Iraqi Journal for Computer Science and Mathematics

Volume 6 | Issue 1 Article 7

2025

Deep Learning-Based Beamforming Optimization for Reconfigurable Intelligent Surface-Assisted Wireless Communication Systems

Mohammed Firas Jassim
Electrical Engineering Technical College, Middle Technical University, Baghdad, Iraq, bbc0086@mtu.edu.iq

Alhamzah Taher Mohammed
Electrical Engineering Technical College, Middle Technical University, Baghdad, Iraq, alhamza_tm@yahoo.com

Osamah Abdullah
Office of the Undersecretary for Scientific Research Affairs, Ministry of Higher Education and Scientific Research, Baghdad, Iraq, osamah.abdullah@wmich.edu

Recommended Citation

Jassim, Mohammed Firas; Mohammed, Alhamzah Taher; and Abdullah, Osamah (2025) "Deep Learning-Based Beamforming Optimization for Reconfigurable Intelligent Surface-Assisted Wireless Communication Systems," *Iraqi Journal for Computer Science and Mathematics*: Vol. 6: Iss. 1, Article 7.
DOI: https://doi.org/10.52866/2788-7421.1233
Available at: https://ijcsm.researchcommons.org/ijcsm/vol6/iss1/7

IRAQI JOURNAL FOR COMPUTER SCIENCE AND MATHEMATICS 2025;6:66–88

ORIGINAL STUDY

Deep Learning-Based Beamforming Optimization for Reconfigurable Intelligent Surface-Assisted Wireless Communication Systems

Mohammed Firas Jassim [a,*], Alhamzah Taher Mohammed [a], Osamah Abdullah [b]

[a] Electrical Engineering Technical College, Middle Technical University, Baghdad, Iraq
[b] Office of the Undersecretary for Scientific Research Affairs, Ministry of Higher Education and Scientific Research, Baghdad, Iraq

ABSTRACT

This research investigates how deep learning might be used to optimize beamforming in wireless communication systems that are helped by Reconfigurable Intelligent Surfaces (RIS). Our goal is to increase the possible data rates by dynamically forecasting the best phase shifts for RIS elements by utilizing Convolutional Neural Networks (CNN) and hybrid CNN-Long Short-Term Memory (CNN-LSTM) models. We assess the performance of these deep learning models against conventional genie-aided techniques by simulating real-world wireless settings using the DeepMIMO dataset. The findings demonstrate that beamforming based on deep learning can reach near-optimal performance, greatly lowering the overhead associated with channel estimation while improving communication efficiency. In order to support the development of next-generation wireless networks like 5G and 6G, this study shows how deep learning approaches can be used to increase the effectiveness of RIS-assisted systems.

Keywords: Reconfigurable intelligent surface (RIS), Beamforming, Deep learning, CNN, LSTM, DeepMIMO, Wireless communication, 5G, 6G

1. Introduction

MULTIPLE Input Multiple Output (MIMO) system has become a focal point of research due to its potential to significantly improve wireless communication and support the rapidly growing Internet of Things (IoT), where billions of devices are anticipated to connect and communicate seamlessly [1–10]. A key innovation driving these advancements is the incorporation of Reconfigurable Intelligent Surface (RIS) technology, which builds upon traditional MIMO frameworks to enhance throughput, broaden cell coverage, and reduce power consumption by leveraging high-gain antenna arrays. RIS operates by manipulating electromagnetic waves, al- lowing for concentrated energy in three dimensions, which supports applications such as wireless power transfer, high- precision sensing, and the transmission of large data volumes [2–5].

A typical RIS system consists of a planar array of numerous reflective elements, each functioning as a phase shifter to control the direction of reflected electromagnetic signals. This mechanism not only alters signal propagation but also enhances communication quality [6, 7]. The RIS reflection matrix optimization is key to improve the quality of communication and increase the amount of data that can be sent. There are two principal methods for this optimization: the first approach leverages exhaustive training at the transmitter/receiver to infer the RIS-assisted channel, which is computation-intensive because of intricate reflections among numbers of reflective elements [8, 9]. The second way of selecting the reflection matrix is by using predetermined quantized codebooks for

Received 1 October 2024; revised 9 January 2025; accepted 15 January 2025.
Available online 12 February 2025

* Corresponding author
E-mail addresses: bbc0086@mtu.edu.iq (M. F. Jassim), alhamza_tm@yahoo.com (A. T. Mohammed), osamah.abdullah@wmich.edu (O. Abdullah).

https://doi.org/10.52866/2788-7421.1233

channel selection without explicit channel estimation and can lead to increasing system complexity and dedicated hardware implementations but at the cost of possibly deteriorated system performance [10].

Based on the recent research, using an on-off scheme for channel estimation can reduce training overhead and a three-step cascaded channel estimation process provides higher efficiency. Standard methods however do not make the best use of prior knowledge within shared channels, which parameters could be investigated and optimized further [11]. Prior studies on Reconfigurable Intelligent Surface (RIS) interactions have been predominantly dedicated to solving problems of channel estimation and beamforming design [12–16]. In this regard, the techniques merged with deep learning (DL) have been suggested in order to reduce training burden of channel and beam [12]. Based on these, an RIS configuration optimization approach was proposed in [13] that has triggered a number of stems regarding this method and its benefits [17]. Well, in terms of channel estimation and beamforming issues coordinated with RIS, supervised DL is proposed to learn how to map pilot signals more efficiently [14], and unsupervised DL are used for those applications improvements. Further, in [15], the minimum variance unbiased estimator has also been proposed to improve the channel estimation accuracy. Besides, deep learning has been used to alternatively determine the RIS reflection matrix so as to maximize the recognition of reflection coefficients that improve system performance [16, 18–27].

1.1. Contribution

This work enhances the research in Reconfigurable Intelligent Surface (RIS) based communication systems with a focus on utilizing Deep Learning (DL) and Long Short-Term Memory (LSTM) networks for optimizing beamforming efficiency. Below are the key contributions of this code and methodology:

- CNN and CNN-LSTM Hybrid for Beamforming Prediction: The integration of LSTM layers and CNN-LSTM hybrid architecture enhances beamforming performance by modeling temporal and spatial dependencies, leading to improved accuracy and efficiency compared to traditional methods.
- Efficient Achievable Rate and Reduced Channel Estimation Overhead: The DL model approximates optimal beamforming performance, offering a computationally efficient alternative with reduced training overhead, particularly in large-scale applications like (RIS).
- Realistic Dataset Evaluation and Open-Source Contribution: The use of the DeepMIMO dataset ensures robust performance evaluation, and the open-source licensing allows other researchers to build on this work for RIS beamforming optimization.

2. Liteature review

One of the key contributions was the creation of a (DL) framework for channel state information (CSI) in (RISs) by Elibir et al. [27]. Pilot signals were received by the user equipment (UE), and the DL model used in their methodology used these signals as input. Artificial training data was generated by forming input-output pairings across several channel realizations by turning RIS elements ON and OFF. The deep neural network (DNN) produced vectorized channel matrices as its output, while direct and cascaded channels supplied the input data. It's interesting to note that the model did not require retraining when the user's location moved by four degrees. The DL architecture was composed of two nine-layer (CNNs) with an optimizer for stochastic gradient descent (SGD) of 128 samples, dropout, and mini-batches.

A two-phase system was proposed by Taha et al. in [28]. The learning phase, also known as Phase 1, involved RIS doing a thorough search for data in order to train the DL model. Selecting the optimal beamforming vector to optimize the achievable rate was necessary. The concept is to produce pairs of the output vector and the input channel vectors, which the model must correctly map before the trajectory is completed. The DL model entered the prediction phase after estimating the beamforming vector from the estimated channel. The architecture was trained on the DeepMIMO dataset using a neural network (NN) with many layers that was built using rectified linear unit (ReLU) activation.

The authors of a different study [29] developed DeepRIS, a DL-based detector that estimates the channel and phase angles at wireless receivers using the received signal. This model was trained offline using random bit sequences, phase alterations, and synthetic channel realizations. Its output provided an estimate of the transmitted symbol and CSI. The architecture employed the Adam optimizer in conjunction with an artificial neural network (ANN) that had a tanh activation function and a customizable number of fully connected layers in order to preserve negative weights.

A deep-learning based denoise model for predicting channel state information (CSI) was proposed in [30] and it is named deep-denoising neural network

Table 1. Related work summarization.

Reference	Database Used	ML Algorithm	Architecture and Methods	Available Source Code?
[27]	Synthetic	DL	Two 9-layer CNNs, dropout, SGD optimizer, minibatch training	Yes, MATLAB R2018b
[28]	DeepMIMO	DL	Flexible-layer adaptive neural network with ReLU activation function	Yes, MATLAB R2018b
[29]	Synthetic	DL	Tanh activation, Adam optimizer, and adaptive neural network with a changeable number of layers	No
[30]	Synthetic	DL	CNN with 64 filters (3 × 3 × 64), ReLU activation, 15 convolutional layers, and Adam optimizer	Yes, MATLAB R2018b and Python
[31]	Synthetic	DL	CNN designs using EDSR and MDSR, and the ReLU activation function	Yes, Python
[32]	Synthetic	DL	Batch normalization, ReLU activation, 3 × 3 filters with conv2D layers, and FFDNet CNN	Yes, Python
[33]	DeepMIMO	DL	CNN with 32 filters, ReLU activation	Yes
[34]	Synthetic	DL	NN architecture with ReLU activation, NMSE loss function	Yes, Python
[35]	Synthetic	DL	ANN with linear layers, sigmoid activation, Adam optimizer	Yes, Python

(DDNN), which aims to act in millimeter wave (mmWave) reconfigurable intelligent surface (RIS) systems. The idea is to exploit the sparsity of cascaded channels and compressive sensing (CS) as not all the components were used in training, then end up using orthogonal matching pursuit (OMP) for reconstructing a full channel matrix from partial data. When we proposed the OMP-DL framework, we further employed an over complete dictionary and more optimized multi-carrier pilot signals to enhance system performance. The model architecture consisted of a 15-layer CNN with ReLU activation and Adam optimization, where each layer has 64 filters of size 3× 3 × 64.

Recently, [31] has introduced a deep learning model with the use of CNNs to accelerate CSI calculation in RIS-assisted networks. For single-ray settings, an extended deep super-resolution neural network (ESDR) model was proposed to accurately estimate the CSI. In the multi-scale deep super-resolution neural network (MSDR), parameters could be adaptively adjusted for multiple scales and sparse, low-resolution devices [11]. Both models use ReLU activations for forecasting the channel matrix.

The Fast and Flexible Denoising Network (FFDNet), a CNN-based approach, was released in [32]. In this model, synthetic channel realizations with independent real and imaginary components were assumed. In the residual block, FFDNet fed noise variance information using Adam optimization, 2D convolutional layers, and ReLU activation functions.

The distributed machine learning (DML) system was first presented in [33] and employed CNN to manage downlink CSI estimation. This approach enhanced the accuracy of estimation by obtaining features from the channel under different conditions. The system operated even when users moved between cells because the base station (BS) generated a global model that users shared and cooperatively trained using local datasets. Max-pooling, batch normalization, ReLU activation, Adam optimization, and 32 filters with a 3 × 3 kernel were all included in the design. The model was trained using the outdoor scenario of the DeepMIMO dataset.

He et al. [34] The techniques to arrive at a deep unfolding solution exploited the cascaded channel matrix as it is rank-deficient thereby minimizing training overhead and improving inceive CSI estimation. It was constructed with linear layers, had synthetic channel realizations as input, used Adam optimization and ReLU activation in all but the last layer.

The performance of the RIS-reflective network was first tuned using multi-user (MU) downlink precoding, channel state information (CSI) estimation, followed by sigmoid activation and Adam optimization [35]. It appeared to be effective particularly when there was a line of sight (LOS) between user devices and the base station (BS). Therefore, the BS utilized low pilot signals along with low feedback overhead over a downlink network for acquiring the CSI.

3. Methodology

3.1. Model system

In the system under examination, a base station (BS) interacts with many users via a wireless communication network supported by a Reconfigurable Intelligent Surface (RIS) [36]. Each of the many passive reflecting components that make up the RIS has the ability to change the incident signal's phase in order to improve communication between the users and the BS. This system model is based on a downlink MU scenario in which the users are single-antenna devices and the base station (BS) has multiple antennas.

The RIS is crucial to enhancing signal quality because the direct communication path between the BS and the users might occasionally be impeded by obstructions or distance [37]. The system may intelligently guide messages towards users by modifying the phase shifts of the RIS components, hence enhancing overall communication efficiency.

The received signal at the k^{th} subcarrier can be represented as the sum of the direct signal from the BS and the reflected signal via the RIS. The received signal $\mathbf{y_k}$ at subcarrier k is expressed as [28]

$$y_k = h_{R,k}^T \Psi_k h_{T,k} s_k + h_{TR,k} s_k + n_k, \tag{1}$$

Where $h_{R,k}^T$ denotes downlink channel matrix, Ψ_k implies interaction matrix of RIS and $h_{T,k}$ means the uplink channel. Here the s_k is refers to transmitted signal vector and (n_k) represents receive noise.

The main objective of the system is to maximize the transmit signal power at the users by optimizing the RIS phase shifts. This optimization seeks to maximize the achievable rate of the system. The problem can be formulated as follows [28]:

$$R^\star = \max_{\psi \in \mathcal{P}} \frac{1}{K} \sum_{k=1}^{K} \log_2 \left(1 + \mathrm{SNR} \left| \left(\boldsymbol{h}_{T,k} \odot \boldsymbol{h}_{R,k} \right)^T \psi \right|^2 \right) \tag{2}$$

In the system model, the key assumption is that the RIS operates in a quasi-static environment, meaning the channel coherence time is sufficiently long for accurate estimation and adjustment of the RIS phase shifts. Additionally, the model assumes perfect channel state information (CSI) at the BS and the RIS, which enables precise control over the RIS phase shifts to maximize signal quality at the users [38].

3.2. Channel model

The wireless communication channel between the users and the base station (BS), both directly and through the Reconfigurable Intelligent Surface (RIS), is taken into consideration in this system [39]. The RIS introduces phase changes to align the reflected signals with the direct signals, enhancing the received signal along the many pathways that make up the total channel [40]. The relationship between the sent signal, the RIS, and the received signal at the user is modeled in this section.

The communication model assumes a narrowband multipath channel with L significant propagation routes. Every path has a unique time delay, complicated gain, and arrival and departure angles [41]. Let M be the number of RIS reflecting elements; let T be the transmission time; and let ρ be the signal power. The following equation describes the channel between the transmitter and the k^{th} the channel vector at subcarrier, known as $\mathbf{h_{T,k}}$ [28]:

$$h_{T,k} = \sqrt{\frac{M}{\rho T}} \sum_{d=0}^{D-1} \sum_{l}^{L} \alpha_l \left(\theta_l, \phi_l\right) p\left(dT_s - \tau_l\right) e^{-j2\pi kd} \tag{3}$$

The complex gain linked to the l^{th} path is denoted by αl in this equation, and the array response vector corresponding to the azimuth and elevation angles θ_l and ϕ_l is represented by $\alpha_l(\theta_l, \phi_l)$. The pulse-shaping function, $p(dT_s - \tau_l)$, takes into account the delay τ_l that the l^{th} route experiences, and e^{-jkd} takes into account the phase shift that the k-th subcarrier of the total K subcarriers introduces.

The RIS works as a passive beamformer, optimizes the phase of the incoming signal at each reflecting element to increase power of the receiving signal. The global signal perceived by the user originates from the direct signal broadcasted from the base station and that reflected by the RIS. This also opens up the possibility to achieve higher data rates and improved communication quality as the system can intelligently combine signals of multiple paths while setting the phase shifts at RIS [28].

For the analysis and optimization of RIS-assisted communication systems, the channel model is very important. Effective beamforming algorithm design and non-ideal system capacity evaluation depend heavily on its ability to represent various routes, phase shifts, and signal delays. In Fig. 1, a communication system is constructed with a Reconfigurable Intelligent Surface (RIS).

As seen in Fig. 1, RIS is made up of several reflecting parts that function as the transmitter's multiple receivers' means of communication (similar to the passive or intelligent surfaces). Fig. 1 illustrates this, showing how the RIS blocks the direct line path between the transmitter and receiver before making up for it. In order to ensure that the signal can get through obstructions like the tree, which is depicted as a blockage, the RIS interacts with the broadcast signal and reflects it towards the receiver. Phase shifters, which modify the incoming signal's phase to enhance signal alignment at the receiver, are a feature of the RIS elements. It is believed that the RIS architecture

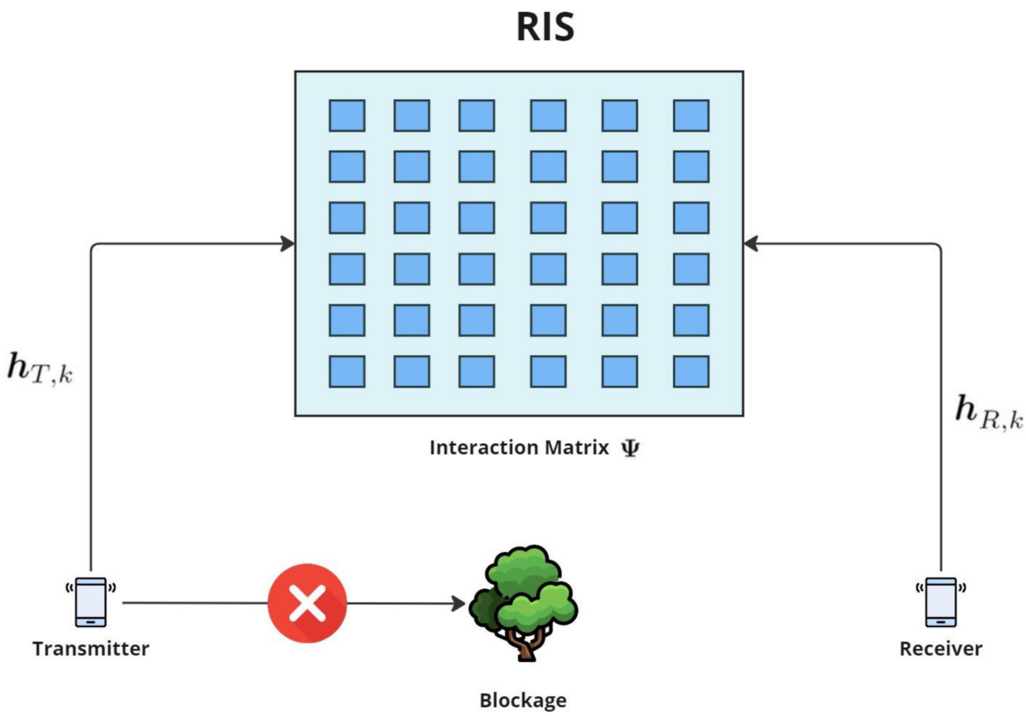

Fig. 1. The system model of an RIS-assisted transceiver system.

includes these phase shifters. The interaction between the RIS elements and the incident signal is modeled using an interaction vector denoted as $[\psi_m] = e^{j\phi m}$ where each element ϕ_m represents the phase shift introduced by the corresponding RIS element.

A predetermined code-book of interaction vectors P is used to choose the interaction vector. The codebook provides different preset phase shift patterns that the RIS can employ to adjust the signal's reflection. As mentioned in [12], the underlying premise is that a small number of active reflecting elements are dispersed at random throughout the passive ones on the RIS. While the passive elements reflect the signal with a fixed phase shift, the active elements can alter their phase shifts dynamically.

The channel between the RIS and the receiver, represented as $h_{R,k}$ and the sampled channel vector between the transmitter and the RIS active elements, indicated as $h_{T,k}$ are represented as $h_{T,k}$ $c^{\mathrm{M}\times 1}$ and $h_{R,k}$ $c^{\mathrm{M}\times 1}$, respectively. The channel vectors can be written as follows [28]:

$$\mathbf{h_{T,k}} = \mathbf{G_{RIS}h_{T,k}} \tag{4}$$

$$\mathbf{h_{R,k}} = \mathbf{G_{RIS}h_{R,k}} \tag{5}$$

where GRIS is an M selection matrix that selects the active RIS elements from the total set of RIS elements. This selection matrix determines which RIS elements participate in the reflection process and is tuned based on the configuration of the RIS elements.

Therefore, the overall RIS channel vector, h_s, can be written as the product of the two channel vectors, as follows:

Here, represents the element-wise product, combining the channels between the transmitter and RIS with the channels between the RIS and the receiver. This model captures how the RIS reflects the signal and how the active elements contribute to enhancing the communication link by compensating for the blocked direct path [28].

3.3. Limitations and impact of the deepmimo dataset on model generalizability

The DeepMIMO dataset, which plays a central role in our simulations, targets realistic urban wireless communication scenarios that are critical to the creation of modern technologies like massive MIMO and Reconfigurable Intelligent Surfaces (RIS). The dataset has been created employing high resolution settings of Wireless InSite by Remcom and is corroborated with real yield measures and consists of higher frequencies such as 3.4 GHz, 3.5 GHz, and even higher millimeter wave bands [42–44]. To have over one-million potential users, it realistically mimics urban areas regarding interactions such as path loss, shadowing, or multipath reflections.

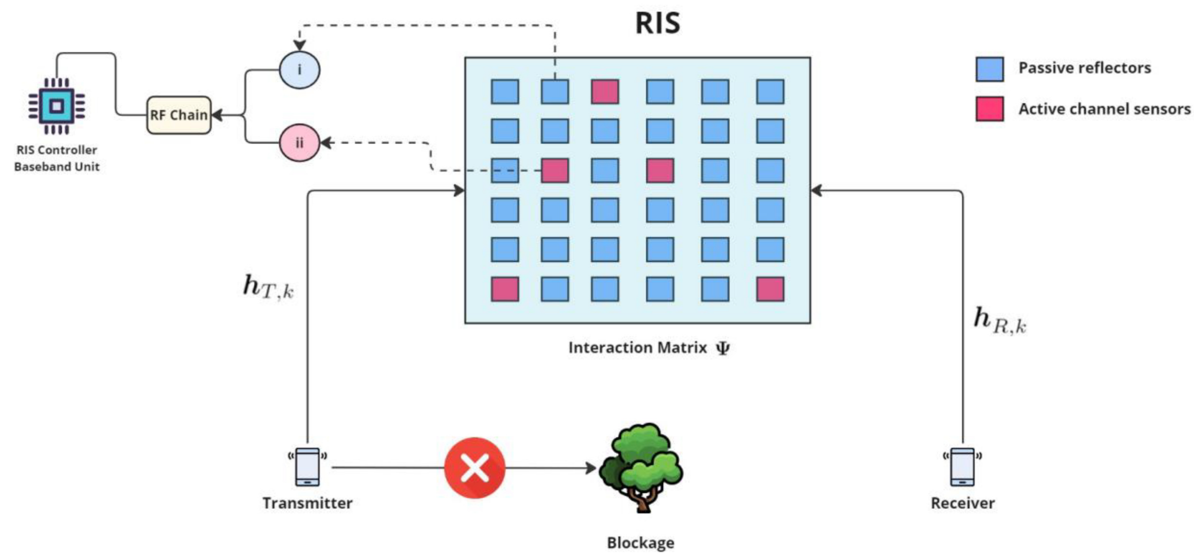

Fig. 2. The proposed RIS architecture with $\bar{M}$ active channel sensors for channel estimation and phase shifting, alongside passive reflectors that apply fixed phase shifts without baseband connection.

In general, DeepMIMO dataset appears to be quite exhaustive, although some shortcoming can still be observed because of its static design and mostly urban scenes. This can however limit the generalizability of the dataset to other types of environments, say those in rural and suburban areas. Also, it lacks dynamic changes like a moving obstacle in the field of view, or changes in atmospheric conditions, which maybe undesirable when it comes to the application of the models trained on this dataset under real-world variability.

For these reasons, and in an effort to reduce bias and increase generalizability of the findings of this study, efforts towards validation of the tool have to be extended across more diverse and dynamic settings. Ensuring more environmental conditions are included in the datasets to make the models from DeepMIMO a more comprehensive set, or including real-time data, also has the potential to drastically enhance the stability and transferability of the models. Some of these steps would assist in guaranteeing that the optimized solutions emanating from such simulations operate optimally not with simulated conditions, but in real implementation contexts.

3.4. Reconfigurable intelligent surface (RIS) architecture

The Reconfigurable Intelligent Surface (RIS) architecture presented here in Fig. 2 features either active or passive designs to enable efficient wireless transmission. Let the RIS consist of total M reflecting elements of which there is a subset of M active channel sensors. These active sensors are placed randomly over the RIS and play an important role in both signal scattering and channel calibration. They operate in two modes. In the channel sense mode, these are connected to the baseband unit and can coordinate for actively estimating the wireless channel between both receiver and RIS and the transmitter and RIS. This channel information is sent to the RIS controller that then adjusts the position of the RIS elements in order to improve the signal reflection towards the receiver. In the second mode of operation, the active sensors mimic passive reflectors, adding a phase shift determined in relation to the incident signals to the communication link without a baseband unity connection [28].

The passive elements represented by the blue reflectors cannot detect channel and supply only phase shift to the incident signals. Although they cannot change their phase shifts based on the channel dynamics, the multiplicity of their numbers will create a collective effort; a big reflecting surface that constantly reshapes to ensure that all the signal is redirected to the intended receiver.

The interaction matrix, Ψ, describes the overall phase shifts provided by both being passive and active phases of the RIS. The baseband controller controls this matrix, which in turn linked with the current active sensors to permit the RIS adjust its reflective channel. This type of architecture combines the relative advantages of both active and passive components guaranteeing a highly dynamic and effective means of improving both the channel estimation and signal reflection policy and still keep the cost constraint in mind [28].

In conclusion, as illustrated in Fig. 2, active sensors enable the RIS the offer fundamental features for real time low latency channel estimation and signal reflection. These sensors allow the system to adjust the reflectivity of the surface on the fly, which will improve the system's performance in facing the unpredictable wireless channels while at the same time reduce the training time needed for the traditional system. The application of active sensors designed as both senders and receivers allows phase shifts of reflected signals to be controlled directly, making their coherent addition at the receiver possible in general. Further, precise enhancements related to the active elements' characteristics allow the adaptation to the channel conditions without constant receiver-side feedback, which enhances system efficiency, decreases feedback latency, and enables it to prevent feedback overheads, rendering this solution suitable for dynamic and real-time wireless networks.

3.4.1. Dual functionality of active sensors in RIS architecture

The active sensors within the RIS architecture possess dual functionality, operating in two distinct modes: the sensing mode and the reflection mode. These modes are important for controlling the behavior of the RIS to response to changes in environment conditions and, more importantly, communication needs, which are evidently the key-performance-indicator elements of a RIS in terms of latency and system complexity.

Sensing Mode: In this mode, the active sensors are interfaced to the baseband unit to facilitate instantaneous CSI collection. The sensors acquire information on the communication channel between the RIS and the transmitter as well as the receiver. This information is very important when in the process of updating the configuration of the RIS so that the signal reflection paths can be changed in a dynamic manner.

Reflection Mode: Once the values of the optimal settings based on CSI obtained during the sensing mode are calculated, active sensors enter the reflection mode. In this mode they work in a manner similar to the passive elements which are used for pointing the beam to the receiver in correct phase. However, unlike the passive elements, these phase shifts can then be adjusted in response to current conditions of the channels.

Mode Switching Mechanism: a regulator that compares the probable need for channel adaptation in real time with the pursuit of the greatest reflection efficacy controls the shift between sensing and reflection modes. This algorithm is special because it does not cause switching delays, which contributes to the systems low latency. The mode switching is initiated by either high mobility of the receiver or variations in the interference levels, which makes a new channel estimation necessary.

In the RIS architecture, active sensors switch dynamically between sensing and reflection modes with high speed, maximizing system latency with this fast switch. But without the algorithms inside the RIS controller properly managing these state changes, there are delays associated with too frequent switching. Moreover, the implementation of dual-mode functionality adds to the complexity of the system, requiring advanced algorithms for autonomous mode switching as well as hardware capable of rapid reconfiguration. These improvements help support the paper's claim of a novel implementation of active sensors and their valuable contribution to the system's dynamic, efficient, and effective operations.

3.5. Convolution neural network

This subsection presents the implementation of a Convolutional Neural Network (CNN) architecture for beamforming optimization in a Reconfigurable Intelligent Surface (RIS)-enabled wireless communication system.The CNN- based DL model is used to predict achievable data rates for users by processing channel state information (CSI) and determining optimal phase shifts at the RIS elements. The system model begins by gathering channel data from a simulated wireless environment using the DeepMIMO dataset. This dataset provides detailed channel matrices that describe the interaction between the transmitter, RIS, and receiver, including the signal's direct and reflected paths.

The channel matrices are processed through the CNN architecture, which begins with an input layer that receives the reshaped input data. The input data consists of real and imaginary parts of the channel response from the RIS to the users, organized into an image-like format for efficient processing. The CNN applies multiple convolutional layers to the input data from which spatial features are extracted. The filters in the first layer scan the input to extract elementary features; batch normalization is only applied for improving training stability and efficiency. A non-linearity introduced by the ReLU activation function allows to learn more complex patterns. After that, a max-pooling layer in order to zoom in on distinctive features and reduce computational complexity reduces spatial dimensions of the feature maps. As the data passes through deeper layers of the CNN, additional convolutional layers are used to learn more abstract and high-level features, with increasing filter sizes to capture finer details of the channel characteristics. Batch normalization, ReLU

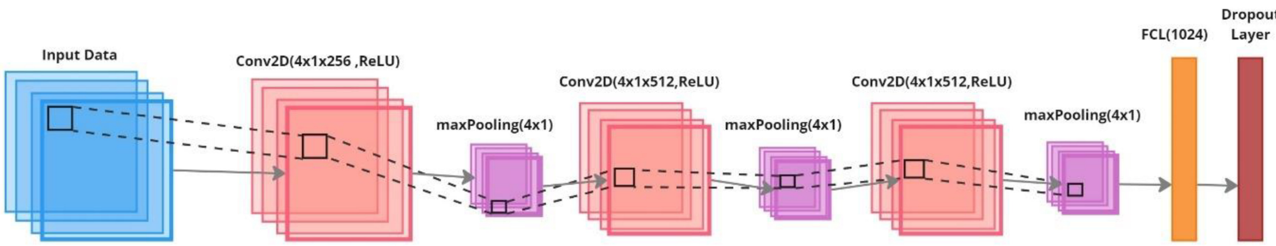

Fig. 3. CNN architecture.

Table 2. CNN model parameters.

Parameter	Value
Input Size	[Size of XTrain, 1, 1]
Convolutional Layer 1 Filter Size	4×1
Convolutional Layer 1 Number of Filters	256
Pooling Layer 1 Size	4×1
Convolutional Layer 2 Filter Size	4×1
Convolutional Layer 2 Number of Filters	512
Pooling Layer 2 Size	4×1
Convolutional Layer 3 Filter Size	4×1
Convolutional Layer 3 Number of Filters	512
Pooling Layer 3 Size	4×1
Fully Connected Layer Size	1024
Dropout Rate	0.5
Final Output Layer Size	[Size of YTrain, 3]
Optimizer	RMSprop
Mini-Batch Size	500
Max Epochs	40
Initial Learning Rate	0.001
Learning Rate Drop Factor	0.5
Learning Rate Drop Period	10 Epochs
L2 Regularization	0.0001
Execution Environment	GPU

activations, and max-pooling layers follow each convolutional layer, which progressively down sample the feature maps while retaining key information. The final set of features is flattened and passed to a fully connected layer, which acts as a dense classifier to output predictions related to the optimal beamforming configuration.

The output of the fully connected layer is designed to match the dimensionality of the target data, which corresponds to the achievable data rates for each user in the system. A regression layer is employed as the final layer, which computes the loss between the predicted and actual rates, and adjusts the model parameters accordingly through backpropagation. The model is trained using the RMSprop optimizer, with a mini-batch size to balance memory usage and training speed. Training is performed over several epochs, and the model's performance is evaluated on a separate validation set.

The CNN-based model is proposed for fast processing of the high-dimensional channel data and better throughput prediction in RIS-assisted communication system with optimal beamforming configuration. Using the spatial filtering capability of convolutional layers, the model is able to learn certain signal propagation patterns that provide it a good generalized performance across different channel conditions.

3.6. Hybrid convolution neural network and long short-term model

The CNN-LSTM model is an enhancement of both the CNN and LSTM to enable efficient generation of beamforming patterns in dynamic communication systems. It characterizes this hybrid architecture as especially suitable for situations where the characteristics of the wireless channel are time-varying, for example in mobile communication or variable interference.

CNN Layers: As for the CNN component of the model, the primary purpose is to identify spatial characteristics derived from the CSI signal composed of channel state information across the subcarriers and antennas. These features help capture the spatial dependence of the data required for the determination of the beam forming vectors.

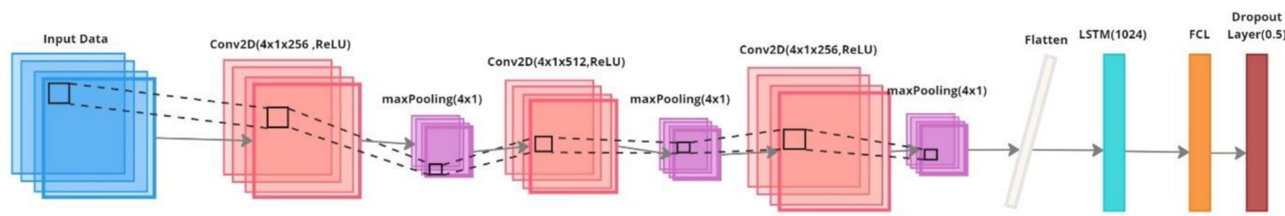

Fig. 4. CNN-LSTM Architecture.

Table 3. CNN-LSTM model parameters.

Parameter	Value
Input Size	[Size of XTrain, 1, 1]
Convolutional Layer 1 Filter Size	4×1
Convolutional Layer 1 Number of Filters	256
Pooling Layer 1 Size	4×1
Convolutional Layer 2 Filter Size	4×1
Convolutional Layer 2 Number of Filters	512
Pooling Layer 2 Size	4×1
Convolutional Layer 3 Filter Size	4×1
Convolutional Layer 3 Number of Filters	256
Pooling Layer 3 Size	4×1
Flatten Layer	Yes
LSTM Layer Size	1024 units
Fully Connected Layer Size	[Size of YTrain, 2]
Dropout Rate	0.5
Optimizer	RMSprop
Mini-Batch Size	500
Max Epochs	20
Initial Learning Rate	0.001
Learning Rate Drop Factor	0.5
Learning Rate Drop Period	10 Epochs
L2 Regularization	0.0001
Execution Environment	GPU

LSTM Layer Integration: After that, the features extracted by the CNN layers are taken to LSTM layers for its processing as shown in Fig. 4 Some of the features include the quality, signal strength, stillness, file transfer data rate, and boot time To process these features over time and to capture the temporal variations in the communication channel, the LSTM plays the following role. This is crucial for anticipating the trends in the channel state and hence giving the right beamforming policy.

Role of LSTM Layers: LSTM layers operate as an efficient means of remembering useful information about state at long intervals to sustain state information smoothness across time steps. Due to this capability, the LSTM can correctly predict the future channel states based on a number of past instances making it easier for the model to adapt the beamforming vectors with the varying channel conditions.

Integration with CNN Layers: That is, CNN layers and LSTM layers are integrated by operating CNN layers, followed by flattening the output and applying it as a sequential input to the LSTM. Such formation guarantees that the received spatial features are processed by LSTM to detect temporal dependences, which are valuable for understanding all the spatial and temporal relations occurring in the channel. The LSTM outputs are utilized with the help of an algorithm to decide the phase shift, which is favorable for beam forming.

By explaining the CNN-LSTM model in detail, the paper traces the individual contributions of the LSTM layers in this communication setting, laying out how the model amalgamates spatial and temporal variation who works together to optimize the beamforming process.

After passing through the CNN layers, the feature maps are flattened into a vector format to prepare them for input into the LSTM component. The LSTM layer is included to model the sequential nature of the channel data, particularly the temporal variations in the communication environment. The LSTM layer processes the temporal sequences of the flattened feature maps, learning the long-term dependencies and correlations between different channel conditions. This is crucial for beamforming in RIS systems, where the signal quality at the receiver depends on the coordinated interaction of multiple signals over time.

Following the LSTM layer, a fully connected layer is ap- plied to the output, which combines the learned features and produces a set of predictions. The fully connected layer helps integrate the spatial and temporal features learned by the CNN and LSTM, resulting in a final output that corresponds

Algorithm 1: Deep Learning-Based Beamforming Optimization in RIS-Assisted Communication Systems

Input:
– ‘L’: Number of paths
– ‘My, Mz’: Dimensions of the reflecting surface in y and z axes
– ‘M_{bar}’: Number of selected RIS reflecting elements
– ‘K_DL’: Number of OFDM subcarriers
– ‘Pt’: Transmit power
– ‘k_{beams}’: Number of top predicted beams
– ‘Training_Size’: Number of training samples

1. System Parameters Setup:
– Define DeepMIMO dataset scenario and bandwidth (‘BW’), number of subcarriers (‘K’), noise figure (‘NF’), and other parameters.
– Calculate noise power ‘noise_power_dB’ using the equation:

$$noise\ power\ dB = -204 + 10\log_{10}\left(\frac{BW}{K}\right) + NF - Process\ Gain$$

– Compute Signal-to-Noise Ratio (SNR) using:

$$SNR = (10^{0.1x(-noise\ power\ dB)})(10^{0.1(Gt+Gr+Pt)})^2$$

2. Beamforming Codebook Generation:
– Generate beamforming codebook (‘BF_codebook’) based on UPA dimensions and oversampling factors.
3. Dataset Generation using DeepMIMO:
– Generate channels for users ‘Ut’ and ‘Ur’ using DeepMIMO.
– Normalize the channel by calculating the maximum value ‘Delta_H_max’ from the product of ‘Ht’ and ‘Hr’ (user channel matrices).
4. Deep Learning Input Construction:
– For each user, compute the channel matrix ‘H_bar’ by adding Gaussian noise to the RIS elements.
– Normalize input ‘DL_input’ as:

$$DL\ input = \frac{DL\ input}{\Delta_{H_{bar\ max}}}$$

5. DL Beamforming Training:
– **For each training size (‘Training_Size’):**
– Split the dataset into ‘Training_Ind’ (training set) and ‘Validation_Ind’ (validation set).
– Construct inputs ‘XTrain’, ‘YTrain’, ‘XValidation’, and ‘YValidation’.
– Build the CNN-LSTM architecture with convolutional, batch normalization, and max-pooling layers.
– Define training options, such as learning rate and batch size.
6. Network Training:
– Train the network using the ‘trainNetwork()’ function with the constructed layers and options.
– Predict achievable rates for both ‘DL-based’ and ‘OPT-based’ beamforming:
– Compute rate using:

$$Rate\,\log_2(1 + SNR(SNR\ sqrt\ var)^2)$$

7. Achievable Rate Calculation:
– **For each validation sample:**
– Identify the top ‘k_{beams}’ from the predicted beams (‘Indmax_DL’).
– Compute the rate for each beam and calculate the average rate for ‘DL’ and ‘OPT’ beamforming strategies.
– Store the mean rates in ‘Rate_DL’ and ‘Rate_OPT’.
8. Output:
– **Return** ‘Rate_DL’: Average data rate achieved using DL-based beamforming.
– **Return** ‘Rate_OPT’: Optimal achievable rate using conventional beamforming methods.

to the achievable data rates for each user in the system. A regression layer is used to compute the loss between the predicted data rates and the actual values, allowing the model to optimize itself through backpropagation.

Training the CNN-LSTM model involves the use of the RMSprop optimizer, with a set of hyperparameters including a defined learning rate, mini-batch size, and a regularization term to prevent overfitting. The model is trained for a specific number of epochs, during which the validation performance is monitored to ensure that the model generalizes well to un- seen data. By combining the strengths of CNN and LSTM architectures, the hybrid model effectively captures both spatial and temporal characteristics of the wireless channel, enabling accurate predictions of optimal beamforming configurations in the RIS-assisted system. This results in enhanced communication performance, improving data throughput and signal quality for users.

4. Comparative analysis of DL-based beamforming with traditional beamforming methods

To provide further insights, DL-based beamforming is not only compared to genie-aided schemes but also other state of the art conventional techniques based on optimization theory and heuristic methods. Optimization algorithms such as SDR and gradient descent optimization is precise but complex and non-scalable. Greedy, tabu search are other examples of near optimal solutions in that they are fast, scalable but tends to be sub-optimal and less versatile. On the other side, due to its data centered framework, DL based beamforming is highly flexible to environment changes and achieves near optimal performance. Similarly, as detailed in the results section of this research analysis, the strengths and limitations of each of the methods discussed were balanced such as to provide a holistic metric of the scalability and suitability for various wireless communication applications of each solution.

5. Results and discussion

5.1. Parameter simulation

The simulation setup utilizes the DeepMIMO dataset for channel modeling. The scenario selected for the experiments is “O1_28” from the DeepMIMO dataset, with three active BS configured to transmit. The antenna spacing is set at half the wavelength, with a total bandwidth of 100 MHz. The user data is based on row 850, with the user position identified as element 90 in the dataset. The receiver is defined across rows 1000 to 1200. In this study, MATLAB 2023a was utilized as the primary software platform for the implementation and evaluation of the proposed DL models. MATLAB’s robust environment allowed for seamless integration of DL toolboxes, which facilitated the development of both Convolutional Neural Network (CNN) and hybrid CNN-Long Short-Term Memory (CNN- LSTM) architectures. The 2023a version provided advanced computational capabilities and optimization tools, enabling efficient handling of large datasets and complex simulations, including the processing of channel state information (CSI) from the DeepMIMO dataset and the optimization of beam- forming parameters for RIS-assisted communication systems. For validation purposes, the dataset size is set to 6200 samples.

The system supports 512 subcarriers, and the mini-batch size for processing is fixed at 500. The number of Reconfigurable Intelligent Surface (RIS) reflecting elements is represented by the product of the dimensions Mx, My, and Mz. This setup is designed to

Table 4. Comparative analysis of beamforming methods.

Aspect	Optimization-based Beamforming	Heuristic Beamforming Approaches	DL-based Beamforming
Focus	Maximizes SINR or minimizes transmit power	Provides practical, faster solutions	Learns from data to adapt dynamically
Precision	High precision with theoretical guarantees	Suboptimal compared to optimization	Near-optimal with sufficient training
Computational Complexity	High, often not scalable	Lower, more scalable	Moderate, depends on training overhead
Scalability	Limited with increasing users or antennas	Better scalability	Scalable with trained models
Adaptability	Limited adaptability to dynamic environments	Limited adaptability	High adaptability to changing conditions
Performance in Dynamic Environments	Moderate, requires frequent recalibration	Moderate, lacks dynamic adjustment	High, adjusts in real-time

Table 5. Simulation parameters and values [28].

Parameter	Value
Scenario	O1_28 (DeepMIMO)
Active Base Stations	3
Antenna Spacing	0.5 (relative to wavelength)
Bandwidth (BW)	100 MHz
User Row (Transmitter)	850
User Position (Transmitter Element)	90
Receiver Rows (Ur)	1000 to 1200
Validation Dataset Size	6200
Number of Subcarriers (K)	512
Mini-Batch Size	500
Number of RIS Reflecting Elements (M)	Mx * My * Mz
Antenna Gains (Gt, Gr)	3 dBi
Noise Figure (NF)	5 dB
Process Gain	10 dB
Noise Power (BW)	Calculated based on system parameters
Number of User Pairs	(Ur_rows(2) - Ur_rows(1)) * 181
Over-Sampling Factors (x, y, z)	1
Number of Paths (L)	Between 1 and 25

simulate multi-path signal propagation with a set of parameters that maximize the efficiency of the model.

The signal-to-noise ratio (SNR) is derived based on various parameters, including the antenna gains Gt and Gr, which are both set to 3 dBi, and a noise figure of 5 dB at the user equipment. The process gain during channel estimation is set to 10 dB. The noise power in the system is calculated using the total bandwidth, number of subcarriers, noise figure, and processing gain, leading to an SNR computation. Furthermore, noise power per subcarrier is considered to model noisy channel conditions effectively.

The user pairings are generated randomly, with the total number of pairs being based on the range between the user rows and the spatial configuration. To accommodate the dynamic nature of the simulation, a random permutation of these user pairs is generated. Various over-sampling factors are set for beamforming in the x, y, and z directions to ensure precise control over the signal propagation.

The input data for the system is structured based on the DeepMIMO dataset parameters, including the number of antennas on the x, y, and z axes, with an antenna spacing of half the wavelength. The dataset supports OFDM with a specific number of subcarriers, bandwidth in GHz, and a maximum number of paths considered for each signal. The system adjusts dynamically based on these input parameters to generate realistic channel data, which is then used for performance evaluation.

5.2. Dataset

The outdoor scenario depicted in the dataset involves two streets intersecting at one point. It features 18 BS, distributed across the main street and the second street. Each BS is positioned at a height of 6 meters, and they utilize isotropic antenna arrays. The BS along the main street are positioned in two rows, one on each side of the street, with 12 BS in total. The separation between BS on one side of the street and the other is approximately 52 meters, ensuring that the network provides wide coverage. Some BS are separated by 100 meters or 62 meters, depending on their locations along the street. The second street, which intersects with the main street, is covered by six BS, three on each side. These BS have a separation of 150 meters, ensuring extensive coverage along the second street as well.

The user distribution is divided into three grids, each positioned across different sections of the streets. The first user grid is distributed along the main street and covers a length of 550 meters with a width of 35 meters. This user grid consists of nearly half a million users, with a uniform arrangement, having users spaced 20 centimeters apart along rows. The second user grid is positioned along the southern side of the cross street, with a similar arrangement of rows and spacing, covering over 199,000 users. The third user grid is placed along the second cross street, containing more than 487,000 users, but with a closer user arrangement, having a spacing of 10 centimeters between users. Each user in the dataset is equipped with an isotropic antenna at a height of 2 meters, ensuring uniform communication conditions.

The site plan includes a detailed representation of the dimensions of the streets and buildings. The main street is 600 meters long and 40 meters wide, while the cross streets have similar widths but shorter lengths. Buildings are placed along both sides of the streets, with consistent dimensions for their bases. Each building's height varies, providing diverse

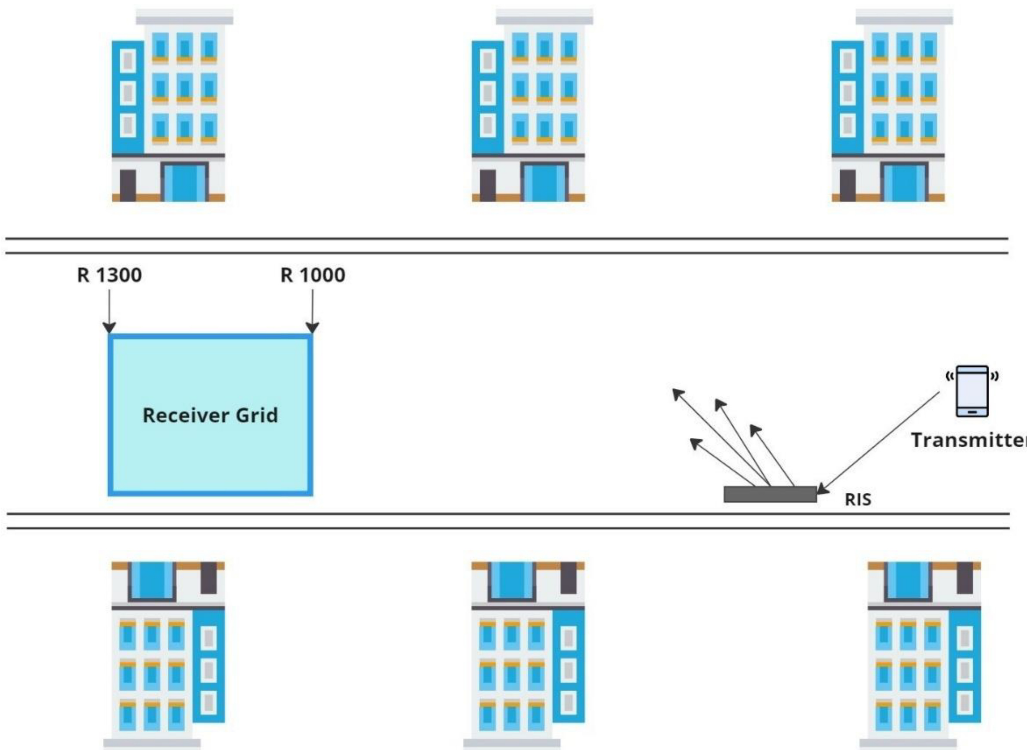

Fig. 5. The assumed RIS ray tracing operation [12].

propagation conditions for the wireless signals. The propagation model used in this scenario assumes that signals can reflect up to four times before reaching the receiver, making it a complex environment for signal processing and analysis. In Fig. 5, the layout of the receiver grid is depicted, showing the distribution of users along the streets and the positioning of BS and the transmitter. The receiver grid is highlighted between rows 1000 and 1300, and signals from the transmitter are reflected off the Reconfigurable Intelligent Surface (RIS) to reach the grid. This setup represents how signals interact with urban environments, facing obstructions like buildings, before reaching the receiver.

The DeepMIMO dataset is designed to support the simulation of realistic wireless communication environments, especially for the study and development of emerging technologies such as massive MIMO and (RIS). The dataset offers a highly detailed and configurable environment, allowing researchers to simulate various scenarios across different operating frequencies, including 3.4 GHz, 3.5 GHz, 28 GHz, and 60 GHz. One of the most significant features of DeepMIMO is its large- scale user grid, which includes over a million candidate users distributed uniformly across urban environments, such as streets and intersections. This comprehensive setup enables the simulation of real-world conditions, with factors such as path loss, shadowing, and multipath reflections incorporated into the channel models. DeepMIMO also provides flexibility in defining the configuration of the system, such as the number of BS, their positions, antenna array designs, and user distribution, making it a powerful tool for evaluating next-generation wireless networks. Through the integration of DeepMIMO, complex environments can be modeled with high fidelity, facilitating advanced research in beamforming, channel estimation, and the design of RIS-enabled communication systems.

6. Experiments results

6.1. CNN experiment results

The experiment results depicted in Figs. 6 to 10 provide an in-depth analysis of the achievable rate for different dataset sizes and system configurations, including varying numbers of active elements, power levels, and system parameters.

In Fig. 6, As the number of active elements increases, the achievable rate improves significantly, with the highest rate achieved when $M_{bar} = 8$. Achievable rate for varying numbers of active

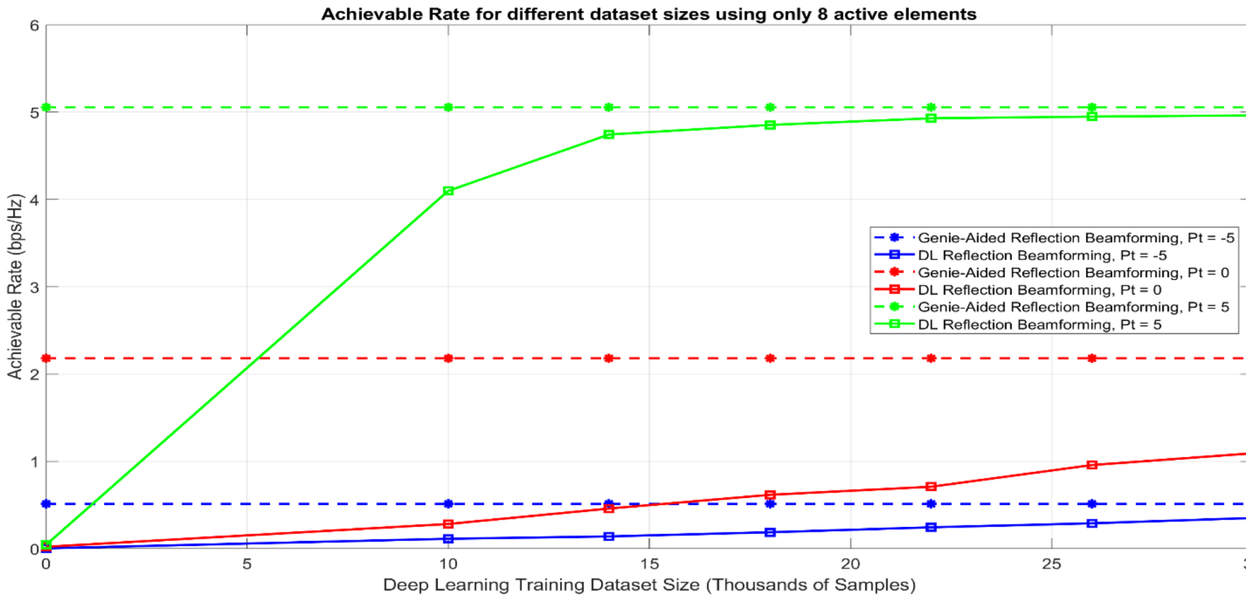

Fig. 6. Achievable rate for different dataset sizes using only 8 active elements, with power transmission values of Pt = −5, 0, and 5. The DL reflection beamforming approaches the performance of genie-aided beamforming, with the highest rate achieved at Pt = 5.

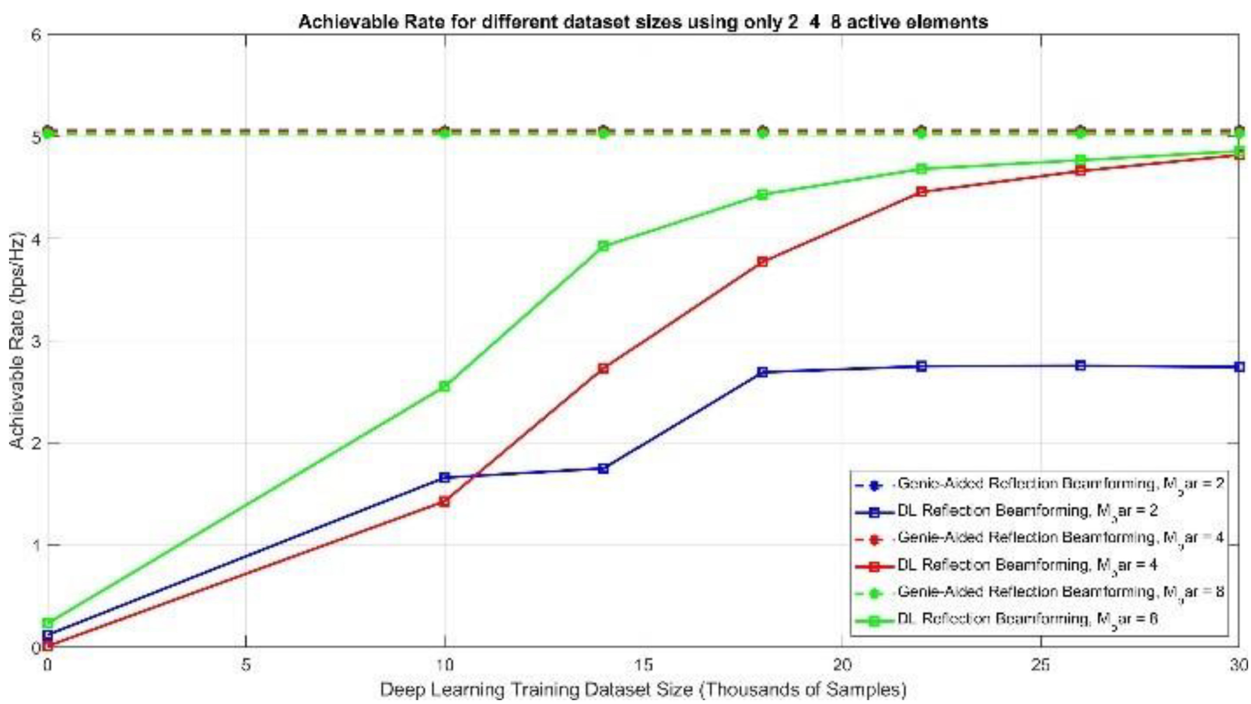

Fig. 7. Achievable rate for varying numbers of active elements (M_{bar} = 2, 4, and 8) over different dataset sizes.

elements (M_{bar} = 2, 4, and 8) over different dataset sizes. As the number of active elements increases, the achievable rate improves significantly, with the highest rate achieved when $M_{bar} = 8$ using only 8 active elements and different transmission powers, with values of Pt = −5, 0, and 5. The green curve, corresponding to Pt = 5, demonstrates the highest achievable rate, stabilizing at around 5 bps/Hz as the dataset size increases beyond 10,000 samples. The DL-based reflection beamforming approach closely follows the genie-aided re- flection performance. For Pt = 0, the performance stabilizes around 1.05 bps/Hz, and for Pt = −5, the rate reaches around 0.43 bps/Hz.

In Fig. 7, the impact of different numbers of active elements, specifically M_{bar} = 2, 4, and 8, is studied. The green line, corresponding to 8 active elements,

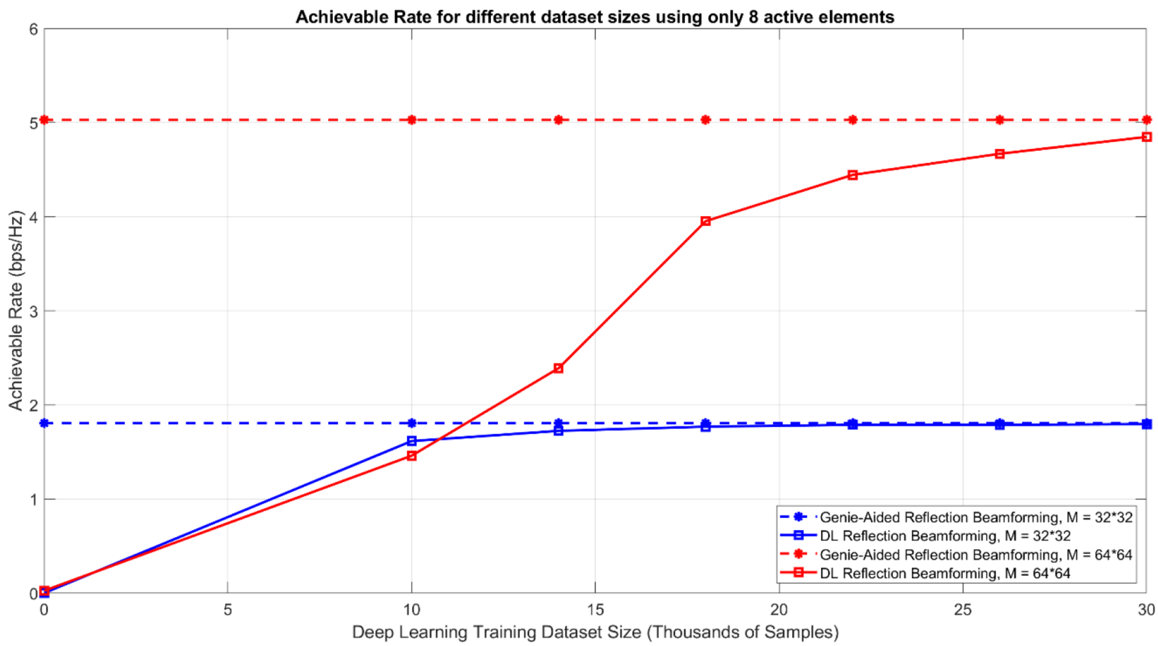

Fig. 8. Achievable rate for different dataset sizes using 8 active elements, comparing different RIS configurations with M = 32 × 32 and M = 64 × 64. Larger RIS sizes provide significant gains in the achievable rate, with DL beamforming approaching genie-aided perf.

achieves the highest rate of about 4.8 bps/Hz, demonstrating the significant gain achieved by increasing the number of active elements. The red and blue lines, representing $M_{bar} = 4$ and $M_{bar} = 2$, respectively, show progressively lower performance, with the rate reaching approximately 4.78 bps/Hz and 2.8 bps/Hz for 4 and 2 active elements, respectively.

Fig. 8 examines the effect of different total numbers of RIS reflecting elements, with configurations of M = 32 × 32 and M = 64 × 64. As the figure illustrates, increasing the size of the RIS substantially improves the achievable rate.

The DL reflection beamforming achieves approximately 5 bps/Hz when M = 64 × 64, while for M = 32 × 32, the achievable rate reaches around 1.8 bps/Hz. This emphasizes the advantage of larger RIS configurations in terms of performance.

Fig. 9 focuses on the variation in k_{beams}, where values of 1, 2, and 3 are considered. With 3 beams, the system attains a high rate of 4.7 bps/Hz. With fewer beams ($k_{beams} = 2$ and 1), the performance slightly decreases, reaching around 4.8 bps/Hz and 4.76 bps/Hz, respectively. This confirms that increasing the number of beams enhances the system's ability to achieve higher rates.

Finally, Fig. 10 highlights the effect of varying the number of paths, L, with values of L = 1, L = 2, and L = 5. For L = 5, the system achieves a maximum rate of 4.8 bps/Hz, while for L = 2, the rate stabilizes at around 2.7 bps/Hz. The single-path scenario, L = 1, shows the lowest performance, with a rate of about 1.2 bps/Hz. The genie-aided reflection beamforming closely follows the DL beamforming performance in all cases.

The results indicate unambiguously that the larger number of active elements, beams and paths lead to substantially higher achievable rate. Significant performance gains are available thanks to RIS configurations with more memory, additional power and paths. This result demonstrates that DL-based beamforming can efficiently reproduce the genie-assisted system performance in diverse environments.

6.2. CNN-LSTM results

In the CNN-LSTM experiment results, we observe how different parameters influence the achievable rate in the system, with a focus on DL dataset sizes and the number of active elements.

In Fig. 11 shows the achievable rate for different dataset sizes using only 8 active elements and varying power transmission values (Pt = −5, 0, and 5). As expected, the achievable rate is higher for Pt = 5, reaching nearly 5 bps/Hz, while for Pt = −5, the rate re- mains lower, barely reaching 0.5 bps/Hz. The DL reflection beamforming closely follows the performance of genie-aided reflection beamforming, particularly when the dataset size exceeds 15,000 samples, showing convergence at higher data volumes.

In Fig. 12, the achievable rate is analyzed for different numbers of paths (L = 1, 2, and 5) using

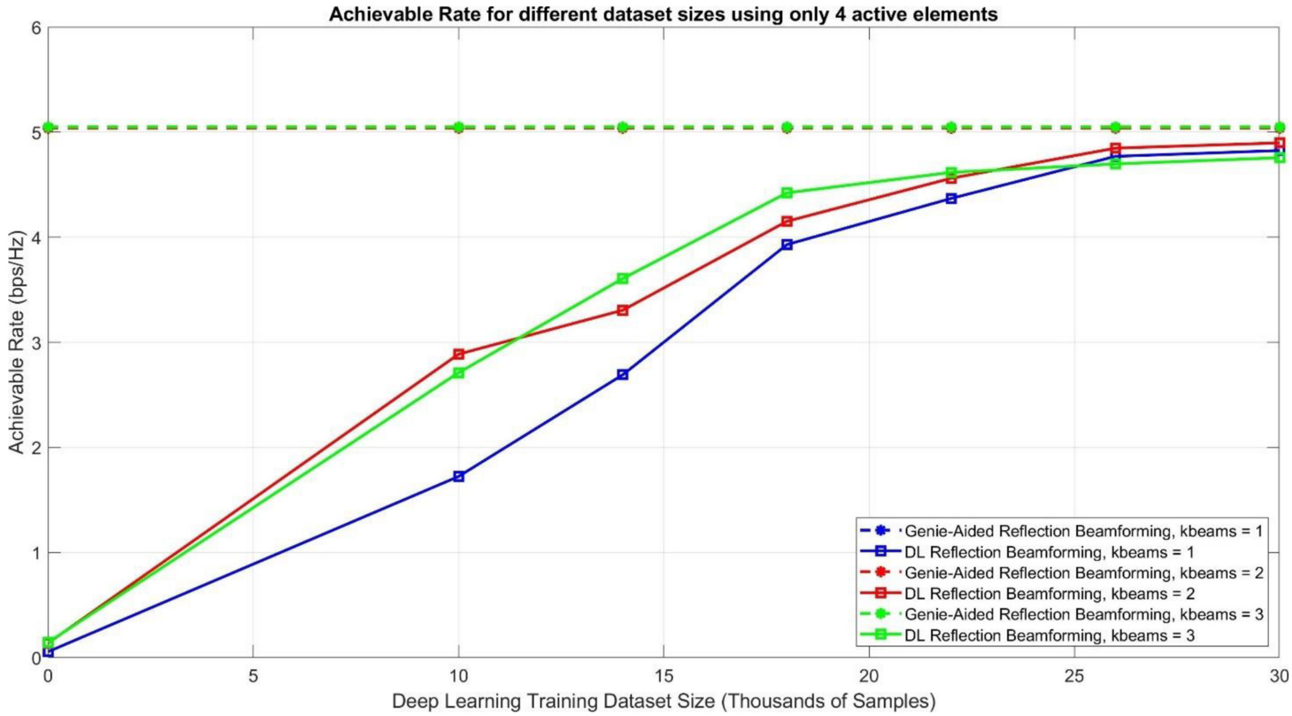

Fig. 9. Achievable rate for varying numbers of beams (k_{beams} = 1, 2, and 3) using 4 active elements. Higher numbers of beams improve the achievable rate, with the maximum rate observed when k_{beams} = 3.

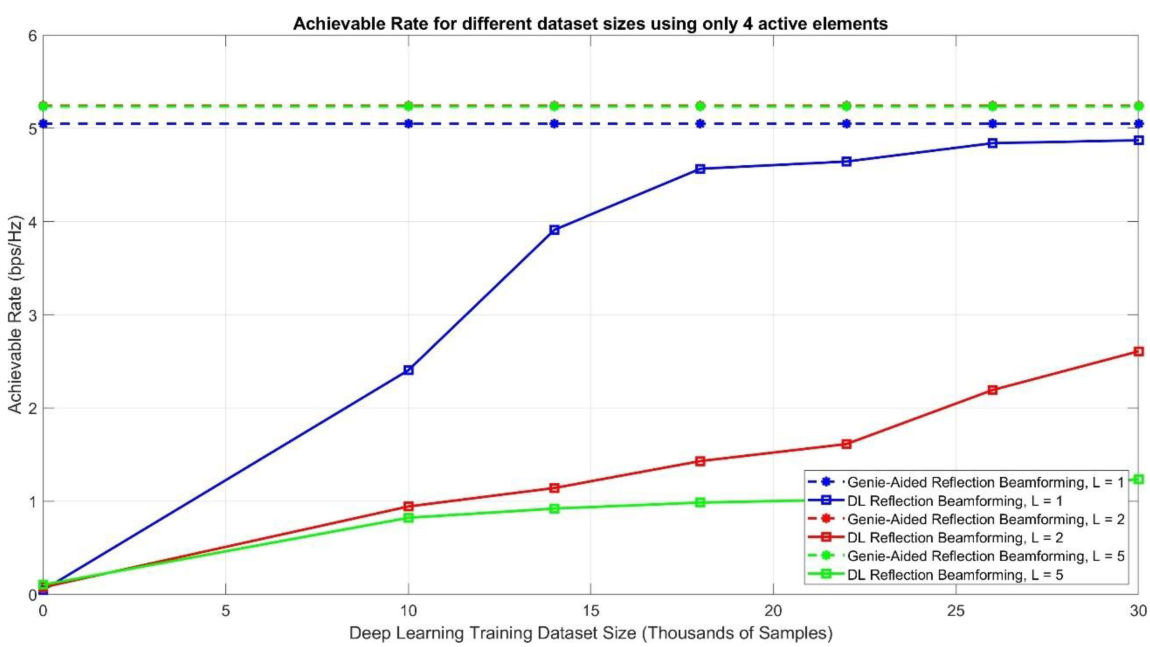

Fig. 10. Achievable rate for different dataset sizes with variations in the number of paths (L = 1, 2, and 5), using 4 active elements. As the number of paths increases, the achievable rate improves, with the best performance achieved for L = 5.

only 4 active elements. The results indicate that increasing the number of paths significantly improves the achievable rate. With L = 1, the achievable rate almost reaches 5 bps/Hz, while for L = 5, the rate plateaus just above 4.5 bps/Hz. The DL reflection beamforming once again approaches the performance of the genie-aided beamforming particularly for larger dataset sizes.

Fig. 13 illustrates the impact of different RIS sizes (M = 32 × 32 and M = 64 × 64) on the achievable

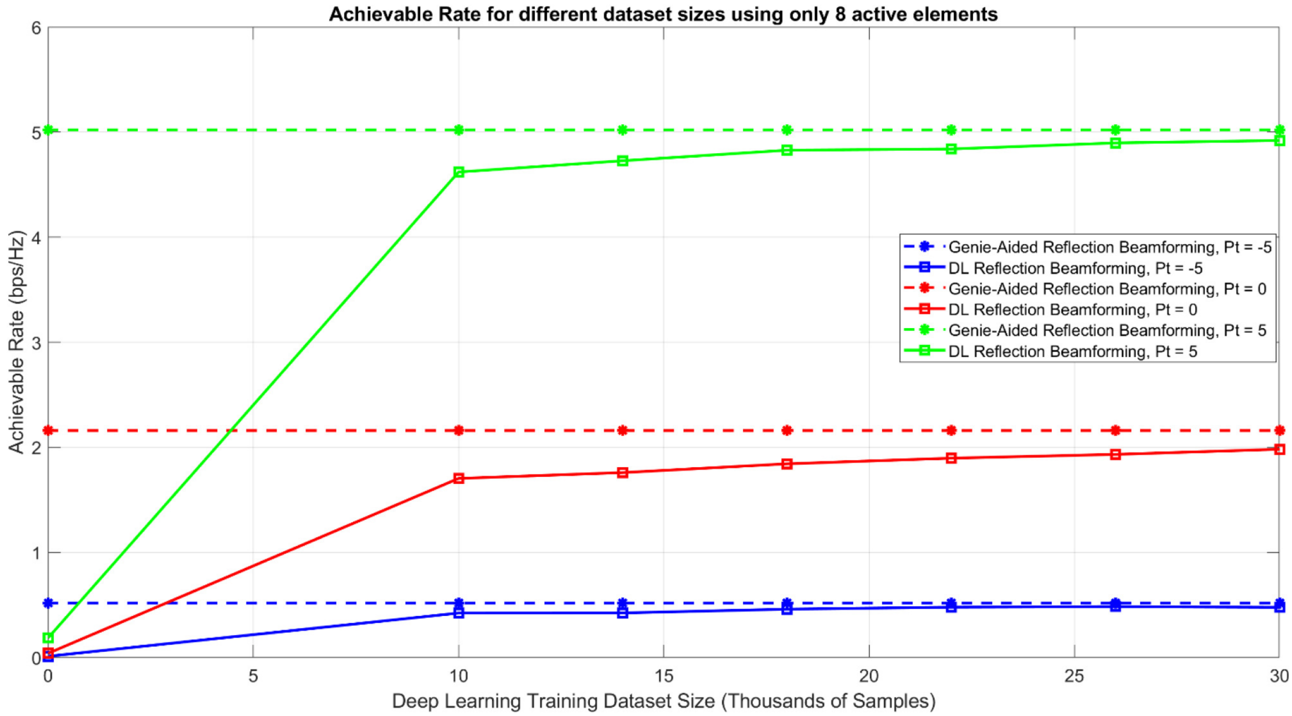

Fig. 11. Achievable rate for different dataset sizes using 8 active elements.

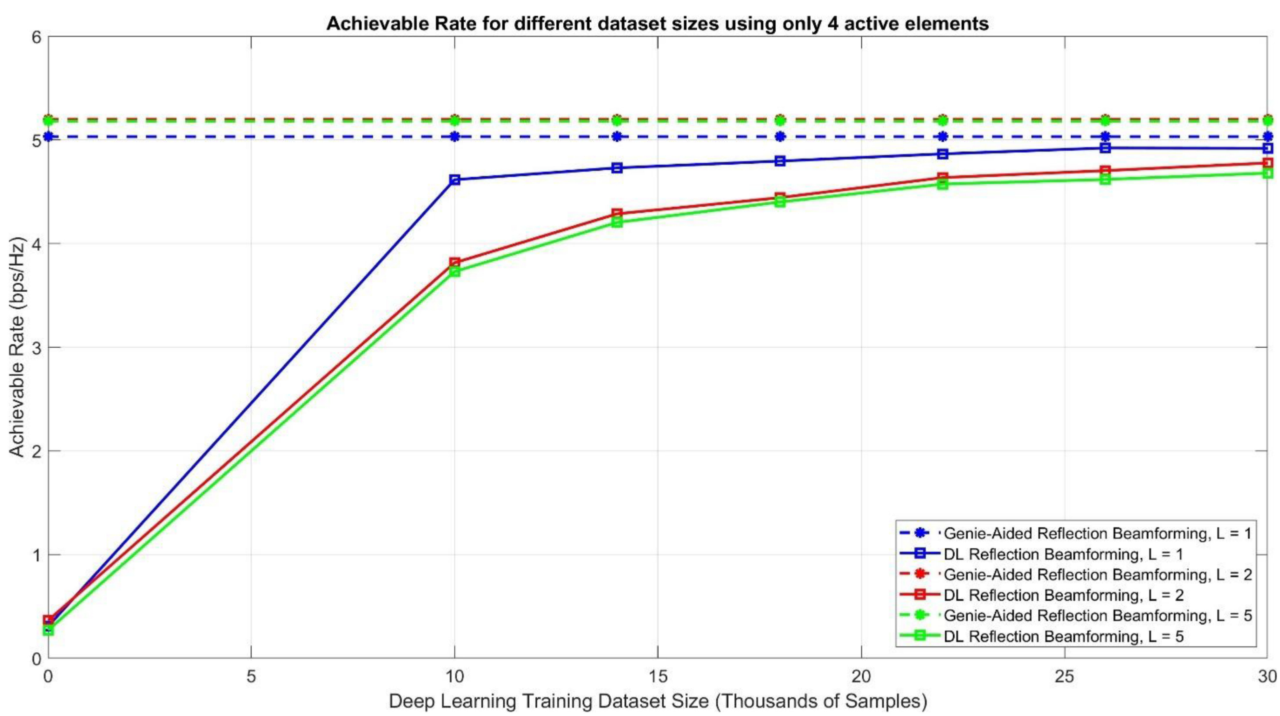

Fig. 12. Achievable rate for different numbers of paths (L = 1, 2, 5) using 4 active elements. Increasing the number of paths leads to higher achievable rates, with the deep learning reflection beamforming converging to the genie-aided beamforming as the dataset.

rate with 8 active elements. For the larger RIS ($M = 64 \times 64$), the achievable rate improves significantly, reaching nearly 5 bps/Hz with a dataset size of 30,000 samples. On the other hand, the smaller RIS ($M = 32 \times 32$) achieves a lower rate of around 1.8 bps/Hz. This shows the benefit of scaling up the RIS size for better performance in wireless communication systems.

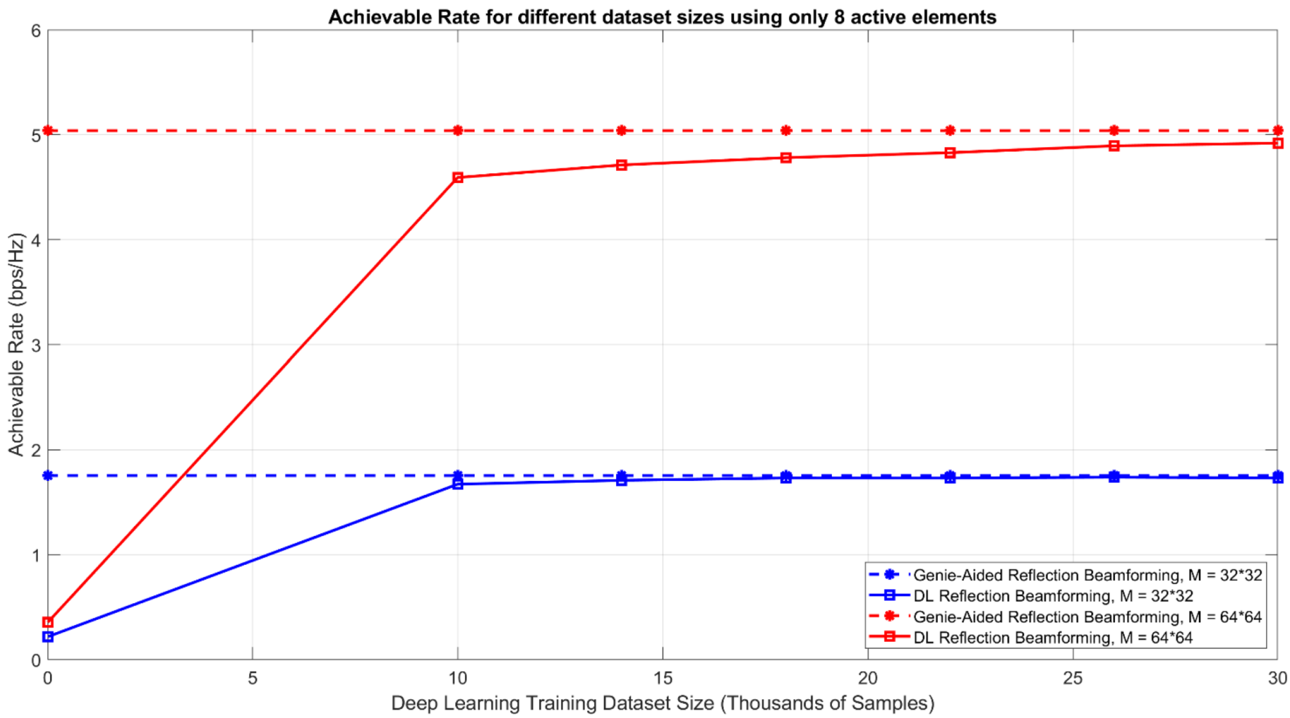

Fig. 13. Achievable rate for different RIS sizes ($M = 32 \times 32$ and $M = 64 \times 64$) using 8 active elements. A larger RIS significantly improves the achievable rate, particularly as the deep learning dataset size increases.

Fig. 14, we analyze the achievable rate for different numbers of active elements ($M_{bar} = 2, 4,$ and 8) as the dataset size increases.

The figures show that the increased number of active elements also increases the rate, with $M_{bar} = 8$ nearly reaching 5 bps/Hz, and the achievable rate achieving around 2.3 bps/Hz for $M_{bar} = 2$. This emphasizes the necessity to employ a higher fraction of active elements in achieving higher rates, especially when DL tools are available for beamforming optimization. Meanwhile,

Fig. 15 shows the achievable rate versus number of beams ($k_{beams} = 1, 2$ and 3) using 4 actives elements in total for reference. The rate improves, as we would expect, by increasing the number of beams. The system converges around 5 bps/Hz for $k_{beams} = 3$ and the plateaus at 4.8 bps/Hz for $k_{beams} = 1$. Hence, more beams for beamforming can provide improved performance in such challenging wireless environment as illustrated by these results.

To summarize, a larger dataset size with more active elements and optimized parameters of the number of beams and paths combined with CNN-LSTM model can improve achievable rate in the system. Experimental outcomes demonstrate all figures that the beamforming methods based on DL perform in synergy with genie-aided techniques especially as data dimension grows, validating thus the effectiveness of this methodology for wireless communication systems.

6.3. Trade-offs in enhancing ris-assisted communication systems

The analysis of the extension of the RIS-assisted communication systems reveals that more active elements are favorable for the manipulation of electromagnetic waves and system performance while the system complexity and energy consumption also increase accordingly. Every additional peak power element required increases the need for the control circuit, thus increasing the power consumption and overall cost of operation. One might say that with more active elements to be managed at any time, the system may need more sophisticated signal processing algorithms that can complicate the task even further in terms of the required computations.

More size of RISs and using more beams allow more possibilities to control the signal and serve more clients, but it is not without its problems. Increasing the number of elements in the RIS array can cause problems with the installation and the costs associated with manufacturing and deployment of the RISs also rise with an increasing number of beams, they can interferences with each other, thus,advanced beamforming strategies are needed. Finally, enabling

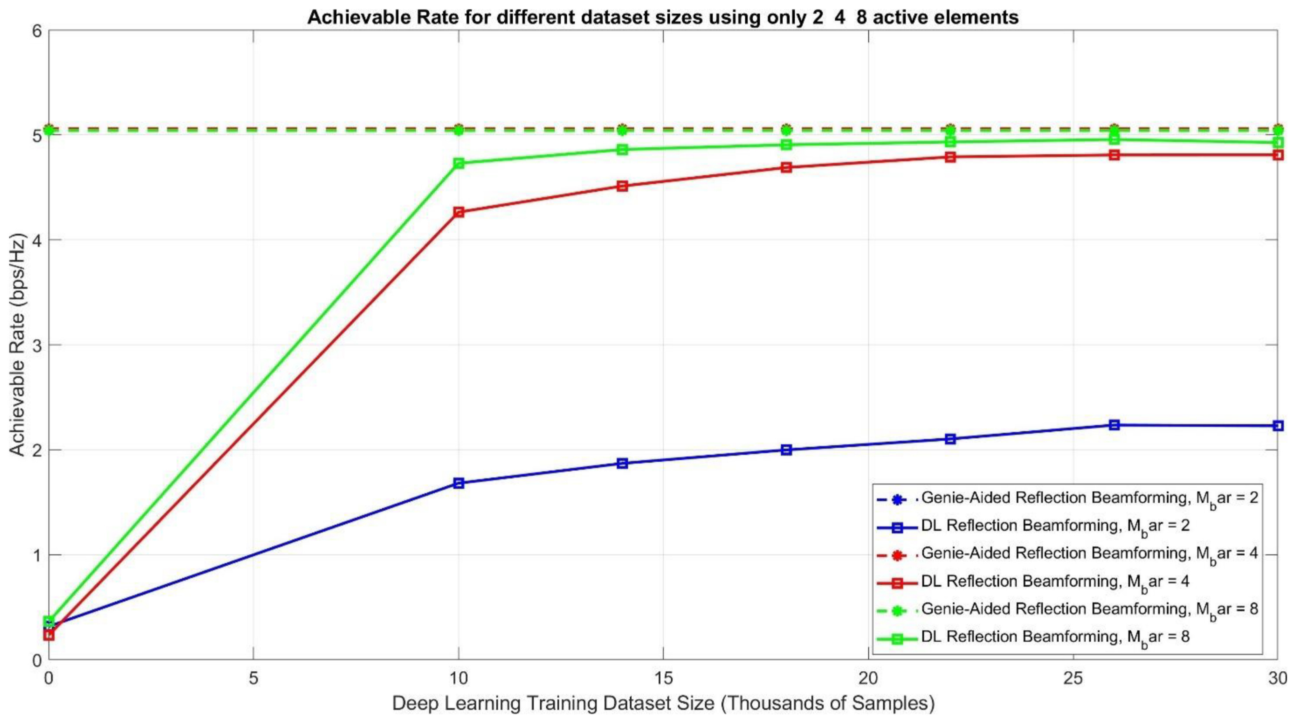

Fig. 14. Achievable rate for different numbers of active elements (M_{bar} = 2, 4, 8) with varying dataset sizes. Increasing the number of active elements results in higher achievable rates, showing the benefits of more active components in the system.

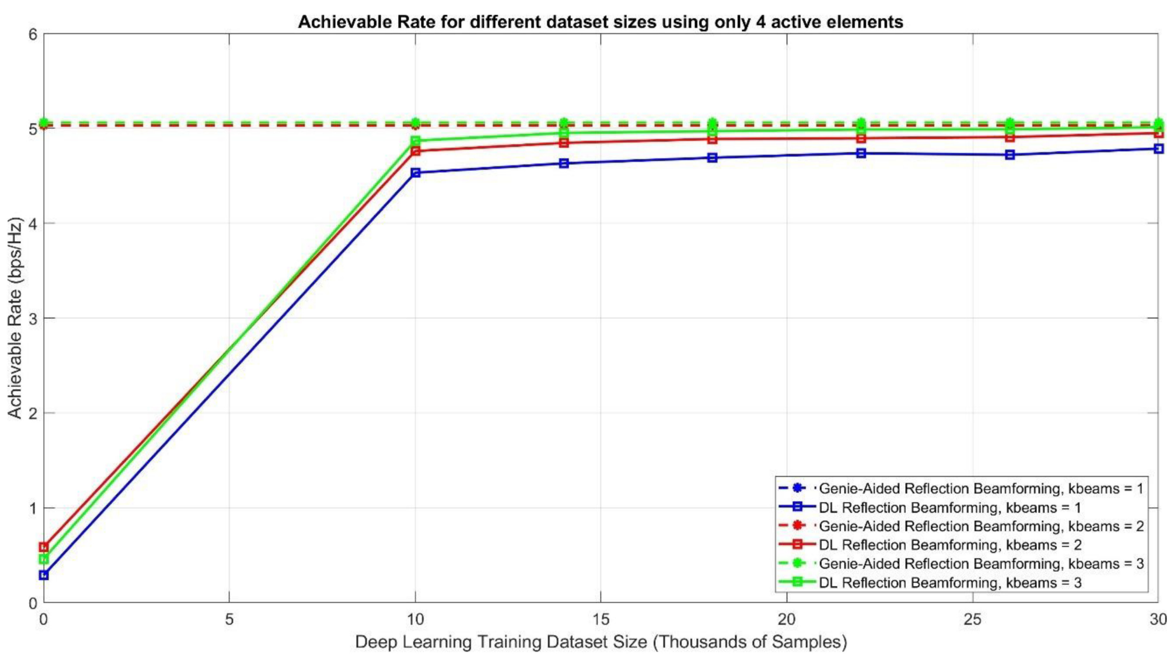

Fig. 15. Achievable rate for different numbers of beams (k_{beams} = 1, 2, 3) using 4 active elements. More beams lead to higher rates, with the deep learning beamforming closely matching the genie-aided approach as the dataset size grows.

multiple beam form operation also has an adverse effect on the overall energy consumption of a system.

6.4. Discussion

The experimental results provided in the previous sections demonstrate the effectiveness of both CNN and CNN-LSTM architectures for enhancing the achievable rate in wireless communication systems using DL based beam- forming. The experiments analyzed various configurations, including different numbers of active elements, paths, beams, and power transmission levels. These analyses revealed several key insights and observations that help in understanding the performance of these models.

One of the most significant findings is the impact of dataset size on the performance of DL based beamforming. As demonstrated across multiple figures, larger datasets enable the models to closely approach the performance of genie-aided reflection beamforming, particularly when optimizing for parameters like the number of active elements, beams, and paths. For example, in both the CNN and CNN- LSTM experiments, the models with larger datasets were able to achieve higher rates, especially with more active elements and higher transmission power (Pt). This is clearly reflected in Fig. 4 and Fig. 9, where the achievable rate improves significantly with increased training samples, especially for larger values of Pt.

Another important observation is the influence of system parameters, such as the number of active elements (M_{bar}), the total number of RIS reflecting elements (M), and the number of beams (k_{beams}). Fig. 7 and Fig. 14 illustrate that increasing the number of active elements leads to a substantial improvement in the achievable rate, with systems configured with $M_{bar} = 8$ achieving close to 5 bps/Hz. Similarly, Fig. 8 and Fig. 13 show that larger RIS configurations (M = 64 × 64) outperform smaller configurations (M = 32 × 32), emphasizing the benefits of scaling up the size of the intelligent surface for more efficient signal processing and reflection.

Moreover, the number of paths (L) and beams (k_{beams}) also have a large say on how well the system performs. Fig. 9, Fig. 10, Fig. 14 and Fig. 15 all exhibit this consistent fact that higher number of beams and paths cause increase in the achievable rate which indicates a straight forward trend towards more performance with larger values of these parameters. More importantly, Fig. 8 and Fig. 13 show that the near-genie-aided performance of DL reflection beamforming is approached by increasing the number of beams and paths, highlighting the necessity of optimal parameter configuration in practice.

In terms of power transmission (Pt), as indicated in Fig. 6 and Fig. 11, larger Pt values come along with a significantly higher rate able to be reached. For instance, Pt = 5 gives uniformly highest rates across the various experiments, always approaching ∼5 bps/Hz, and has a stable performance in more moderate values (Pt = −5 with significantly worse results compared to optimal cases. This demonstrates the necessity of transmitting power optimization in combination with DL models for achieving optimal system performance.

In comparison of CNN and CNN-LSTM results highlight the ability to not only generalize the DL approach but its resilience as well for beamforming applications. The CNN-LSTM architecture performs slightly better in some setups, especially for larger RIS sizes and multiple paths because it can model temporal dependencies better and learn patterns from the dataset.

However, both architectures exhibit strong potential in approaching genie- aided performance under various conditions, especially as the dataset size increases and system parameters are optimized.

In conclusion, the experiments provide a comprehensive analysis of how different system configurations, dataset sizes, and DL architectures can be used to optimize beamforming in wireless communication systems. The results clearly demonstrate that DL based beamforming is a viable and effective alternative to traditional methods, capable of approaching near-optimal performance with appropriate parameter tuning and dataset size. These findings have significant implications for future wireless networks, where intelligent surfaces and machine learning.

6.5. Challenges and feasibility of real-world deployment

The simulation results show that the proposed DL-based beamforming system is indeed effective as the numerical results reveal but adapting DL for real-world deployment is difficult because most practical networks have their own constraints in terms of physical hardware, environment, latency and interaction with existing networks. When deploying Reconfigurable Intelligent Surfaces (RIS) with active and passive components, issues regarding power consumption, heat dissipation and physical size come into consideration. Also the practical constraints like weather conditions, mobility of the user and interference require high reliability adaptive algorithms. Making beamforming as low-latency as possible may need great hardware investment or edge computing executives or accelerators and must conform to existing wireless standards such as 5G and 6G.

For the purpose of establishing practical usability, more future research should be devoted to cross-environmental experimentations that would involve constructing prototypes of RIS setups in cooperation with industry players.

7. Conclusion

This work has aimed at presenting a review on the deep learning paradigm with emphasis on CNN and the CNN-LSTM framework for the design of beamforming configuration in RIS-based wireless communication system. These advanced architectures of DL have shown a considerable enhancement in achieved rates and overall communication reliability and efficiency under different arrangements. All these results demonstrate that the parameter of larger databases and optimized parameters such as active elements, the size of the RIS, the number of bean, and the number of paths help to increase the effectiveness of this reflection.

In particular, for the CNN and CNN-LSTM models, the results are close to genie-aided beamforming and the CNN-LSTM model is better for dynamic and time-variant scenarios due to the possibility of taking into account temporal dependencies. It can be concluded that the simulation results show that the achievable rates can be improved by increasing the power transmission (Pt), the active elements (N) and the size of RIS (M) in complex environmental areas such as urban areas with multi-path fading. In such cases, the proposed DL models proved their ability to effectively update big data sets and generate accurate predictions for beamforming control.

The proposed DL-based beamforming system is flexible, computationally efficient and scalable and directly applicable to real time large scale wireless networks necessary for future 5G and potentially future 6G systems. The merit of these models is that they relieve the channel estimation burden and the tuning of the RIS reflection matrix, offering a viable way to enhance communication in environments where existing conventional methods fail. However, current early scalability for ultra-large-scale 6G networks is still an open problem owing to highly computational and high memory requirements. Solving these problems will necessitate some strategies like model compression, shared learning via edge and cloud computing, and techniques hybridizing DL with traditional optimization techniques in order to improve competences and minimize cost.

Moreover, the analysis of the theoretically proposed DL-based beamforming method demonstrates its advantage over conventional approaches in terms of adaptability, computational complexity, and scalability. Using the statistical information and updates as to covariance matrices, thereby providing a way to dynamically update certain parameters in high mobility situations but with considerably lower computational cost. The CNN and CNN-LSTM structures improve scalability and generalization; our method is close to genie-aided while using less computation to train. Proofing of similar effectiveness and benefits with other data sets in future and real-world testing may reinforce those perspectives.

In conclusion, this study brings a mark toward implementing DL for the RIS-aided wireless communication in line with demonstrating how the deployment of machine learning can revolutionise beamforming and channel estimation. As for future work, there remains much opportunity for subsequent improvements towards the fine-tuning of these models in real-time using reinforcement learning with real-word scenarios, which have been proven more challenging in terms of mobility and interference management. Finally, the merging of intelligent surfaces with DL models will help achieve the necessary transformation for next generation wireless networks.

References

1. S. Hu, F. Rusek, and O. Edfors, “Beyond massive MIMO: the potential of data transmission with large intelligent surfaces,” *IEEE Transactions on Signal Processing*, vol. 66, pp. 2746–2758, May 2018. DOI: 10.1109/TSP.2018.2816577.
2. M. T. Mamaghani and Y. Hong, “Aerial intelligent reflecting surface enabled terahertz covert communications in beyond-5G Internet of Things,” *IEEE Internet of Things Journal*, vol. 9, pp. 19012–19033, Oct. 2022. DOI: 10.1109/JIOT.2022.3163396.
3. S. Hu, K. Chitti, F. Rusek, and O. Edfors, “User assignment with distributed large intelligent surface (LIS) systems,” in *2018 IEEE 29th Annual International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC)*, pp. 1–6, Sep. 2018. DOI: 10.1109/PIMRC.2018.8580913.
4. Q. Wu and R. Zhang, “Intelligent reflecting surface enhanced wireless network via joint active and passive beamforming,” *IEEE Transactions on Wireless Communications*, vol. 18, pp. 5394–5409, Nov. 2019. DOI: 10.1109/TWC.2019.2936025.
5. J. Qiao, C. Zhang, A. Dong, J. Bian, and M.-S. Alouini, “Securing intelligent reflecting surface assisted terahertz systems,” *IEEE Transactions on Vehicular Technology*, vol. 71, pp. 8519–8533, Aug. 2022. DOI: 10.1109/TVT.2022.3176460.
6. S. Abeywickrama, R. Zhang, Q. Wu, and C. Yuen, “Intelligent reflecting surface: Practical phase shift model and beamforming optimization,” *IEEE Transactions on Communications*, vol. 68, pp. 5849–5863, Sep. 2020. DOI: 10.1109/TCOMM.2020.3001125.
7. X. Shao, C. You, W. Ma, X. Chen, and R. Zhang, “Target sensing with intelligent reflecting surface: Architecture and performance,” *IEEE Journal on Selected Areas in Communications*, vol. 40, pp. 2070–2084, Jul. 2022. DOI: 10.1109/JSAC.2022.3157160.

8. C. You, B. Zheng, and R. Zhang, “Channel estimation and passive beamforming for intelligent reflecting surface: Discrete phase shift and progressive refinement,” *IEEE Journal on Selected Areas in Communications*, vol. 38, pp. 2604–2620, Nov. 2020. DOI: 10.1109/JSAC.2020.3007031.
9. B. Zheng, C. You, and R. Zhang, “Intelligent reflecting surface assisted multi-user OFDMA: Channel estimation and training design,” *IEEE Transactions on Wireless Communications*, vol. 19, pp. 8315–8329, Dec. 2020. DOI: 10.1109/TWC.2020.3024880.
10. T. Jiang, H. V. Cheng, and W. Yu, “Learning to reflect and to beamform for intelligent reflecting surface with implicit channel estimation,” *IEEE Journal on Selected Areas in Communications*, vol. 39, pp. 1931–1945, Jul. 2021. DOI: 10.1109/JSAC.2021.3067430.
11. C. Pan, G. Zhou, K. Zhi, S. Hong, T. Wu, Y. Pan, H. Ren, M. D. Renzo, A. L. Swindlehurst, R. Zhang, and A. Y. Zhang, “An overview of signal processing techniques for RIS/IRS-aided wireless systems,” *IEEE Journal of Selected Topics in Signal Processing*, vol. 16, no. 5, pp. 883–917, Aug. 2022. DOI: 10.1109/JSTSP.2022.3187371.
12. A. L. Swindlehurst, R. Zhang, and A. Y. Zhang, “An overview of signal processing techniques for RIS/IRS-aided wireless systems,” *IEEE Journal on Selected Topics in Signal Processing*, vol. 16, pp. 883–917, Aug. 2022. DOI: 10.1109/JSTSP.2022.3187371.
13. B. Zheng, C. You, W. Mei, and R. Zhang, “A survey on channel estimation and practical passive beamforming design for intelligent reflecting surface aided wireless communications,” *IEEE Communications Surveys & Tutorials*, vol. 24, no. 2, pp. 1035–1071, 2022. DOI: 10.1109/COMST.2022.3149280.
14. C. Huang, G. C. Alexandropoulos, C. Yuen, and M. Debbah, “Indoor signal focusing with deep learning designed reconfigurable intelligent surfaces,” in *IEEE 20th International Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*, pp. 1–5, IEEE, 2019. DOI: 10.1109/SPAWC.2019.8815374.
15. J. Gao, C. Zhong, X. Chen, H. Lin, and Z. Zhang, “Unsupervised learning for passive beamforming,” *IEEE Communications Letters*, vol. 24, no. 5, pp. 1052–1056, 2020. DOI: 10.1109/LCOMM.2020.2978010.
16. T. L. Jensen and E. De Carvalho, “An optimal channel estimation scheme for intelligent reflecting surfaces based on a minimum variance unbiased estimator,” in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 5000–5004, IEEE, 2020. DOI: 10.1109/ICASSP40776.2020.9054617.
17. A. Taha, Y. Zhang, F. B. Mismar, and A. Alkhateeb, “Deep reinforcement learning for intelligent reflecting surfaces: Towards standalone operation,” in *Proc. IEEE 21st International Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*, pp. 1–5, IEEE, 2020. DOI: 10.1109/SPAWC48557.2020.9154320.
18. A. M. Elbir and K. V. Mishra, “A survey of deep learning architectures for intelligent reflecting surfaces,” *arXiv preprint arXiv:2009.02540*, 2020. DOI: 10.48550/arXiv.2009.02540.
19. A. Taha, Y. Zhang, F. B. Mismar, and A. Alkhateeb, “Deep reinforcement learning for intelligent reflecting surfaces: Towards standalone operation,” in *2020 IEEE 21st International Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*, pp. 1–5, May 2020. DOI: 10.1109/SPAWC48557.2020.9154320.
20. X. Yuan, Y.-J. A. Zhang, Y. Shi, W. Yan, and H. Liu, “Reconfigurable intelligent-surface empowered 6G wireless communications: Challenges and opportunities,” *arXiv preprint arXiv:2001.00364*, 2020. Accessed: Aug. 17, 2020. DOI: 10.48550/arXiv.2001.00364.
21. J. Park *et al.*, “Extreme URLLC: Vision, challenges, and key enablers,” *arXiv preprint arXiv:2001.09683*, 2020. Accessed: Aug. 17, 2020. DOI: 10.48550/arXiv.2001.09683.
22. Y. Zeng and X. Xu, “Towards environment-aware 6G communications via channel knowledge map,” *arXiv preprint arXiv:2007.09332*, 2020. Accessed: Jul. 31, 2020. DOI: 10.48550/arXiv.2007.09332.
23. M. D. Renzo *et al.*, “Smart radio environments empowered by reconfigurable AI meta-surfaces: An idea whose time has come,” *Journal of Wireless Communications and Networking*, vol. 2019, p. 129, Dec. 2019. DOI: 10.1186/s13638-019-1507-6.
24. C. Huang, R. Mo, C. Yuen, and S. Debbah, “Reconfigurable intelligent surface assisted multiuser MISO systems exploiting deep reinforcement learning,” *arXiv preprint arXiv:2002.10072*, 2020. Accessed: May 08, 2020. DOI: 10.48550/arXiv.2002.10072.
25. L. Subrt and P. Pechac, “Intelligent walls as autonomous parts of smart indoor environments,” *IET Communications*, vol. 6, pp. 1004–1010, May 2012. DOI: 10.1049/iet-com.2011.0146.
26. G. Lee, M. Jung, A. T. Z. Kasgari, W. Saad, and M. Bennis, “Deep reinforcement learning for energy-efficient networking with reconfigurable intelligent surfaces,” in *2020 IEEE International Conference on Communications (ICC)*, pp. 1–6, Jun. 2020. DOI: 10.1109/ICC40277.2020.9149063.
27. A. M. Elbir, A. Papazafeiropoulos, P. Kourtessis, and S. Chatzinotas, “Deep channel learning for large intelligent surfaces aided mmWave massive MIMO systems,” *IEEE Wireless Communications Letters*, vol. 9, pp. 1447–1451, 2020. DOI: 10.1109/LWC.2020.3013165.
28. A. Taha, M. Alrabeiah, and A. Alkhateeb, “Enabling large intelligent surfaces with compressive sensing and deep learning,” *IEEE Access*, vol. 9, pp. 44304–44321, 2021. DOI: 10.1109/ACCESS.2021.3055524.
29. S. Khan, K. S. Khan, N. Haider, and S. Y. Shin, “Deep-learning-aided detection for reconfigurable intelligent surfaces,” *arXiv preprint arXiv:1910.09136*, 2019. DOI: 10.48550/arXiv.1910.09136.
30. S. Liu, Z. Gao, J. Zhang, M. Di Renzo, and M. S. Alouini, “Deep denoising neural network assisted compressive channel estimation for mmWave intelligent reflecting surfaces,” *IEEE Transactions on Vehicular Technology*, vol. 69, pp. 9223–9228, 2020. DOI: 10.1109/TVT.2020.2998523.
31. Y. Jin, J. Zhang, X. Zhang, H. Xiao, B. Ai, and D. W. K. Ng, “Channel estimation for semi-passive reconfigurable intelligent surfaces with enhanced deep residual networks,” *IEEE Transactions on Vehicular Technology*, vol. 70, pp. 11083–11088, 2021. DOI: 10.1109/TVT.2021.3094040.
32. N. K. Kundu and M. R. McKay, “Channel estimation for reconfigurable intelligent surface aided MISO communications: From LMMSE to deep learning solutions,” *IEEE Open Journal of the Communications Society*, vol. 2, pp. 471–487, 2021. DOI: 10.1109/OJCOMS.2021.3066017.
33. L. Dai and X. Wei, “Distributed machine learning based downlink channel estimation for RIS assisted wireless communications,” *IEEE Transactions on Communications*, vol. 70, pp. 4900–4909, 2022. DOI: 10.1109/TCOMM.2022.3166340.
34. J. He, H. Wymeersch, M. Di Renzo, and M. Juntti, “Learning to estimate RIS-aided mmWave channels,” *IEEE Wireless Communications Letters*, vol. 11, pp. 841–845, 2022. DOI: 10.1109/LWC.2022.3150810.
35. M. Wu, Z. Gao, Y. Huang, Z. Xiao, D. W. K. Ng, and Z. Zhang, “Deep learning-based rate-splitting multiple access

for reconfigurable intelligent surface-aided tera-hertz massive MIMO," *IEEE Journal on Selected Areas in Communications*, vol. 41, pp. 1431–1451, 2023. DOI: 10.1109/JSAC.2023.3242117.
36. Y. Liu *et al.*, "Reconfigurable Intelligent Surfaces: Principles and Opportunities," in *IEEE Communications Surveys & Tutorials*, vol. 23, no. 3, pp. 1546–1577, thirdquarter 2021. doi: 10.1109/COMST.2021.3077737.
37. S. K. Das, F. Benkhelifa, Y. Sun, H. Abumarshoud, Q. H. Abbasi, M. A. Imran, and L. Mohjazi, "Comprehensive review on ML-based RIS-enhanced IoT systems: basics, research progress and future challenges," *Computer Networks*, vol. 224, p. 109581, 2023. https://doi.org/10.1016/j.comnet.2023.109581.
38. M. Gao, J. Yang, H. Li, and Y. Wang, "Robust Beamforming Optimization Design for RIS-Aided MIMO Systems With Practical Phase Shift Model and Imperfect CSI," in *IEEE Internet of Things Journal*, vol. 11, no. 1, pp. 958–973, 1 Jan.1, 2024. doi: 10.1109/JIOT.2023.3288137.
39. E. Basar, M. Di Renzo, J. De Rosny, M. Debbah, M.-S. Alouini, and R. Zhang, "Wireless Communications Through Reconfigurable Intelligent Surfaces," in *IEEE Access*, vol. 7, pp. 116753–116773, 2019. doi: 10.1109/ACCESS.2019.2935192.
40. C. Pan *et al.*, "An Overview of Signal Processing Techniques for RIS/IRS-Aided Wireless Systems," in *IEEE Journal of Selected Topics in Signal Processing*, vol. 16, no. 5, pp. 883–917, Aug. 2022. doi: 10.1109/JSTSP.2022.3195671.
41. Agbotiname Imoize, Augustus Ibhaze, Aderemi Atayero, and K. V. N. Kavitha, "Standard Propagation Channel Models for MIMO Communication Systems," *Wireless Communications and Mobile Computing*, 36, 2021. https://doi.org/10.1155/2021/8838792.
42. Wireless EM Propagation Software | Wireless InSite®| Remcom. (2024, December 29). Retrieved from https://www.remcom.com/wireless-insite-propagation-software.
43. A. Alkhateeb, S. Alex, P. Varkey, Y. Li, Q. Qu, and D. Tujkovic, "Deep learning coordinated beamforming for highly-mobile millimeter wave systems," *IEEE Access*, vol. 6, pp. 37 328–37 348, 2018. doi: https://doi.org/10.1109/ACCESS.2018.2850226.
44. W. Khawaja, O. Ozdemir, Y. Yapici, I. Guvenc, M. Ezuma, and Y. Kakishimay, "Indoor coverage enhancement for mmwave systems with passive reflectors: Measurements and ray tracing simulations," *arXiv preprint arXiv:1808.06223*, 2018. https://doi.org/10.48550/arXiv.1808.06223.