopting $R = 25$ can achieve more accurate height estimation.

4.3.3 3D positioning results

4.3.3.1 Overview of the PE

Table 4-3 listed the simulation results of 3D positioning in terms of the average PE, the minimum PE and the maximum PE. In general, the average of the positioning error of all 3D positioning simulations could achieve 2.39 cm, while the minimum and maximum PEs were 8.3187×10^{-2} cm and 27.55 cm, respectively. It could be inferred from the table that when APD was placed on the ground, it had the most stable positioning accuracy, enabling the coordinates of APD to be estimated precisely. It could be inferred that the farther from the LED in indoor 3D positioning, the higher the positioning accuracy. This distance consists of the vertical and horizontal distance of the receiver from the LED. Due to the limited height of the room, the vertical distance would not increase infinitely, but the horizontal distance may increase according to the change of the horizontal and vertical coordinates of the receiver.

Therefore, when designing an indoor 3D positioning algorithm, the designer needs to first determine the maximum signal coverage of each LED. In this range, it is best to place the receiver as low as possible. For example, it can be mounted on the side bottom of a mobile robot for higher precision and more stable positioning performance.

Table 4-3: Simulation results of 3D position

Height of APD (m)	Average PE (m)	Minimum PE (m)	Maximum PE (m)
0	0.0165	8.3187×10^{-4}	0.0751
0.4	0.0168	8.7702×10^{-4}	0.0746
0.8	0.0177	9.3143×10^{-4}	0.0693
1.2	0.2222	0.001	0.0871
1.6	0.0313	0.0011	0.1455
2	0.0474	4.3088×10^{-4}	0.3077

4.3.3.2 Distribution of positioning results

Figure 4-6 illustrates the distribution of reference points of the APD and its positioning results in different heights. Blue dots and green dots represent real positions and estimated positions of APD, respectively, while red dots are the positions of LEDs. If the blue dots and the green dots are close to each other, the positioning results were more accurate, and vice versa. At the same time, the greater the height difference between the green and the blue dots, the less accurate the algorithm estimated the height. Conversely, if the green and the blue dots were distributed on the same height plane, it means that the algorithm could accurately predict the height of the receiver.

Firstly, when the APD was placed on the ground, the distribution of positioning results is displayed in Figure 4-6 (a). It shows that most of the green and blue dots are in the same plane, which means they were accurately estimated heights. There are some green dots above the blue dots, represents they were estimated to be higher than their real position. When the receiver was placed on the height of 0.4 m displayed in Figure 4-6 (b), there are some green dots on the plane with height of 0 m, which means that the height of some points was underestimated. Only two green dots are above the blue dots, indicating that only a very small number of heights were overestimated. This situation could be summarized as the distance between the APD and the LEDs is mostly estimated to be a large value rather than a small value.

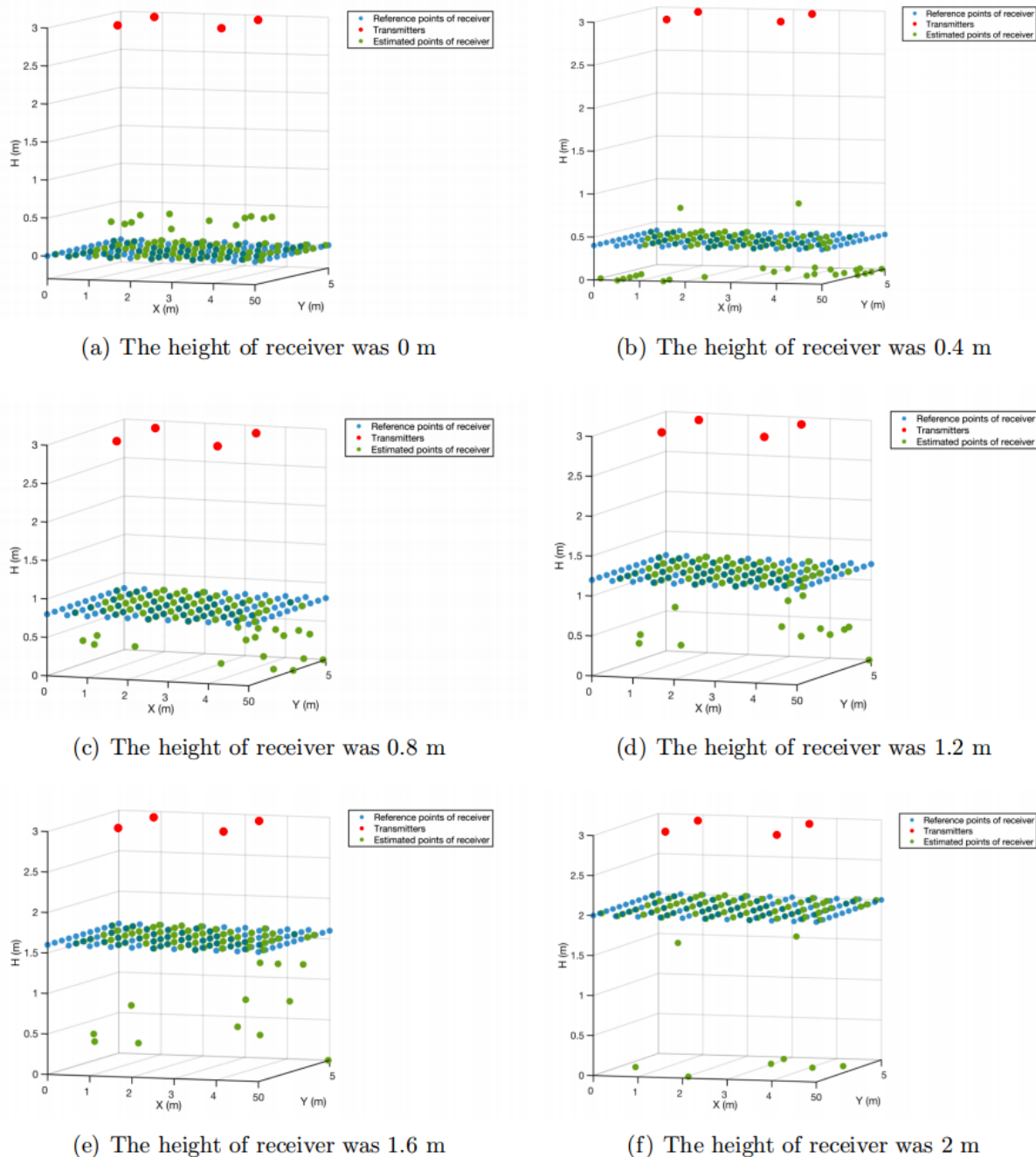


Figure 4-6: Distribution of 3D positioning results when height of receiver was from 0 m to 2 m.

We continued to place the receiver up, then the receiver was on a plane of 0.8 m, 1.2m and 1.6 m, which are shown in Figure 4-6 (c) to (e), it could be observed that there are still some green dots below the plane where the blue points are, but there are no green dots are above the plane anymore. We continue to put the receiver up, and the receiver is on the planes of 0.8 m, 1.2 m, and 1.6 m, as shown in the Figure 4-6 (c) to (e), which can be the plane where the blue point is observed. There are still some green dots below, but no more green dots above the plane. One possible reason is that due to the addition of noise when the optical signal propagates in the OWC, making the estimated distance between the LED and the APD farther than the actual distance, so the estimated coordinates of the APD are also slightly farther away from the LED.

The last case was the simulation performed when the receiver was placed on the plane with a height of 2 m. The positioning results are displayed in Figure 4-6 (f). It is obvious that only a few green dots are located below their true height plane, which also shows that most of the height of reference points could be accurately estimated.

Inspection of the positioning results in different heights indicated that about 80% of the points were estimated to the correct height, and the rest were mostly estimated to be lower than their true heights. These misestimation points were mainly distributed in the corner of the positioning area, which means that height estimations for corner areas still need to be improved as discussed previously. Even so, as long as the APD was placed at a distance of 50 cm from the corner of the positioning area, the height of the APD could be accurately estimated.

On the other hand, estimations of x and y coordinates were precise. It could be easily observed that in horizontal view, green and blue dots are close and most of them already overlap. It turned out that the proposed algorithm continued the advantages of high accuracy in 2D positioning.

To sum up, the overall average PE of 3D positioning algorithm could achieve 2.39 cm while over the 90% of the results could be under 4.59 cm. It is recommended not to place the APD within 1.5 m of LEDs, because this may lead unstable positioning results. Meanwhile, it is also not recommended to be placed too close to the corners of the positioning areas, as the optical signal strength will become weaker, which may cause the misestimation on APD's height. Except for this small part of the special area, the proposal could achieve a high accuracy and high stability positioning performance in 3D scenario.

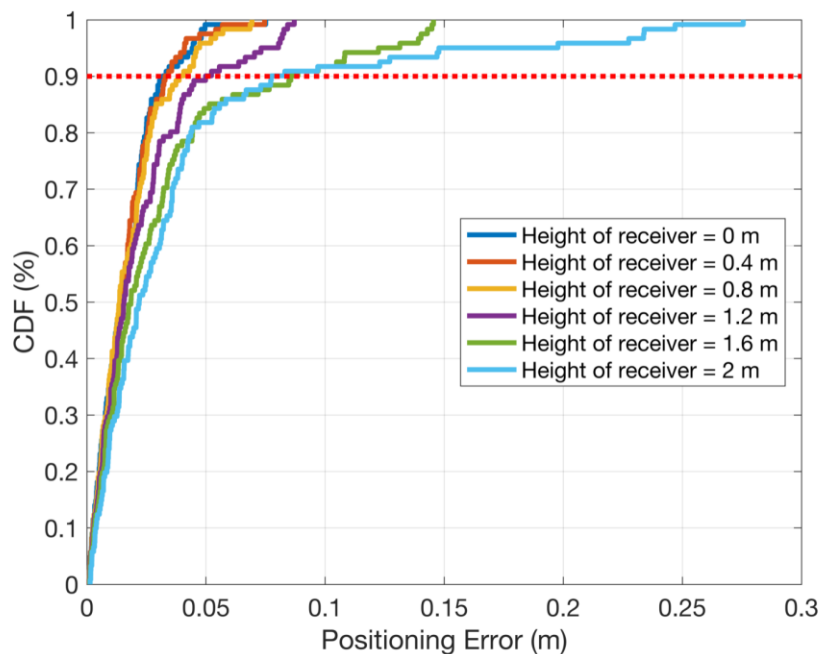


Figure 4-7: CDF of the positioning results in different height of receiver by using the proposal.

4.3.3.3 CDFs of PE in different height of receiver

Figure 4-7 displays the CDF of PEs when the receiver was placed in the height of h_z , $h_z \in H_c = \{0, 0.4, 0.8, 1.2, 1.6, 2\}$, which gives a graphical representation of the cumulative probability

distribution of the PE. The left side of the figure represents the cumulative probability of the PEs being less than or equal to a certain value in x-axis, while the right side represents the cumulative probability of the PEs being greater than or equal to that value in x-axis. To analyze the statistical results of PE, we can first select the range of PE values from the x-axis, and then use the y-axis to determine the cumulative probability of being lower than the PE within this range. It could be observed that the CDF when the receiver was placed on the ground is at the far left of the figure, while the CDF when the receiver was placed on the height of 2 m extends to the right of the figure. We could infer that when the APD was placed in a lower height, the PE curve was closer to the left side of the figure, which means the performance of the positioning accuracy was better.

At the same time, the dark blue, red and yellow lines are very close together, which means their CDFs are similar. What these three colors represent is when the receiver was placed at a height of less than 1 m, thus we could infer that a similar CDF distribution can be obtained as long as the height of the receiver is less than 1 m. Moreover, when APD's height was under 50 cm, 90% of the PEs were under 3.8 cm. On the contrary, when the height of APD was higher than 1 m, larger PEs began to appear. Even so, 90% of all PEs were less than 4.59 cm.

4.3.3.4 Time and space complexity analysis of proposed algorithm

Since steps 2 and 3 include the entire process from receiving the PRS to computing the receiver's estimated coordinates, we want to know how long it takes to perform a positioning process. For step 1, as there is no need to repeat the preparation steps in the positioning process, so the time complexity of this step is not considered here and the focus is on the last two steps. Generally speaking, if the APD is closer to the LED, the less will be lost in the OWC when the optical signal is transmitted, and the more difficult it is to determine the position of the APD based on the loss. Therefore, as the height of the receiver increases, the algorithm theoretically needs more time to estimate the receiver's position since it is closer to the LED.

In order to visually verify the performance of our proposal in terms of algorithmic time complexity, we placed the receiver at different reference points at different heights to run the positioning process, and recorded the time it took to estimate the coordinate once, which are shown in Table 4-4. The positioning process of each point was repeated 5 times, and the average time required for 5 times was taken. The simulations were conducted using a personal computer equipped with a dual-core Intel Core i5 processor with a clock speed of 1.6 GHz and 8 GB of memory. The localization process was performed at room temperature using Matlab R2019b [38] software. The control variable method was used each time the running time was recorded, and only the changed parameters were changed.

Table 4-4: Average time taken to perform a positioning for one reference point in different height of receiver.

Height of APD (m)	0	0.4	0.8	1.2	1.6	2
Average time taken to perform a positioning process (s)	0.0961	0.0985	0.0995	0.0962	0.0956	0.0984

It could be observed from the table that no matter what height the receiver was placed, the average time it took to run a positioning process was less than 0.1 second. Meanwhile, the average positioning time across all simulations is 0.0974 seconds per reference point. This not only means that the proposed algorithm maintains a stable running time even if the receivers

are placed at different heights in space. In other words, the proposed algorithm maintains a stable running time even if the receivers are placed at different heights in space. The space complexity of the proposed algorithm is mainly manifested in the stored environment index classifier group. In Step 1, once the height set H_c and the number of reference points K for each height are determined, the size of the classifier group can be determined. In Steps 2 and 3, no matter how many positioning processes are executed, the classifier group obtained from Step 1 can be used for Step 3 without regenerating the classifier and it is easy to implement with minimum complexity.

5 Conclusions

Preliminary functional tests of a TDoA localization measurement system was presented and found to measure transmission time to 0.33 ps accuracy which is an equivalent to distance of 0.1 mm accuracy.

A novel integration of deep neural network and reinforced learning (DLIRL) based scheme for intelligent beamforming in massive MIMO wireless communication has been investigated in this paper. The DLIRL algorithm learns from its surrounding environment and trains the network as per the requirement of B5G system. Such intelligent focusing of beams enhances the spectral efficiency between downlink integrated access and backhaul and user equipment data transmission and reception. As the DLIRL exhibits a narrow deviation as compared to reinforced learning and deep neural network-based beamforming techniques, the DLIRL may track the user equipment at very high accuracy in B5G networks. Hence, the spectral efficiency with DLIRL-based beamforming is found to be 53.33% and 51.66% more as compared to deep neural network and reinforced learning-based beamforming techniques, respectively.

A 3D environment-aware Visible Light Positioning (VLP) system was introduced. With our novel algorithm, we observed that the estimated height corresponded to the true height of the receiver in over 80% of the cases. An average positioning accuracy of 2.39cm was attained, and more than 90% of the positioning results were found to be within a margin of 4.59cm. In conclusion, our proposed 3D environment-aware VLP algorithm has been demonstrated to effectively estimate not only the height of the receiver but also its position in 3D space, all the while maintaining a high level of positioning accuracy.

References

- [1] Cox, H., Zeskind, R., Owen, M.: Robust adaptive beamforming. *IEEE Trans. Acoust. Speech Signal Process.* 35(10), 1365– 1376 (1987)
- [2] Kela, P., Costa, M., Turkka, J., Koivisto, M., Werner, J., Hakkarainen, A., Valkama, M., Jantti, R., Leppanen, K.: Location based beamforming in 5G ultra-dense networks. In: 2016 IEEE 84th Vehicular Technology Conference (VTC-Fall). Montréal, Canada, pp. 1– 7 (2016)
- [3] Rihan, M., Soliman, T.A., Xu, C., Huang, L., Dessouky, M.I.: Taxonomy and performance evaluation of hybrid beamforming for 5G and beyond systems. *IEEE Access* 8, 74605– 74626 (2020)
- [4] Lin, Z., Lin, M., Wang, J.B., Huang, Y., Zhu, W.P.: Robust secure beamforming for 5G cellular networks coexisting with satellite networks. *IEEE J. Sel. Areas Commun.* 36(4), 932– 945 (2018)
- [5] Jang, J., Chung, M., Hwang, S.C., Lim, Y.G., Yoon, H.J., Oh, T., Min, B.W., Lee, Y., Kim, K.S., Chae, C.B., Kim, D.K.: Smart small cell with hybrid beamforming for 5G: Theoretical feasibility and prototype results. *IEEE Wireless Commun.* 23(6), 124– 131 (2016)
- [6] Zhou, B., Liu, A., Lau, V.: Successive localization and beamforming in 5G mmWave MIMO communication systems. *IEEE Trans. Signal Process.* 67(6), 1620– 1635 (2019)
- [7] Zhu, X., Qi, F., Feng, Y.: Deep-learning-based multiple beamforming for 5 g uav iot networks. *IEEE Network* 34(5), 32– 38 (2020)
- [8] Liu, C., Yuan, W., Wei, Z., Liu, X., Ng, D.W.K.: Location-aware predictive beamforming for UAV communications: A deep learning approach. *IEEE Wireless Commun. Letters* 10(3), 668– 672 (2020)
- [9] Eappen, G., Shankar, T., Nilavalan, R.: Advanced squirrel algorithm-trained neural network for efficient spectrum sensing in cognitive radio-based air traffic control application. *IET Commun.* 15(10), 1326– 1351 (2021)
- [10] Mismar, F.B., Evans, B.L., Alkhateeb, A.: Deep reinforcement learning for 5G networks: Joint beamforming, power control, and interference coordination. *IEEE Trans. Commun.* 68(3), 1581– 1592 (2019)
- [11] Alkhateeb, A., Alex, S., Varkey, P., Li, Y., Qu, Q., Tujkovic, D.: Deep learning coordinated beamforming for highly-mobile millimeter wave systems. *IEEE Access* 6, 37328– 37348 (2018)
- [12] Huang, H., Peng, Y., Yang, J., Xia, W., Gui, G.: Fast beamforming design via deep learning. *IEEE Trans. Veh. Technol.* 69(1), 1065– 1069 (2019)
- [13] Paszke, A., Chaurasia, A., Kim, S., Culurciello, E.: Enet: A deep neural network architecture for real-time semantic segmentation. *arXiv preprint arXiv:1606.02147* (2016)
- [14] El Ayach, O., Rajagopal, S., Abu-Surra, S., Pi, Z., Heath, R.W.: Spatially sparse precoding in millimeter wave MIMO systems. *IEEE Trans. Wireless Commun.* 13(3), 1499– 1513 (2014)

- [15] Va, V., Choi, J., Heath, R.W.: The impact of beamwidth on temporal channel variation in vehicular channels and its implications. *IEEE Trans. Veh. Technol.* 66(6), 5014– 5029 (2016)
- [16] Li, X., Alkhateeb, A.: Deep learning for direct hybrid precoding in millimeter wave massive MIMO systems. In: 2019 53rd Asilomar Conference on Signals, Systems, and Computers. Pacific Grove, CA, pp. 800– 805 (2019)
- [17] Eappen, G., Shankar, T., Nilavalan, R.: Advanced squirrel algorithm-trained neural network for efficient spectrum sensing in cognitive radio-based air traffic control application. *IET Commun.* 15(10), 1326– 1351 (2021)
- [18] Eappen, G., Shankar, T., Nilavalan, R.: Cooperative relay spectrum sensing for cognitive radio network: Mutated MWOA-SNN approach. *Appl. Soft Comput.* 114, 108072 (2022)
- [19] Vizziello, A., Savazzi, P., Chowdhury, K.R.: A Kalman based hybrid precoding for multi-user millimeter wave MIMO systems. *IEEE Access* 6, 55712– 55722 (2018)
- [20] Va, V., Vikalo, H., Heath, R.W.: Beam tracking for mobile millimeter wave communication systems. In: 2016 IEEE Global Conference on Signal and Information Processing (GlobalSIP). Washington, DC, pp. 743– 747 (2016)
- [21] Shaham, S., Ding, M., Kokshoorn, M., Lin, Z., Dang, S., Abbas, R.: Fast channel estimation and beam tracking for millimeter wave vehicular communications. *IEEE Access* 7, 141104– 141118
- [22] Kishor, A., Chakraborty, C., Jeberson, W.: Reinforcement learning for medical information processing over heterogeneous networks. *Multimedia Tools and Applications* 80(16), 23983– 24004 (2021)
- [23] Kishor, A., Chakraborty, C., Jeberson, W.: A novel fog computing approach for minimization of latency in healthcare using machine learning. *International Journal of Interactive Multimedia and Artificial Intelligence* 6, 7– 17 (2021)
- [24] Kishor, A., Chakraborty, C.: Task offloading in fog computing for using smart ant colony optimization. *Wireless Personal Communications* 1– 22 (2021)
- [25] Liu, Y., Wang, X., Boudreau, G., Sediq, A.B., Abou-zeid, H.: Deep learning based hotspot prediction and beam management for adaptive virtual small cell in 5G networks. *IEEE Trans. Emerging Top. Comput. Intell.* 4(1), 83– 94 (2020)
- [26] ElHalawany, B.M., Hashima, S., Hatano, K., Wu, K., Mohamed, E.M.: Leveraging machine learning for millimeter wave beamforming in beyond 5G networks. *IEEE Syst. J.* 16(2), 1739– 1750 (2022)
- [27] Santos, G.L., Endo, P.T., Sadok, D., Kelner, J.: When 5G meets deep learning: A systematic review. *Algorithms* 13(9), 208 (2020)
- [28] Ebrahiem, K.M., Soliman, H.Y., Abuelenin, S.M., El-Badawy, H.M.: A deep learning approach for channel estimation in 5G wireless communications. In: 2021 38th National Radio Science Conference (NRSC). Mansoura, Egypt, pp. 117– 125 (2021)
- [29] Lavdas, S., Gkonis, P.K., Zinonos, Z., Trakadas, P., Sarakis, L.: An adaptive hybrid beamforming approach for 5G-MIMO mmWave wireless cellular networks. *IEEE Access* 9, 127767– 127778 (2021)

- [30] Mohamed, K.S., Alias, M.Y., Roslee, M., Raji, Y.M.: Towards green communication in 5G systems: Survey on beamforming concept. *IET Commun.* 15(1), 142– 154 (2021)
- [31] Zhou, Y., Chen, J., Zhang, M., Li, D., Gao, Y.: Applications of machine learning for 5G advanced wireless systems. In: 2021 International Wireless Communications and Mobile Computing (IWCMC). pp. 1700– 1704. IEEE, Piscataway, NJ (2021)
- [32] Silva, D.H., Ribeiro, D.A., Ramírez, M.A., Rosa, R.L., Chaudhary, S., Rodríguez, D.Z.: Selection of beamforming in 5G MIMO scenarios using machine learning approach. In: 2022 19th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON). Thailand, pp. 1– 4 (2022)
- [33] Yu, P., Zhou, F., Zhang, X., Qiu, X., Kadoch, M., Cheriet, M.: Deep learning-based resource allocation for 5G broadband TV service. *IEEE Trans. Broadcast.* 66(4), 800– 813 (2022)
- [34] Shen, L.H., Chang, T.W., Feng, K.T., Huang, P.T.: Design and implementation for deep learning based adjustable beamforming training for millimeter wave communication systems. *IEEE Trans. Veh. Technol.* 70(3), 2413– 2427 (2021)
- [35] Yang, H., Jee, J., Kwon, G., Park, H.: Deep transfer learning-based adaptive beamforming for realistic communication channels. In: 2020 International Conference on Information and Communication Technology Convergence (ICTC). Jeju Island, Korea, pp. 1373– 1376 (2020)
- [36] Geoffrey Eappen, John Cosmas, Shankar T, Rajesh A, Nilavalan R, and Joji Thomas “Deep Learning Integrated Reinforcement Learning for Adaptive Beamforming in B5G Networks” *IET Communications Journal*, 28 September 2022 <https://doi.org/10.1049/cmu2.12501>
- [37] Weeping Guan, Linyi Huang and et al., “Robust Robotic Localisation using Visible light Positioning and Inertial Fusion”. *IEEE Sensors Journal*, 2021.
- [38] Emma Afroza an Pallab K. Choudhury. “An efficient three-dimensional indoor visible light positioning algorithm using multiple transmitters and single receivers”. *Results in Optics*, 2021.