scientific reports

Check for updates

OPEN

Deep learning approach for hybrid beamforming design in MU-MISO mmWave systems

Ebrahim Ghaith[1], Tamer Mekkawy[1]✉, Ahmed A. Abouelfadl[2] & Ashraf Mahran[3]

Hybrid beamforming is a promising approach to alleviate hardware complexity in multi-user multiple-input single-output (MU-MISO) systems while maintaining high data rate performance. Unfortunately, hybrid beamforming architecture design is a challenging non-convex optimization problem due to stringent hardware constraints. However, traditional hybrid beamforming design methods, such as alternating minimization (AltMin) algorithms, rely on iterative optimization procedures that introduce heavy computational overhead and make them impractical for real-time applications. In this paper, we propose a deep learning (DL)-based hybrid beamforming method (DL-HBF) that aims to reduce computational latency while achieving acceptable sum-rate performance. Furthermore, we evaluate these methods based on a realistic channel model to ensure practical significance and their performance on imperfect channel state information (CSI). Additionally, we propose dataset generation procedures, which reduce the dataset creation and training overhead compared to existing DL-based hybrid beamforming methods that help in rapid deployment and scalability. Simulation results show that the proposed DL-HBF achieves an acceptable sum rate compared to traditional methods while reducing the computational complexity and maintaining robustness against channel estimation errors, which provides a practical solution for real-time hybrid beamforming for next-generation wireless systems.

Keywords Hybrid beamforming, MmWave, Deep learning, DeepMIMO, MU-MISO

Vehicular communications aim to improve safety and traffic efficiency, as well as real-time data collection and sharing, which demands high data rates and low latency. These demands motivate the use of millimeter waves (mmWave) for the next generations of wireless systems[1]. The mmWave communication technology represents a new approach to solve spectrum congestion problems because it possesses a wide frequency range between 30 and 300 GHz. Despite this potential, its signal suffers from high path loss and atmospheric absorption[2]. However, the short wavelength of mmWave makes it suitable for deploying a large number of antennas in a small dimension and opens the possibility of beamforming techniques to mitigate path loss and increase spectral efficiency[3].

In practical mmWave systems, the base station (BS) is equipped with a large antenna array, while user terminals often employ a single antenna due to hardware complexity and power consumption. Therefore, it leads naturally to a multi-user multiple-input single-output (MU-MISO) architecture. Compared to single-user systems, MU-MISO mmWave systems have additional challenges related to inter-user interference management, simultaneously serving multiple single-antenna users via spatial beamforming, and beamforming design under stringent hardware constraints[4]. Consequently, the development of efficient beamforming strategies is essential for MU-MISO to achieve a high sum rate while maintaining low computational and hardware complexity.

The concept of analog beamforming was introduced in[5,6], where analog phase shifters were used to set signal phases between individual antenna elements. However, this system architecture, which has a single RF chain, lacks the capability to support spatial multiplexing and is unsuitable for transmitting parallel data streams or simultaneously serving multiple users. Nonetheless, it has less hardware complexity and less power consumption. On the other hand, digital beamforming provides full control over magnitude and phase values within the baseband domain. Digital beamforming requires individual RF chains and both digital-to-analog (DACs) and analog-to-digital (ADCs) for every antenna element. Despite its flexibility and providing parallel data streams for multiple users, the use of separate RF chains, DACs, and ADCs for each antenna element leads to substantial hardware costs and high power consumption, making the approach unsuitable for large-scale

[1]Avionics Department, Military Technical College, Cairo 11511, Egypt. [2]Electrical Engineering Department, Military Technical College, Cairo 11511, Egypt. [3]Electronics and Communications Department, Egypt University of Informatics, Cairo 11511, Egypt. ✉email: mekkawy@ieee.org

antenna systems[7]. Therefore, a demand for hybrid beamforming has appeared, which represents an energy-efficient and cost-effective structure that integrates the advantages of analog and digital beamforming methods[8]. This balanced architecture provides sufficient flexibility for advanced multiplexing and multi-user strategies, while reducing hardware complexity and power consumption[9].

In addition, it provides a higher sum rate than analog beamforming and approaches the performance of digital beamforming[10]. The design of hybrid beamforming requires decomposing the fully digital precoder into analog and digital precoders under strict hardware constraints such as unit-modulus phase shifters and coupling between analog and digital precoders in the signal-to-interference-plus-noise ratio (SINR) expression, resulting in a bilinear structure that leads to a non-convex optimization problem. Existing approaches found optimal hybrid beamforming solutions based on iterative algorithms, which solve the non-convex problem, such as the manifold optimization alternating minimization (MO-AltMin) algorithm, the semidefinite relaxation alternating minimization (SDR-AltMin) algorithm, and phase extraction alternating minimization (PE-AltMin)[11], which first two use a MATLAB optimization software package such as Manopt[12] and CVX[13]. For a fully connected structure, where all RF chains are connected to all antennas via phase shifters, the MO-AltMin algorithm alternately updates the digital beamforming using least squares and refines the analog beamforming through conjugate gradient descent on the manifold and repeats until convergence. This method achieves a near-optimal spectral efficiency, despite the high cost of complexity. On the other hand, the PE-AltMin algorithm reduced this complexity by enforcing the orthogonality constraint and updating the analog beamforming from simple phase extraction to an equivalent digital beamforming, which also significantly improves performance. For the partially connected structure, in which each RF chain is connected to a subset of antennas, the SDR-AltMin algorithm is developed, which exploits the block-diagonal structure of the analog beamforming and provides closed-form phase updates and optimal digital beamforming via convex optimization, and makes a significant improvement over only analog beamforming. However, these existing iterative approaches for hybrid beamforming face limitations of prohibitive computational complexity and excessive latency. Therefore, they cannot satisfy the demanding real-time needs of vehicular communications because they fail to address the requirements of latency-sensitive mission-critical communications[14].

Deep learning (DL) has emerged as a promising paradigm for communication system applications because it differs from other iterative methods in behavioral applications[15]. In addition, it has a significant role in dealing with difficult problems in the physical layer[16], including channel estimation[17], global navigation satellite system (GNSS) jamming detection[18,19], and beamforming[20–22]. However, solving complicated non-convex optimization problems that usually affect beamforming design is not the only achievement. On the other hand, DL models require a lot of offline training, and after training, they can make low-latency decisions during real-time operation owing to their powerful pattern recognition and regression capabilities. The main advantage of DL in beamforming is supposed to be apparent in 5G/6G cellular networks[23], where instead of just overcoming the severe computational bottleneck limitations found in traditional algorithms, it also achieves real-time requirements in applications such as autonomous vehicular communications.

The application of DL in hybrid beamforming was introduced in[20], which uses a convolutional neural network (CNN)-based beamforming design to map imperfect channel state information (CSI) matrices to the analog beamforming matrix trained by exhaustive labeled data from perfect CSI to obtain high spectral efficiency while reducing computational complexity for a single user. However, it cannot support multi-user beamforming where inter-user interference plays a major role and there is no enforcement to make sure of the unit-modulus constraint, which hinders the implementation of the analog beamforming matrix. Additionally,[21] proposed a DL method that jointly selects antennas and hybrid beamforming for a single user by two CNN models, in which the first model selects the subarrays of antennas that maximize the spectral efficiency, and the second model predicts the analog and digital beamforming matrices. It was trained offline with exhaustive label data to be robust against imperfect CSI. Moreover, in[22], a CNN-MIMO-based hybrid beamforming method is used in a multi-user mmWave massive MIMO system, where the model learns a mapping between the imperfect CSI and analog beamforming matrix in the transmitter and the receiver with perfect CSI label data. All of the aforementioned algorithms require a long training time, which complicates retraining when system configuration changes. Moreover, they are data-hungry in the training phase[22], which demands a large memory allocation for storage. Additionally, they didn't use realistic channel models, which may produce suboptimal beamforming matrices upon application in a practical scenario.

In this paper, the problem of designing hybrid beamforming in MU-MISO mmWave systems is addressed. Beyond this, scalability and durability are critical system-level considerations for large-scale or resource-constrained wireless deployments. Recent research has shown that abstraction-based modeling, such as geographical abstraction in mega-constellation networks[24] and connectivity analysis in wireless-powered sensor networks[25], can greatly improve scalability and robustness in large-scale and resource-constrained wireless networks at the system level. These perspectives strengthen the proposed DL-HBF framework's applicability for large or resource-constrained deployments, where system resilience and observation-space abstraction are critical. The key contributions are:

- Structured procedures are developed to generate the supervised learning dataset. The input data consists of the angle, real and imaginary components of the channel matrices, while the output labels correspond to the indices of the optimal analog beamforming matrices. This dataset is generated based on DeepMIMO with a detailed configuration of environmental parameters, BS and user antenna arrays, and the specification of the number of users in MU-MISO mmWave systems.
- A DL-based hybrid beamforming method (DL-HBF) is proposed to design hybrid beamforming matrices in MU-MISO mmWave systems. It employs a CNN model to classify the optimal analog beamforming matrix

directly from the input channel matrices. Subsequently, the digital beamforming matrix is calculated, enabling a complete hybrid beamforming design.

- A Comprehensive performance evaluation and robustness analysis of the proposed DL-HBF method and the traditional hybrid beamforming algorithms, such as MO-AltMin, PE-AltMin, SDR-AltMin, and orthogonal matching pursuit (OMP). This analysis employs a realistic ray-tracing-based mmWave channel model capturing practical propagation characteristics, demonstrating an acceptable sum rate while significantly reducing execution time and maintaining robustness under imperfect CSI—highlighting its real-world applicability and suitability for real-time deployments in future wireless systems.

Notation: Throughout this paper: A is a matrix, a is a vector, a is a scalar, and $\mathcal{A}$ is a tensor. $|\mathrm{A}|$ is the determinant of A, where A^{T}, A^{H}, A^{*} are its transpose, hermitian (conjugate transpose), and conjugate, respectively. I_K is the identity matrix of dimension K and $\mathcal{CN}(\mathrm{m}, \mathrm{R})$ is a complex Gaussian random vector with mean m and covariance R. For a matrix A, $[\mathrm{A}]_{i,j}$ denotes the (i, j)th entry. Furthermore, $[\mathcal{A}]_{i,j,k,l}$ refers to the (i, j, k, l)-th element of a tensor $\mathcal{A}$. $\mathbb{E}\{\cdot\}$ denotes the statistical expectation and $\|\cdot\|_{\mathrm{F}}$ is the Frobenius norm. The notation $(\cdot)^{\dagger}$ denotes the Moore-Penrose pseudo-inverse, while $\angle\{\cdot\}$ denotes the angle of a complex scalar/vector/matrix. The operator $\Re\{\cdot\}$ denotes the real part, while $\Im\{\cdot\}$ denotes the imaginary part of a complex quantity. The symbol $\odot$ is the Hadamard product, and $*$ is the convolution operation.

System and channel models

A narrowband MU-MISO mmWave downlink system with a fully connected hybrid beamforming architecture is considered, as shown in Fig. 1. The BS is equipped with N_{t} antennas and N_{RF} RF chains. Without loss of generality, we assume each user needs only a single stream. Because the total number of streams, N_{s}, equals the number of users, K, thus $K \leq N_{\mathrm{RF}} \leq N_{\mathrm{t}}$, and this affects the maximum number of users, which are limited by the number of RF chains. In the hybrid downlink beamforming architecture, the BS processes the data streams in baseband using a digital precoder, $\mathrm{V}_{\mathrm{BB}} = [\mathrm{v}_{\mathrm{BB}_1}, \ldots, \mathrm{v}_{\mathrm{BB}_K}] \in \mathbb{C}^{N_{\mathrm{RF}} \times K}$, such that $\mathrm{v}_{\mathrm{BB}_k}$ is associated as a digital precoder for the data stream, which is intended for each user to transmit the symbol vector $\mathrm{s} = [s_1, \ldots, s_K]^{\mathrm{H}} \in \mathbb{C}^{K}$. That is satisfying $\mathbb{E}\{\mathrm{s}\mathrm{s}^{\mathrm{H}}\} = \frac{P}{K}\mathrm{I}_K$, where P is the total average transmitted power, under the assumption of equal power allocation across different users' streams. Subsequently, the analog precoder $\mathrm{V}_{\mathrm{RF}} \in \mathbb{C}^{N_{\mathrm{t}} \times N_{\mathrm{RF}}}$, is implemented via analog phase shifters where $|\mathrm{V}_{\mathrm{RF}}(i, j)|^2 = 1, \quad \forall i, j$, which governs the direction of transmission across the N_{t} antenna elements. The transmitted signal x can be constructed by:

$$\mathrm{x} = \mathrm{V}_{\mathrm{RF}}\mathrm{V}_{\mathrm{BB}}\mathrm{s} = \sum_{k=1}^{K} \mathrm{V}_{\mathrm{RF}}\mathrm{v}_{\mathrm{BB}_k} s_k, \tag{1}$$

where $\mathrm{x} \in \mathbb{C}^{N_{\mathrm{t}}}$ and s_k is the transmitted symbol for the k^{th} user. Consequently, the received signal of the k^{th} user is y_k:

$$y_k = \underbrace{\mathrm{h}_k^{\mathrm{H}}\mathrm{V}_{\mathrm{RF}}\mathrm{v}_{\mathrm{BB}_k} s_k}_{\text{desired signals}} + \underbrace{\mathrm{h}_k^{\mathrm{H}} \sum_{\ell \neq k} \mathrm{V}_{\mathrm{RF}}\mathrm{v}_{BB_\ell} s_\ell}_{\text{interference signals}} + \underbrace{n_k}_{\text{noise}}, \tag{2}$$

where $\mathrm{h}_k \in \mathbb{C}^{N_{\mathrm{t}}}$ is the channel vector between BS and the k^{th} user and n_k is a complex number that denotes the complex additive white Gaussian noise (AWGN) with $n_k \sim \mathcal{CN}\left(0, \sigma^2\right)$.

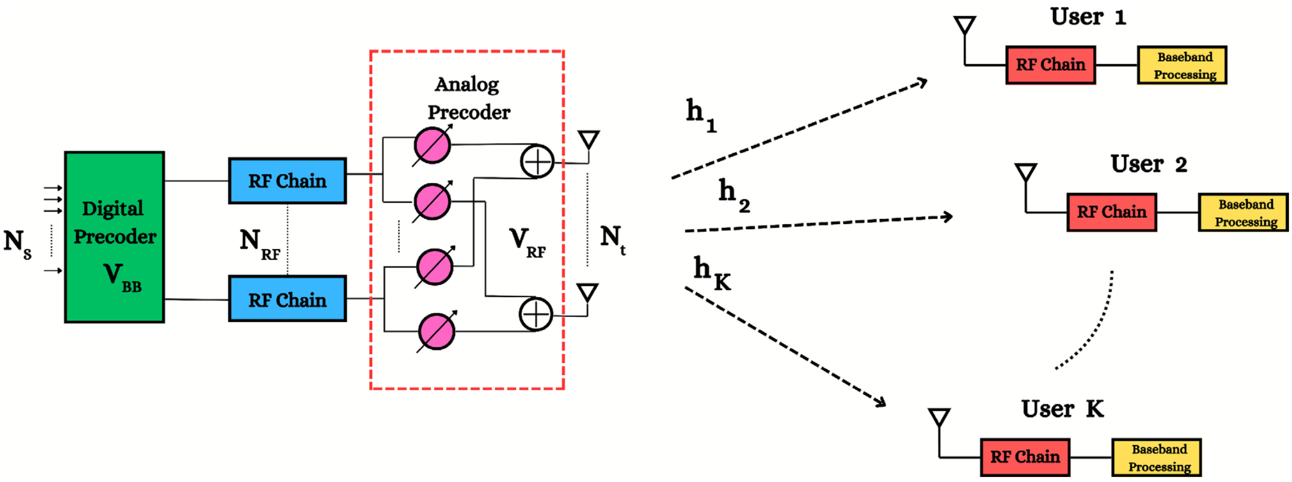


Fig. 1. A block diagram of MU-MISO system with hybrid beamforming architecture at BS and a single antenna K users.

Here, a flat-fading, narrow-band geometric mmWave channel model is considered with limited scattering. Therefore, we assume that the channel vector h_k has L scattering paths. Taking into account that we use a uniform linear array (ULA), therefore the channel vector can be expressed as[8]:

$$\mathrm{h}_k = \sqrt{\frac{N_t}{L_k}} \sum_{l=1}^{L_k} \alpha_{k,l} \mathrm{a}_t(\theta_{k,l}), \tag{3}$$

where L_k is the number of scattering paths for the k^{th} user, $\alpha_{k,l} \sim \mathcal{CN}(0,1)$ is the complex gain for the l^{th} path for the k^{th} user, $\theta_{k,l}$ is the angle of departure (AoD) of the l^{th} path for the k^{th} user and $\mathrm{a}_t(\theta_{k,l}) \in \mathbb{C}^{\mathrm{N_t}}$ is the antenna array response vector of the transmitter at the BS, which is given by:

$$\mathrm{a}_t(\theta_{k,l}) = \frac{1}{\sqrt{N_t}} \begin{bmatrix} 1 \\ e^{j\frac{2\pi}{\lambda} d \sin(\theta_{k,l})} \\ \vdots \\ e^{j\frac{2\pi}{\lambda} d(N_t-1)\sin(\theta_{k,l})} \end{bmatrix}, \tag{4}$$

where λ is the wavelength of the signal and d is the antenna spacing.

Problem formulation

For vehicular communications to serve high-throughput and latency-sensitive applications, real-time data interchange between automobiles and BS must be supported. As a result, the hybrid beamforming architecture of the MU-MISO system is required to optimize the sum rate, which can be defined as[9]:

$$\begin{aligned} \max_{\mathrm{V_{RF}}, \mathrm{V_{BB}}} \quad & R_{\text{sum}} = \sum_{k=1}^{K} \log_2(1 + \mathrm{SINR}_k), \\ \text{subject to} \quad & \|\mathrm{V_{RF}} \mathrm{V_{BB}}\|_F^2 = K, \\ & |[\mathrm{V_{RF}}]_{i,j}| = 1, \quad \forall i, j \end{aligned} \tag{5}$$

where the SINR for the k^{th} user is:

$$\mathrm{SINR}_k = \frac{\frac{P}{K} |\mathrm{h}_k^{\mathrm{H}} \mathrm{V_{RF}} \mathrm{v}_{\mathrm{BB}_k}|^2}{\frac{P}{K} \sum_{j \neq k} |\mathrm{h}_k^{\mathrm{H}} \mathrm{V_{RF}} \mathrm{v}_{\mathrm{BB}_j}|^2 + \sigma^2}. \tag{6}$$

In this manuscript, the main objective is to find the optimal matrix $\mathrm{V_{RF}}$ and $\mathrm{V_{BB}}$, which maximizes the sum rate. This problem poses some challenges: firstly, it is a nonconvex problem because of the coupled variables $\mathrm{V_{RF}}$ and $\mathrm{V_{BB}}$, which result in bilinear and unit modulus constraints. Secondly, interference-dominant environments for low SINR, and finally, dynamic conditions for users that need real-time adaptability.

The optimal analog and digital beamforming matrices—in terms of maximizing the sum rate – can be obtained using a variety of conventional methods, including manifold optimization, successive convex approximation, alternating optimization, and semidefinite relaxation[11]. Unfortunately, most of these methods are computationally complex and are unable to adapt in real time to changing conditions[11]. DL, therefore, has a promising potential for solving this issue.

Dataset generation

Obtaining the dataset is the first step in training DNNs. This dataset can be gathered for real deployments by the BS using real-time channel readings[26]. But in this instance, the system model is used to simulate the training dataset as mentioned in the previous section.

Channel generation

First, we use the DeepMIMO framework[27] to generate a suitable dataset for the MU-MISO mmWave system. This dataset is parameterized, which can be used in various configurations of wireless systems with the ability to adjust the channel parameters and represents environmental channels. It is generated with the aid of the Wireless InSite ray-tracing simulator and validated by channel measurements[28]. By specifying a particular simulation scenario, the realistic channel model is produced using the MU-MIMO system model. The active BS, the group of chosen users, the antenna arrangement at the BS and users, and the pertinent bandwidth are all important parameters of this simulation. Thus, each system and environmental characteristics are required to generate the appropriate channel impulse response and accurately mimic the required propagation environment.

Input data

The input channel matrix H for the K users is constructed as follows:

$$\mathrm{H} = [\mathrm{h}_1 \quad \mathrm{h}_2 \quad \cdots \quad \mathrm{h}_K]^{\mathrm{H}} \in \mathbb{C}^{K \times N_t}. \tag{7}$$

where h_k is the channel impulse response vector of the k^{th} user.

Three matrices are then obtained from H forming $\mathrm{X}_A, \mathrm{X}_R, \mathrm{X}_I$. The first matrix is $[\mathrm{X}_A]_{i,j} = \angle[\mathrm{H}]_{i,j}$, which is the angle of the channel matrix elements. The second and third matrices are the real and imaginary parts of H, i.e., $[\mathrm{X}_R]_{i,j} = \Re\{[\mathrm{H}]_{i,j}\}$ and $[\mathrm{X}_I]_{i,j} = \Im\{[\mathrm{H}]_{i,j}\}$. This representation enables the CNN model to accurately capture the spatial structure of the complex-valued channel impulse response. Therefore, the model gains direct access to the linear components of the channel by providing both the real and imaginary parts. In addition, the inclusion of phase information, formulated via the nonlinear arctangent relationship of the real and imaginary components, enhances the model's ability to capture the inherent nonlinearity of the channel structure[29]. Furthermore, through simulation results, which will be shown shortly in the result section , we noticed that this representation demonstrates a notable improvement in the model's accuracy.

Output data

According to Algorithm 1, to generate the training labels required for the proposed DL-HBF method, an offline exhaustive search explored all feasible analog beamforming matrices in a predefined codebook $\mathcal{V} = \{\mathrm{V}_{\mathrm{RF}}^{(1)}, \mathrm{V}_{\mathrm{RF}}^{(2)}, \ldots, \mathrm{V}_{\mathrm{RF}}^{(N_{cb})}\}$ where N_{cb} is the number of codewords that maximize the sum rate of our system model. The analog beamforming codebook $\mathcal{V}$ is constructed based on the discrete Fourier transform (DFT) matrix $\mathrm{U} \in \mathbb{C}^{N_t \times N_t}$[30], which is defined as:

$$[\mathrm{U}]_{s,c} = \frac{1}{\sqrt{N_t}} e^{-j2\pi(s-1)(c-1)/N_t}, \quad 1 \le s, c \le N_t, \tag{8}$$

where each column vector v of U is orthogonal and represents beams for analog beamforming matrix V_{RF}. The codebook is generated by selecting unique combinations of N_{RF} columns from U, yielding the analog beamforming matrix:

$$\mathrm{V}_{\mathrm{RF}}^{(b)} = \left[\mathrm{v}_1^{(b)}, \ldots, \mathrm{v}_{N_{\mathrm{RF}}}^{(b)}\right], \quad b = 1, \ldots, N_{\mathrm{cb}}. \tag{9}$$

Then it is normalized:

$$\mathrm{V}_{\mathrm{RF}}^{(b)} = \frac{\mathrm{V}_{\mathrm{RF}}^{(b)}}{\|\mathrm{V}_{\mathrm{RF}}^{(b)}\|_F}. \tag{10}$$

Given a channel matrix H, we select the analog beamforming matrix, V_{RF}, from the codebook that maximizes the sum rate and compute the optimal digital precoder V_{BB}. Assuming $\mathrm{H}_{\mathrm{eff}} = \mathrm{H}\mathrm{V}_{\mathrm{RF}}$ is the effective channel matrix and using minimum mean square error (MMSE)[31]:

$$\mathrm{V}_{\mathrm{BB}} = (\mathrm{H}_{\mathrm{eff}}^{\mathrm{H}}\mathrm{H}_{\mathrm{eff}} + \frac{1}{\mathrm{SNR}}\mathrm{I}_K)^{-1}\mathrm{H}_{\mathrm{eff}}, \tag{11}$$

where SNR is the signal-to-noise ratio. By normalizing $\mathrm{v}_{\mathrm{BB}_k}$, such that $\mathrm{v}_{\mathrm{BB}_k} = \frac{\mathrm{v}_{\mathrm{BB}_k}}{\|\mathrm{V}_{\mathrm{RF}}\mathrm{v}_{\mathrm{BB}_k}\|_F}$. To sum up, the output label data is calculated by specifying the index of the optimal analog beamforming matrix in the codebook, which leads to maximum sum rate due to Algorithm 1, and then digital beamforming is calculated by (11). Training and testing data are constructed from input data and output data mentioned previously to combine them as follows:

$$\mathcal{D}_{N_{\mathrm{i}}} = \left(\left(\mathcal{X}^{(1)}, z^{(1)}\right), \ldots, \left(\mathcal{X}^{(N_{\mathrm{i}})}, z^{(N_{\mathrm{i}})}\right)\right), \tag{12}$$

and

$$\mathcal{D}_{\mathrm{M}} = \left(\left(\mathcal{X}^{(1)}, z^{(1)}\right), \ldots, \left(\mathcal{X}^{(M)}, z^{(M)}\right)\right), \tag{13}$$

where $\mathcal{X}^{(N_{\mathrm{i}})} \in \mathbb{R}^{K \times N_t \times 3}$ is the input data for each N_{i} sample, $z^{(N_{\mathrm{i}})}$ is the index of the optimal analog beamforming matrix V_{RF} class for N_{i} sample, $\mathcal{D}_{N_{\mathrm{i}}}$ is the training dataset, N_{i} is the number of samples in the dataset which is divided into training and validation, $\mathcal{D}_{\mathrm{M}}$ is the testing dataset, and M is number of samples in the test dataset.

Based on the procedures illustrated in Fig. 2, the dataset can be generated as follows. **Firstly**, specifying the raytracing scenario that shows the BS's location, the users' positions geographically distributed in a certain outdoor environment, BS and users are selected, design the BS and users' antenna configurations (how many antennas, and antenna spacing and rotation) and system bandwidth then the DeepMIMO generator is used to generate channel parameters (angles of arrival/departure, path gains, $\cdots$ etc.) between the BS and users. **Secondly,** we specify how many users are assumed in our case of the MU-MISO system to group the new channel matrix as in (7). **Thirdly**, the output will have two paths, first path is to create the input data with $\angle[\mathrm{H}], \Re\{[\mathrm{H}\}, \Im\{[\mathrm{H}\}$ and the second path will be input to algorithm 1 to select the optimal analog beamforming matrix (codeword) that maximizes the sum rate of the MU-MISO system according to Algorithm 1. **Finally**, the input and output data will create the dataset needed for training the proposed DL-HBF method.

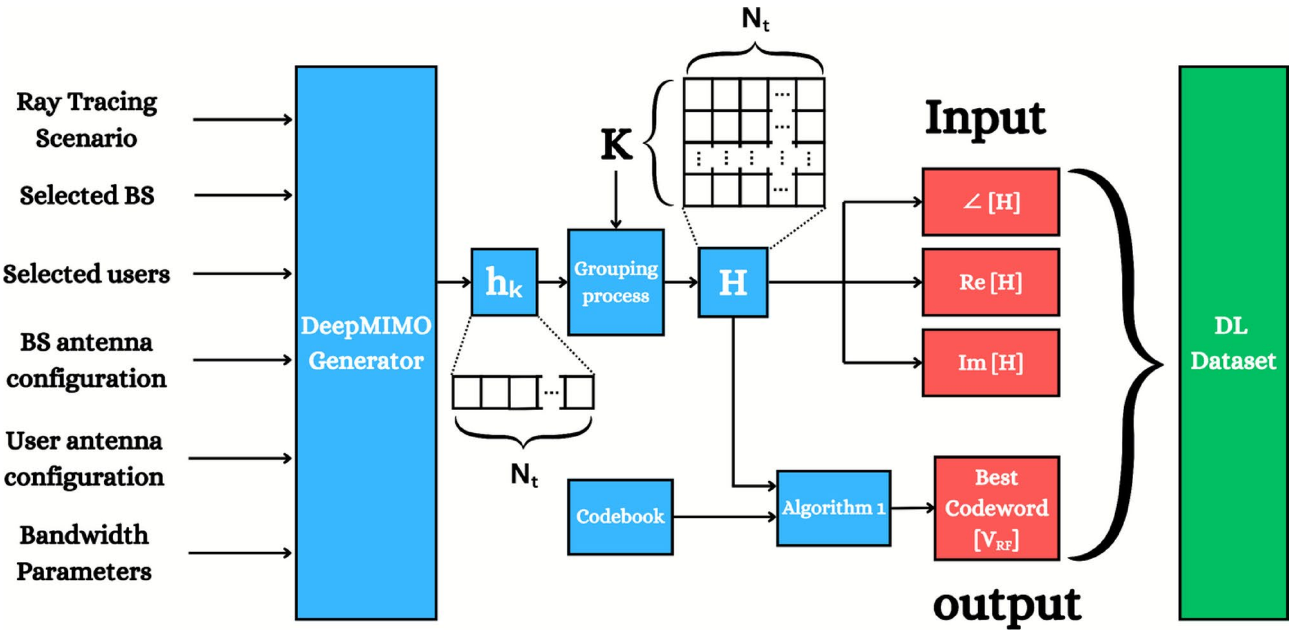


Fig. 2. A block diagram of the procedures of dataset generation.

```
Input: $\mathbf{H}$, $\mathcal{V} = \{\mathbf{V}_{\text{RF}}^{(1)}, \mathbf{V}_{\text{RF}}^{(2)}, \ldots, \mathbf{V}_{\text{RF}}^{(N_{cb})}\}$
Output: $\hat{\mathbf{V}}_{\text{RF}}$
Initialize: max_rate $\leftarrow 0$
for $i = 1$ to $N_{cb}$ do
    $\mathbf{V}_{\text{RF}} \leftarrow \mathbf{V}_{\text{RF}}^{(i)}$
    $\mathbf{H}_{\text{eff}} = \mathbf{H}\mathbf{V}_{\text{RF}}$
    $\mathbf{V}_{\text{BB}} = (\mathbf{H}_{\text{eff}}^{\text{H}}\mathbf{H}_{\text{eff}} + \frac{1}{\text{SNR}}\mathbf{I}_K)^{-1}\mathbf{H}_{\text{eff}}$
    for $k = 1$ to $K$ do
        Normalize:
            $$\mathbf{v}_{\text{BB}_k} \leftarrow \frac{\mathbf{v}_{\text{BB}_k}}{\|\mathbf{V}_{\text{RF}}\mathbf{v}_{\text{BB}_k}\|_F}$$
    end for
    Compute total sum rate $R_i$ as in (5)
    if $R_i >$ max_rate then
        max_rate $\leftarrow R_i$
        $\hat{\mathbf{V}}_{\text{RF}} \leftarrow \mathbf{V}_{\text{RF}}$,
    end if
end for
return $\hat{\mathbf{V}}_{\text{RF}}$
```

Algorithm 1. Hybrid Precoding for MU-MISO

The proposed DL-HBF method

To address the limitations mentioned in the proplem formulation section, we propose a DL-HBF method to obtain the optimal analog and digital beamforming matrices that maximize the sum rate based on DL. The objective of the CNN model shown in Fig. 3 is to classify the input data $\mathcal{X}$ into the index of the optimal analog beamforming matrix V_{RF} selected from the codebook $\mathcal{V}$. This approach will be illustrated in the upcoming subsections.

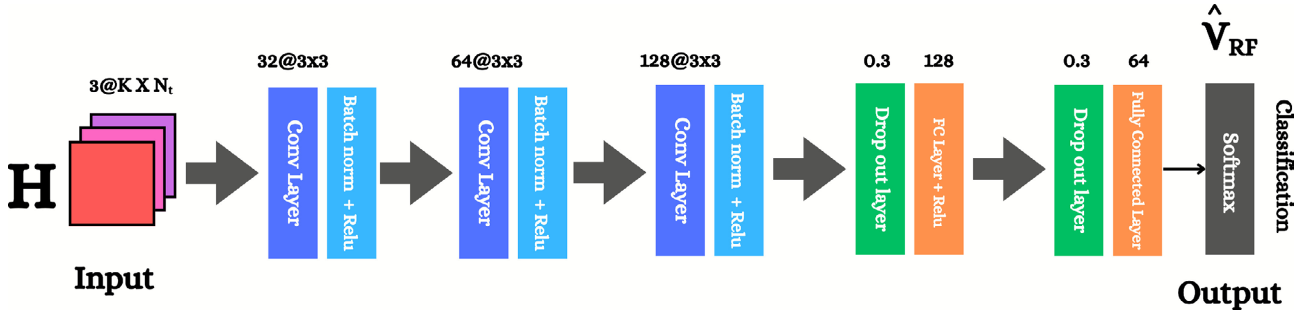


Fig. 3. A block diagram of the proposed DL model. The model depends on the input of (Angle, Real, Imag) channel tensor $\mathcal{X}$ to predict the class of optimal analog beamforming matrix.

Input representation and normalization

The input data $\mathcal{X}^{(N_{\mathrm{i}})}$, which has encapsulated features and characteristics of the channel between the BS and users is divided into $\angle[\mathrm{H}], \Re\{\mathrm{H}\}, \Im\{\mathrm{H}\}$ angle, real part and imaginary part of the channel matrix, respectively. To enhance numerical stability and speed up convergence, the Z-score normalization method will be used to normalize the input features along the sample axis, which enforces a zero mean and a variance of unity. Input elements $[\mathcal{X}^{(N_{\mathrm{i}})}]_{i,j,c}$ are normalized by Z-score:

$$\left[\mathcal{X}^{(N_{\mathrm{i}})}\right]^{\mathrm{norm}}_{i,j,c} = \frac{\left[\mathcal{X}^{(N_{\mathrm{i}})}\right]_{i,j,c} - \mu_{i,j,c}}{\sigma_{i,j,c} + \epsilon}, \tag{14a}$$

$$\mu_{i,j,c} = \frac{1}{N_i} \sum_{n=1}^{N_i} \left[\mathcal{X}^{(n)}\right]_{i,j,c} \tag{14b}$$

$$\sigma_{i,j,c} = \sqrt{\frac{1}{N_i} \sum_{n=1}^{N_i} \left(\left[\mathcal{X}^{(n)}\right]_{i,j,c} - \mu_{i,j,c}\right)^2} \tag{14c}$$

where $\mu_{i,j,c}$ is the mean of the input data, $\sigma_{i,j,c}$ is the standard deviation of the input data across the sample dimension N_{i} for each element (*i*, *j*, *c*) and ϵ is a small constant to avoid division by zero.

Model architecture

A convolutional neural network is designed in order to deal effectively with the spatial and temporal characteristics and features of the CSI. As shown in Fig. 3, the designed architecture contains three consecutive convolutional blocks, which perform hierarchical feature extraction and then connect to fully connected layers that reduce dimensionality before ending with a Softmax activation function (output layer) for multi-class classification.

- **Convolutional layer** The hierarchical feature extraction ability of CNNs learn very particular kinds of local spatial patterns and combines them to form increasingly abstract features that will help optimize beamforming matrices.

$$\left[\mathcal{Z}^{(k)}\right]_{i,j,n} = \sum_{c=1}^{D_{in}} \sum_{u=1}^{R} \sum_{v=1}^{R} \left[\mathcal{A}^{(k-1)}\right]_{i+u-1,j+v-1,c} \cdot \left[\mathcal{W}^{(k)}\right]_{u,v,c,n} + \mathrm{b}_n^{(\mathrm{k})}, \tag{15}$$

where $\mathcal{Z}^{(k)} \in \mathbb{R}^{K \times N_t \times D_{out}}$ is the output of the convolutional layer, D_{out} is the output depth, $\mathcal{W}^{(k)} \in \mathbb{R}^{R \times R \times D_{in} \times D_{out}}$ is the filter weights of k^{th} layer, each filter has a spatial dimention of $R \times R$ which is a learnable weight matrix that slides over the input data and performs the convolutional operation to extract features and spatial pattern, D_{in} is the input depth, $\mathcal{A}^{(k-1)} \in \mathbb{R}^{K \times N_t \times D_{in}}$ is the output of previous activation layer, $\mathcal{A}^{(0)} = \mathcal{X}^{\mathrm{norm}}$, besides, it is the input to the k^{th} layer and $\mathrm{b}_n^{(k)} \in \mathbb{R}^{D_{out}}$ is the bias term of k^{th} layer. There are three convolutional layers with increasingly larger feature maps: $\mathcal{W}^{(1)}$ has 32 filters with size 3×3, $\mathcal{W}^{(2)}$ has 64 filters of size 3×3, and $\mathcal{W}^{(3)}$ has 128 filters of size 3×3. The three layers have the same filter size but double the channel depth at each stage to enable hierarchical feature learning as in[32].

- **Batch normalization layer and activation function** It normalizes the activations of a layer in a mini-batch, stabilizes, accelerates the training process, and reduces internal covariate shift by allowing each layer to learn from inputs of a fixed distribution[33].

$$\left[\mathcal{Z}_B^{(k)}\right]^{\text{norm}}_{i,j,n} = \gamma_n^{(k)} \left(\frac{[\mathcal{Z}_B^{(k)}]_{i,j,n} - \mu_{\text{n}}^{(k)}}{\sqrt{(\sigma^2)_{\text{n}}^{(k)} + \epsilon}} \right) + \beta_n^{(k)}, \tag{16}$$

where $[\mathcal{Z}_B^{(k)}]^{\text{norm}}$ is the normalized output of the k^{th} batch normalization layer for mini-batch size per depth *n*, $\mathcal{Z}_B^{(k-1)}$ is the input to the batch normalization layer for mini-batch size of depth *n*, $\mu_n^{(k)}$, $(\sigma^2)_n^{(k)}$ are the mean and variance of the k^{th} batch normalization layer, which is computed per depth n over mini-batch size, respectively, and $\gamma_n^{(k)}$, $\beta_n^{(k)}$ are learnable scale and shift parameters for depth n in the k^{th} batch normalization layer. Rectified linear unit (ReLU) activation is used as an activation function, which adds non-linearity to the model so that it has the capability to learn complex mappings between input and output, and is less computationally expensive than tanh and sigmoid functions.

$$\left[\mathcal{A}_B^{(k)}\right]_{i,j,n} = \max\left(0, \left[\mathcal{Z}_B^{(k)}\right]^{\text{norm}}_{i,j,n}\right), \tag{17}$$

where $[\mathcal{Z}_B^{(k-1)}]^{\text{norm}}$ is the ReLU activation input at the k^{th} layer and $\mathcal{A}_B^{(k)}$ is the ReLU activation output in the k^{th} layer.

- **Fully connected layer** Upon feature extraction via convolutional layers, fully connected layers are then used for projecting the learned features to the output, and they help integrate information from different parts of the input and then make the final choice.

$$\text{u}^{(\text{k})} = \text{F}^{(\text{k})}\text{o}^{(\text{k-1})} + \text{d}^{(\text{k})}, \tag{18}$$

where $\text{u}^{(\text{k})} \in \mathbb{R}^Q$ is the output of the k^{th} fully connected layer, $\text{o}^{(k-1)} \in \mathbb{R}^G$ is the input for the fully connected layer which is flattened and *G* is total number of elements after flattening and dropout, $\text{d}^{(\text{k})} \in \mathbb{R}^Q$ is the bias term of k^{th} fully connected layer and $\text{F}^{(k)} \in \mathbb{R}^{Q \times G}$ is the associated weights of the k^{th} fully connected layer.

- **Dropout layer** To minimize overfitting, dropout regularization is applied during the training process when a fraction p of the neuronal units is randomly shut down with probability p at each training iteration, forcing the network to learn robust features. The model uses a dropout rate of p = 0.3[34].

$$\text{u}^{(k)}_{\text{dropout}} = \text{m}^{(k)} \odot \text{u}^{(k)}. \quad m_i \sim \text{Bernoulli}(1-p) \tag{19}$$

- **Softmax activation function (Output Layer)** It's the final step for multi-class classification, and Softmax converts the raw output scores (logits) into probabilities, which are defined as

$$P(u_i) = \frac{e^{u_i}}{\sum_{j=1}^{N_{cb}} e^{u_j}}, \quad i = 1, \ldots, N_{cb} \tag{20}$$

where u_i is the raw score for class *i*, N_{cb} is the number of classes, which equals 64 in our case.
Subsequently, the optimal analog beamforming index can be obtained by selecting the index corresponding to the highest predicted probability calculated by the Softmax activation function, which is given by:

$$\hat{u} = \arg\max_i P(u_i) \quad i = 1, \ldots, N_{cb}, \tag{21}$$

where $\hat{u}$ denotes the index of the predicted codeword $\hat{\text{V}}_{\text{RF}}$.

Training and loss function

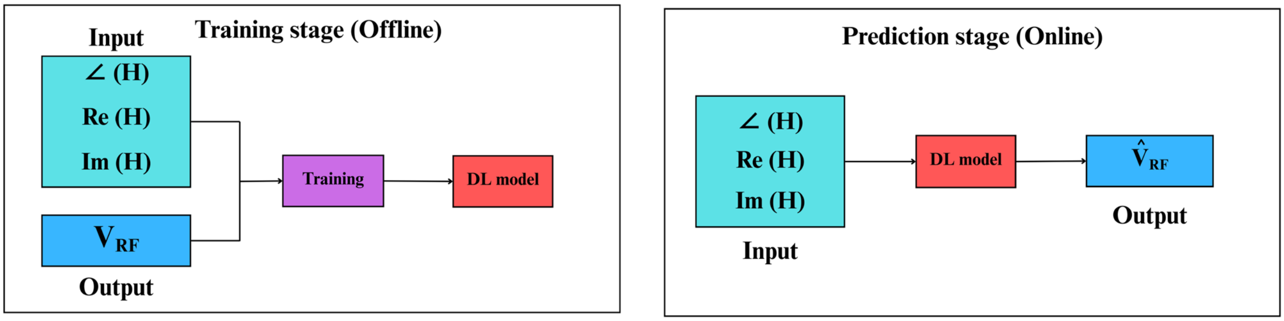
Cross-entropy loss $\mathcal{L}$, which has a probabilistic foundation and provides smooth gradients, making it suitable for multi-class classification tasks, is used to train the classification model. The objective is to minimize this loss by adjusting the weights of the network during the training phase as follows[35]:

$$\mathcal{L} = -\sum_{i=1}^{N_{cb}} \bar{u}_i \log P(u_i), \tag{22}$$

where $\bar{u}_i \in \{0, 1\}$ is the one-hot encoded ground-truth label for class *i* and $P(u_i)$ is the predicted probability for class *i*.

The adaptive moment estimation (Adam) optimizer is used to optimize the model parameters because of its efficient handling of sparse gradients, adaptability to different learning rates, and better convergence rate.

$$\theta_{t+1} = \theta_t - \eta \frac{\hat{m}_t}{\sqrt{\hat{v}_t} + \epsilon}, \tag{23}$$



(a)

(b)

Fig. 4. The training and prediction phases of the proposed DL-HBF model.

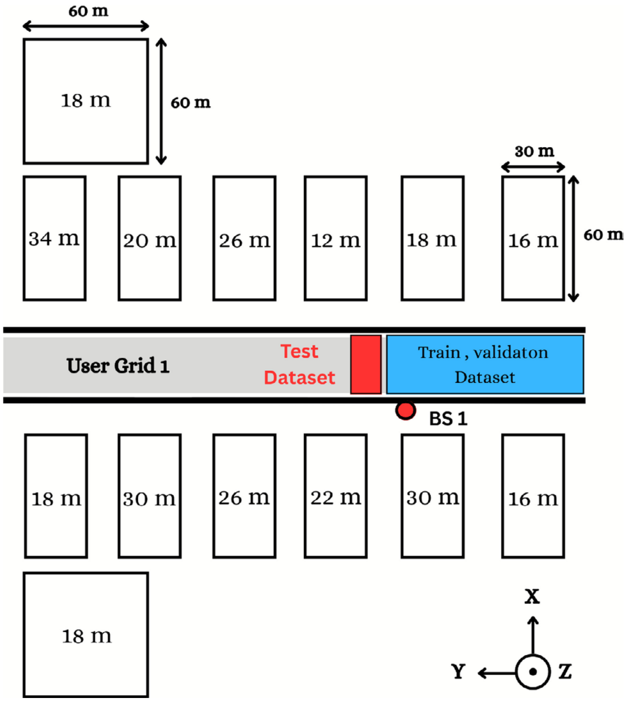


Fig. 5. The top view of environment setup in which BS serves users located in the user grid 1, highlighting their locations and regions for training and testing datasets.

where θ_t denotes an individual model parameter (weight or bias) in the time step t, η is the learning rate, $\hat{m}_t$ is the bias-corrected first moment estimate and $\hat{v}_t$ is the bias-corrected second moment estimate.

The CNN model has two stages as shown in Fig. 4. The first stage (training stage), where input channel tensor including $\angle[\mathrm{H}]$, $\Re\{[\mathrm{H}\}$, $\Im\{[\mathrm{H}\}$ and optimal analog beamforming V_{RF} codewords are used to train the model. After training the CNN model, the second stage will appear (the classification or online stage), which will classify the optimal analog beamforming matrix V_{RF} then the digital beamforming matrix V_{BB} will be calculated as mentioned before in the output data subsection.

Results and discussion

In this section, the performance of the proposed DL-HBF method is evaluated and compared with other benchmark methods in the MU-MISO system, including fully digital MMSE, MO-AltMin, SDR-AltMin, and PE-AltMin algorithms. Here, an MU-MISO downlink transmission system is considered with a BS that has a fully connected hybrid beamforming architecture with transmitting antennas $N_{\mathrm{t}} = 8$, RF chains $N_{\mathrm{RF}} = 4$, $N_{\mathrm{s}} = 4$, and every user has a single stream of data. The BS serves $K = 4$ users with a single antenna. We implement the proposed DL-HBF algorithms using MATLAB R2024a on a computer with an E5-1650 V3 CPU, 32 GB RAM, and an Nvidia Quadro K2200 4 GB GPU.

Environment setup

As shown in Fig. 5, an mmWave system is adopted, where the BS simultaneously serves multiple users over 60 GHz. A DeepMIMO scenario (O1_60), which shows the environment geometry, is used where:

- **Site plan**: The grid for users is in a 300 m long and 40 m wide street, the heights of all buildings are indicated, the buildings along the street have bases with 30 m x 60 m dimensions and two other far buildings with bases 60 m x 60 m dimensions, and a 60 GHz 3-layer dielectric material for the buildings, as shown on Fig. 5. A 60 GHz single-layer dielectric for the ground proves the importance of ray-tracing parameters such as reflection and penetration coefficients that accurately describe the model of the mmWave system. Each channel can undergo a maximum of 4 reflections before reaching the receiver.
- **BS**: BS 1 is 6 m in height, it has a ULA with 8 antennas where each antenna is an isotropic antenna. They use 30 dBm transmitter power, half-wavelength antenna spacing, and the bandwidth is 50 MHz.
- **Users**: The user has a single isotropic antenna at a height of 2 m. Users are uniformly distributed in the user grid (UG1) in the main street with a length of 550 m and a width of 35 m, has 2751 rows, and each row has 181 users and 20 cm spacing between adjacent users.

Model evaluation and tuning

Here, we evaluate the CNN model by calculating the accuracy, precision, recall, and F1-score, respectively:

- Accuracy: which directly indicates how the model predicts the right index of classes.

$$\text{Accuracy} = \left(\frac{\text{Number of Correct Predictions}}{\text{Total Number of Predictions}}\right) \times 100 \tag{24}$$

- Precision: which provides the percentage of correct positive predictions among all positive predictions.

$$\text{Precision}_i = \frac{TP_i}{TP_i + FP_i} \tag{25}$$

- Recall (sensitivity): which provides the percentage of correct positive predictions among all actual positives.

$$\text{Recall}_i = \frac{TP_i}{TP_i + FN_i} \tag{26}$$

- F1-score: which combines precision and recall in one metric and avoids the model when it is very selective (high precision with low recall) or catches many positives but with wrong ones (high recall with low precision).

$$\text{F1-score}_i = 2 \times \frac{\text{Precision}_i \times \text{Recall}_i}{\text{Precision}_i + \text{Recall}_i} \tag{27}$$

where *i* denotes the class index, *TP* is true positive, *FP* is false positive and *FN* is false negative. After the parameters are selected as input to the DeepMIMO generator as mentioned before, an output channel is generated for every user, then a new channel matrix is created by combining 4 users' channels and Algorithm 1 is used to generate the dataset as shown in Fig. 2. Users from "active_user_first" = 1 to "active_user_last" = 560 were used for the first dataset with 10000 samples. The dataset is split into 80% for training and 20% for validation. The second dataset, which is dedicated to testing, uses users from "active_user_first" = 561 to "active_user_last" = 610 and takes 2000 samples.

Parameter	Set value
Mini-batch size	128
Initial learning rate	0.0001
Number of epochs	50
Optimizer	Adam
Dropout rate	0.3
Filter size	3×3
Zero padding	1
Number of convolutional layers	3
Loss function	Cross-Entropy
Batch normalization momentum	0.99
Batch normalization ϵ	10^{-3}

Table 1. Proposed DL_HBF hyper-parameters.

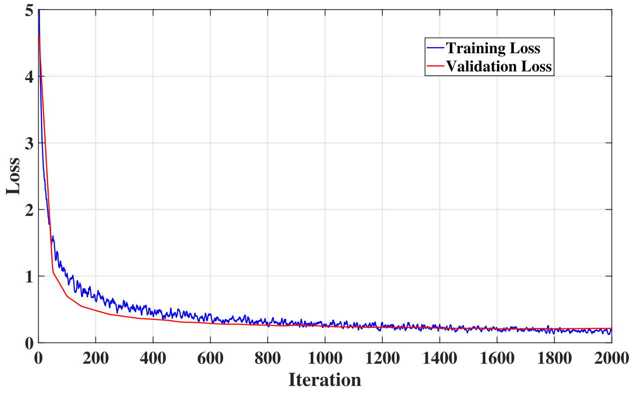


Fig. 6. The model's training and validation loss curves.

Metric	Validation dataset	Testing dataset
Accuracy (%)	93.2	91.6
Precision (%)	91.0	90.0
Recall (%)	79.0	71.0
F1 Score (%)	84.0	79.0

Table 2. Performance metrics.

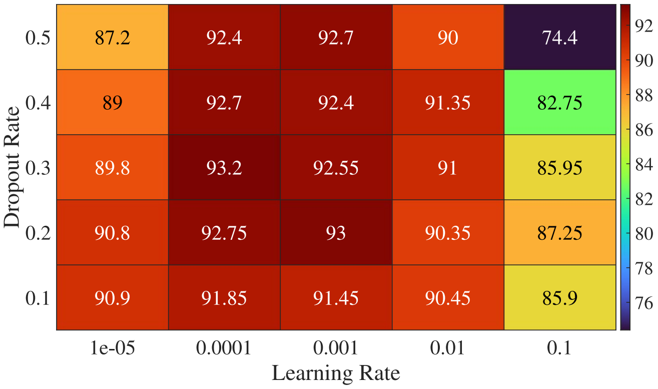


Fig. 7. The model's accuracy of different learning rates and dropout rates.

As shown in Table 1, the hyperparameters of our CNN model used in the proposed DL-HBF method are listed to obtain the optimal analog beamforming class.

Figure 6 illustrates the learning curves, which demonstrate the critical insights of the stability and convergence behavior of the proposed DL-HBF model. As observed from Fig. 6, the training loss decreases steadily, which indicates that the model is learning from the training data. Moreover, the validation loss also decreases and closely follows the training loss, which indicates the model is generalizing well to unseen channel realizations and not significantly overfitting. Consequently, both curves appear to converge to a low stable value, which is a strong indicator of success and a stable training process.

As shown in Table 2, the proposed DL-HBF model achieves a classification accuracy of 93.2% in validation and 91.6% in testing, demonstrating strong learning and generalization capability. Additionally, it has high precision, which indicates that the model reliably predicts the correct analog beamforming class and effectively minimizes false selections, which is essential to avoid power leakage toward unintended directions. Despite the low recall values, this reflects the model's conservative decision behavior, favoring precision to maintain stable performance. On the other hand, the F1-score confirms the trade-off between precision and recall.

These metrics, while standard in classification tasks, are also indirectly related to hybrid beamforming performance. Specifically, accurate classification leads to the selection of analog beamforming matrices whose corresponding digital beamforming matrices maximize the achievable sum-rate.

As shown in Fig. 7, which presents a comprehensive evaluation of how dropout rate and learning rate influence the accuracy of the proposed DL-HBF model. For the learning rate, the highest accuracy is obtained

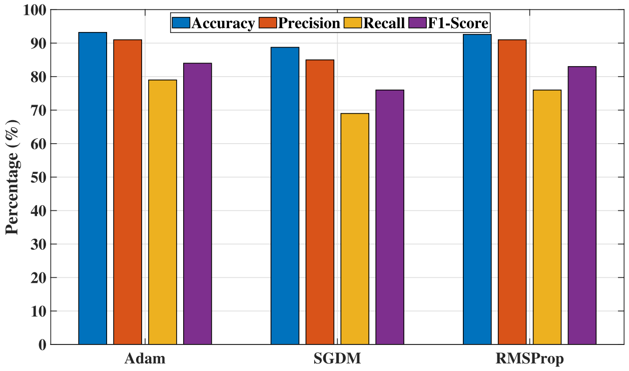


Fig. 8. The model's accuracy of different optimizers.

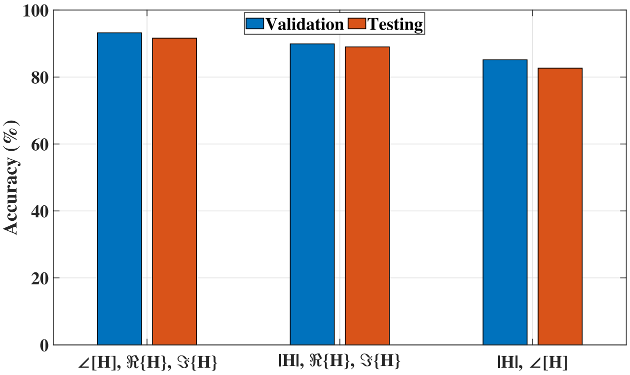


Fig. 9. The model's accuracy of different input representations.

at learning rates 0.0001 and 0.001, as these values achieve the appropriate balance between convergence speed and stability. Conversely, the performance is degraded at a learning rate of 0.1, which indicates potential instability or divergence during training. The dropout rates of 0.2 and 0.3 yield a relatively high accuracy, which indicates effective regularization and improved performance. This aligns with the regularization theory in DL to mitigate overfitting and encourage the model to learn robust features that generalize effectively to unseen data. Additionally, it shows that the optimal dropout rate depends on the learning rate, and these accuracy values support the importance of tuning hyperparameters rather than optimizing them in isolation.

Figure 8 shows the different optimization techniques such as Adam, stochastic gradient descent with momentum (SGDM) and root mean square propagation (RMSProp) which provide a full picture of classification effectiveness by metrics' evaluation such as *Accuracy*, *Precision*, *Recall* and $F1\_score$. Adam optimizer yields the best performance because it combines the advantages of the adaptive gradient algorithm (AdaGrad) and RMSProp, which uses adaptive learning rates and estimate the moment to speed up convergence and avoid local minima. On the other hand, the SGDM relies on the fixed learning rate and lacks adaptive gradient scaling, which results in the lowest performance. Furthermore, RMSProp provides intermediate performance that is better than SGDM due to its ability to adapt the learning rate and slightly lower than Adam, as it lacks the momentum terms. Overall, Adam optimizer proves most effective in capturing the nonlinear relationship of the input channel matrix and the optimal beamforming class, which highlights the importance of choosing the optimizer in DL models.

The input channel tensor $\mathcal{X}$ is divided into different matrices, which will be the features of the model. As shown in Fig. 9, which illustrates that $\angle[\mathrm{H}], \Re\{\mathrm{H}\}, \Im\{\mathrm{H}\}$ offers better accuracy than the other representations due to the $|\mathrm{H}|, \angle[\mathrm{H}]$ lack of directly providing real and imaginary, which reduced the ability of the model to capture complex spatial correlation in the input data. In addition, $|\mathrm{H}|, \Re\{\mathrm{H}\}, \Im\{\mathrm{H}\}$ have redundant information, which potentially confuses the network, despite $\angle[\mathrm{H}]$, which requires a nonlinear operation that is not easy for CNN to learn, and by providing it directly, it will be more effective.

The training time is a crucial aspect of the system's overall efficiency and complexity in offline mode. The relevant metric to measure that is the number of trainable parameters and an indication of the complexity of the model in offline mode. From Table 3, which illustrates that the proposed model has fewer parameters, resulting in a reduced training overhead and shorter training time. To ensure objective comparison for different models, the same system configuration is used: $K = 4$, $N_{\mathrm{RF}} = 4$, and $N_{\mathrm{t}} = 8$. Additionally, in[21], the dataset generation time was approximately 2 hours and 2 minutes dedicated to this task to generate 10000 samples with the publicly available code and the author's permission. However, our dataset generation required 53.7

Model	Number of parameters
Proposed DL-HBF	626,368
CNN-MIMO[22]	11,053,481
HBDL[20]	1,887,136
$\text{CNN}_{AS} + \text{CNN}_{RF}$[21]	951,226

Table 3. Parameters in DL-based hybrid beamforming models.

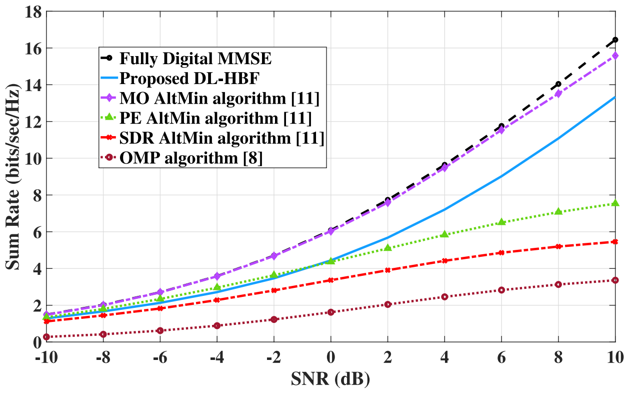


Fig. 10. The sum rate of the proposed DL-HBF in MU-MISO system at $N_t = 8,\ N_{RF} = K = 4$.

seconds to generate the same samples. This comparison was implemented under the same system model and hardware setup mentioned before to ensure fairness. In contrast, this reduction in time is attributed to different data preparation methodologies. Our dataset relies on precomputed ray-tracing results generated by Recom Wireless InSite, which retrieved and processed depending on specific configurations, plus Algorithm 1 to construct the output data, reducing computational overhead. On the other hand, [21] generates the dataset from scratch, including random data generation, numerous nested loops to construct the channel matrix, and the optimization algorithm to construct output data, considering delay and computational complexity. Therefore, the time efficiency of our method proves especially advantageous in the training phase, including scalability and retraining new configurations and scenarios.

System performance

To evaluate the proposed model's performance in terms of maximizing sum rate, it should be compared with traditional methods. Algorithms such as the MO-AltMin algorithm, the SDR-AltMin algorithm, and the PE-AltMin algorithm in[11] and the OMP algorithm in[8], we adopt these algorithms in the MU-MISO system to suit our multiple users case and apply them on the channel generated from DeepMIMO. These algorithms require a fully digital beamforming matrix as input to construct the analog and digital beamforming matrices, except the OMP algorithm, which additionally requires a predefined DFT codebook. The minimum mean square error (MMSE) is used as the optimal digital beamforming matrix needed for these traditional algorithms.

Figure 10 illustrates that the DL-HBF method performs better than the PE-AltMIN, SDR-AltMin and OMP algorithms but less than the MO-AltMin algorithm in identifying the optimal hybrid beamforming matrices that maximize the sum rate in the MU-MISO system. However, the performance gap of the MO-AltMin algorithm is primarily attributed to advanced optimization techniques, including computationally intensive procedures such as the conjugate gradient algorithm on a Riemannian manifold to get the nearest optimal solutions. In contrast, the PE-AltMIN algorithm is a simpler algorithm but with limited performance due to enforcing an orthogonality constraint on the digital beamforming matrix. Additionally, the SDR-AltMin algorithm adopts a partially connected structure, which inherently reduces beamforming gains. For the OMP algorithm yields the lowest performance due to its ignorance of the joint effect of N_{RF} columns which can't capture interactions between RF chains, and also the lack of iterative refinement.

As shown in Table 4, the computational complexity of the proposed DL-HBF method is analyzed by considering its two stages. The first stage, the prediction of the analog beamforming matrix via the CNN model, is dominated by the convolution layer, which has a complexity of $\mathcal{O}(KN_\text{t}R^2D_{in}D_{out})$. The second stage, the calculation of the digital beamforming matrix, involves several matrix operations, which is dominated by $\mathcal{O}(K^3)$. Therefore, the overall complexity of the proposed DL-HBF method is driven by these two stages, which is the forward pass that does not depend on I_L iterations. Conversely, the traditional hybrid beamforming algorithms such as MO-AltMIN, PE-AlTMin, and SDR-AltMin are iterative in nature, and their computational complexity scales linearly with the number of iterations I_L. This highlights the competitive advantage of the proposed DL-HBF method at computational complexity in systems where low-latency beamforming is required. For the execution time of the system mentioned before, the proposed DL-HBF method will execute the optimal

Method	Computational complexity
Proposed DL-HBF	$\mathcal{O}(KN_{\mathrm{t}}R^2D_{in}D_{out}+K^3)$
MO-AltMIN algorithm[11]	$\mathcal{O}(I_L(N_{\mathrm{t}}N_{\mathrm{RF}}^2+N_{\mathrm{t}}N_{\mathrm{RF}}K))$
PE-AlTMin algorithm[11]	$\mathcal{O}(I_L(KN_{\mathrm{t}}N_{\mathrm{RF}}+K^2N_{\mathrm{RF}})$
SDR-AltMin algorithm[11]	$\mathcal{O}(I_L(KN_{\mathrm{RF}})^6)$
OMP algorithm[8]	$\mathcal{O}(N_{\mathrm{RF}}N_{\mathrm{cb}}N_{\mathrm{t}}K)$

Table 4. The computational complexity of the proposed DL-HBF.

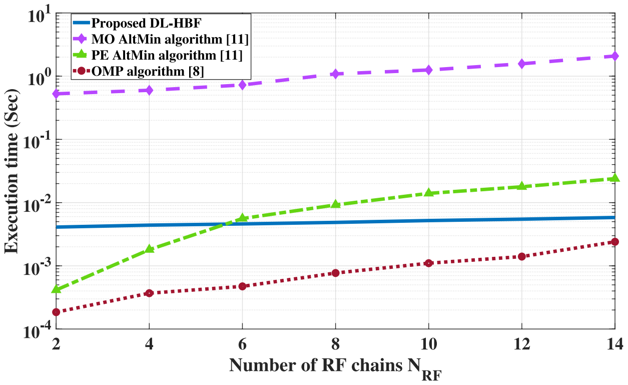


Fig. 11. The execution time of the proposed DL-HBF in MU-MISO system at $N_t = 16$.

analog beamforming matrix and then calculate the digital beamforming matrix in 0.0042 s, the MO-AltMIN algorithm in 0.4397 s, the PE-AlTMin algorithm in 0.0022 s, the SDR-AltMin algorithm in 3.18774 s, and the OMP algorithm in 0.00025 s. Therefore, the SDR-AltMin algorithm is out of the comparison.

As shown in Fig. 11, which presents the average execution time of the proposed DL-HBF method, the MO-AltMIN, the PE-AlTMin and the OMP algorithms for different values of N_{RF} and constant N_{t} = 16. Results show that the MO-AltMIN algorithm has more execution time than others due to its exhaustive iterations, and this time increases significantly as N_{RF} increases. However, the PE-AlTMin algorithm has less execution time than the proposed DL-HBF method in N_{RF} = 2 and 4, it will take more execution time than the proposed DL-HBF method by increasing N_{RF} and iterative updates. Additionally, the OMP algorithm achieves the lowest execution time due to its simplicity, but it increases proportionally to N_{RF}. This highlights the robustness and scalability of the proposed DL-HBF method, which maintains a nearly constant execution time as N_{RF} increases. These merits underscore the potential of the proposed method for real-time applications.

The availability of perfect CSI is unrealistic due to estimation and feedback errors. To analyze the robustness of algorithms for imperfect CSI cases and evaluate their performance in terms of normalized mean square error (NMSE). The estimated channel matrix $\hat{\mathrm{H}}$ can be expresed as:

$$\hat{\mathrm{H}} = \mathrm{H} + \mathrm{E}, \tag{28}$$

where $\mathrm{E} \sim \mathcal{CN}\left(0, \sigma_e^2\mathrm{I}\right)$ represents the channel estimation error matrix. Therefore, the degree of imperfection is characterised by:

$$\mathrm{NMSE}_{\mathrm{H}} = \frac{\|\hat{\mathrm{H}} - \mathrm{H}\|_F^2}{\|\mathrm{H}\|_F^2}. \tag{29}$$

To assess the approximation quality of hybrid beamforming matrices, NMSE is computed with respect to the optimal fully digital beamforming matrix by:

$$\mathrm{NMSE}_{\mathrm{V}} = \frac{\|\mathrm{V}_{\mathrm{RF}}\mathrm{V}_{\mathrm{BB}} - \mathrm{V}_{\mathrm{opt}}\|_F^2}{\|\mathrm{V}_{\mathrm{opt}}\|_F^2}, \tag{30}$$

where $\mathrm{V}_{\mathrm{opt}}$ is the optimal fully digital beamforming matrix. When the hybrid beamforming is designed based on imperfect CSI instead of perfect CSI, its performance deteriorates. This degradation represented by the difference between the NMSE values obtained under imperfect and perfect CSI:

$$\Delta_{\mathrm{NMSE}} = \mathrm{NMSE}_{\mathbf{V_i}} - \mathrm{NMSE}_{\mathbf{V_p}}, \tag{31}$$

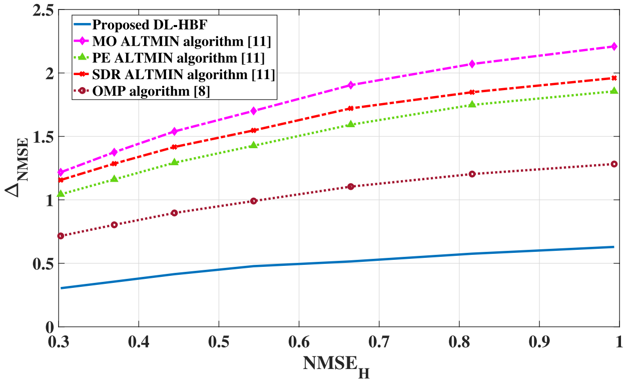


Fig. 12. The degradation performance of the proposed DL-HBF under imperfect CSI.

where $\text{NMSE}_{\mathbf{V}_\text{i}}$ calculated related to imperfect CSI and $\text{NMSE}_{\mathbf{V}_\text{P}}$ calculated related to perfect CSI. As shown in Fig. 12, all algorithms experience a degradation in performance as the CSI error increases. However, the extent of degradation varies significantly across algorithms. The proposed DL-HBF method consistently achieves the lowest Δ_{NMSE} across all NMSE_{H}, demonstrating superior robustness to channel estimation error due to its training on some noisy channels. On the other hand, other algorithms result in high degradation increases proportionally with estimated channel error. This confirms their reliance on perfect CSI and their limitations with noisy and imperfect CSI.

Conclusion

In this paper, we presented a proposed DL-HBF method to get the optimal beamforming matrices designed for MU-MISO systems to maximize the sum rate. Unlike traditional algorithms, such as MO-AltMin, SDR-AltMin, and PE-AltMin—which suffer from high computational complexity—and OMP, which achieves lower sum rate performance, all conventional algorithms exhibit degradation and limited adaptability to imperfect CSI, making them impractical for real-time implementation. In contrast, the proposed DL-HBF method achieves an acceptable sum rate with low latency and its superior robustness to channel estimation errors, which makes it suitable for practical deployment in real-time wireless systems. We validate these methods in a realistic channel model to ensure robustness and applicability. On the other hand, we make procedures to generate the dataset in a short time and reduce the training time compared to existing DL-based hybrid beamforming methods. Our results underscore the potential for DL in hybrid beamforming to provide next-generation wireless systems with high sum rates, low execution time and its strong capability in mapping imperfect CSI to the near-optimal hybrid beamforming matrices, where its application demands that. Future work may discuss extending this system to a complex system configuration, such as multi-cell and massive MIMO scenarios.

Data availability

All data generated or analysed during this study are included in this published article

Received: 5 November 2025; Accepted: 5 January 2026
Published online: 04 February 2026

References

1. Ananta, R. et al. Regression machine learning-based highly efficient dual band MIMO antenna design for mm-Wave 5G application and gain prediction. Scientific Reports **15**, https://doi.org/10.1038/s41598-025-13514-1 (2025).
2. Han, S., Chih-Lin, I., Xu, Z. & Rowell, C. Large-scale antenna systems with hybrid analog and digital beamforming for millimeter wave 5G. *IEEE Communications Magazine***53**, 186–194. https://doi.org/10.1109/MCOM.2015.7010533 (2015).
3. Gao, X., Dai, L., Han, S., I, C.-L. & Heath, R. W. Energy-Efficient Hybrid Analog and Digital Precoding for MmWave MIMO Systems With Large Antenna Arrays. IEEE Journal on Selected Areas in Communications **34**, 998–1009, https://doi.org/10.1109/JSAC.2016.2549418 (2016).
4. Li, H., Li, M. & Liu, Q. Hybrid Beamforming With Dynamic Subarrays and Low-Resolution PSs for mmWave MU-MISO Systems. *IEEE Transactions on Communications***68**, 602–614. https://doi.org/10.1109/TCOMM.2019.2950905 (2020).
5. Zhang, X., Molisch, A. & Kung, S.-Y. Variable-phase-shift-based RF-baseband codesign for MIMO antenna selection. *IEEE Transactions on Signal Processing***53**, 4091–4103. https://doi.org/10.1109/TSP.2005.857024 (2005).
6. Kutty, S. & Sen, D. Beamforming for Millimeter Wave Communications: An Inclusive Survey. *IEEE Communications Surveys and Tutorials***18**, 949–973. https://doi.org/10.1109/COMST.2015.2504600 (2016).
7. Roh, W. et al. Millimeter-wave beamforming as an enabling technology for 5G cellular communications: theoretical feasibility and prototype results. *IEEE Communications Magazine***52**, 106–113. https://doi.org/10.1109/MCOM.2014.6736750 (2014).
8. Ayach, O. E., Rajagopal, S., Abu-Surra, S., Pi, Z. & Heath, R. W. Spatially Sparse Precoding in Millimeter Wave MIMO Systems. *IEEE Transactions on Wireless Communications***13**, 1499–1513. https://doi.org/10.1109/TWC.2014.011714.130846 (2014).
9. Alkhateeb, A., Leus, G. & Heath, R. W. Limited Feedback Hybrid Precoding for Multi-User Millimeter Wave Systems. *IEEE Transactions on Wireless Communications***14**, 6481–6494. https://doi.org/10.1109/TWC.2015.2455980 (2015).
10. Alkhateeb, A., Heath, R. W. & Leus, G. Achievable rates of multi-user millimeter wave systems with hybrid precoding. In 2015 IEEE International Conference on Communication Workshop (ICCW), 1232–1237, https://doi.org/10.1109/ICCW.2015.7247346 (2015).

11. Yu, X., Shen, J.-C., Zhang, J. & Letaief, K. B. Alternating Minimization Algorithms for Hybrid Precoding in Millimeter Wave MIMO Systems. *IEEE Journal of Selected Topics in Signal Processing***10**, 485–500. https://doi.org/10.1109/JSTSP.2016.2523903 (2016).
12. Boumal, N., Mishra, B. & Absil, P. Manopt, a Matlab Toolbox for Optimization on Manifolds. Journal of Machine Learning Research **15** (2013).
13. CVX Research, I. CVX: Matlab Software for Disciplined Convex Programming, version 2.0. https://cvxr.com/cvx (2012).
14. Zrar Ghafoor, K. et al. Millimeter-Wave Communication for Internet of Vehicles: Status, Challenges, and Perspectives. IEEE Internet of Things Journal **7**, 8525–8546, https://doi.org/10.1109/JIOT.2020.2992449 (2020).
15. Wang, T. et al. Deep learning for wireless physical layer: Opportunities and challenges. *China Communications***14**, 92–111. https://doi.org/10.1109/CC.2017.8233654 (2017).
16. O'Shea, T. & Hoydis, J. An Introduction to Deep Learning for the Physical Layer. *IEEE Transactions on Cognitive Communications and Networking***3**, 563–575. https://doi.org/10.1109/TCCN.2017.2758370 (2017).
17. Wang, M., Wang, A., Liu, Z. & Chai, J. Deep learning based channel estimation method for mine ofdm system. Scientific Reports **13**, https://doi.org/10.1038/s41598-023-43971-5 (2023).
18. Reda, A., Mekkawy, T., Tsiftsis, T. A. & Mahran, A. Deep Learning Approach for GNSS Jamming Detection-Based PCA and Bayesian Optimization Feature Selection Algorithm. *IEEE Transactions on Aerospace and Electronic Systems***60**, 8349–8363. https://doi.org/10.1109/TAES.2024.3429049 (2024).
19. Reda, A. & Mekkawy, T. GNSS jamming detection using attention-based mutual information feature selection. Discover Applied Sciences **6**, https://doi.org/10.1007/s42452-024-05792-7 (2024).
20. Elbir, A. M. CNN-Based Precoder and Combiner Design in mmWave MIMO Systems. *IEEE Communications Letters***23**, 1240–1243. https://doi.org/10.1109/LCOMM.2019.2915977 (2019).
21. Elbir, A. M. & Mishra, K. V. Joint Antenna Selection and Hybrid Beamformer Design Using Unquantized and Quantized Deep Learning Networks. *IEEE Transactions on Wireless Communications***19**, 1677–1688. https://doi.org/10.1109/TWC.2019.2956146 (2020).
22. Elbir, A. M. & Papazafeiropoulos, A. K. Hybrid Precoding for Multiuser Millimeter Wave Massive MIMO Systems: A Deep Learning Approach. *IEEE Transactions on Vehicular Technology***69**, 552–563. https://doi.org/10.1109/TVT.2019.2951501 (2020).
23. Wang, J. & Lu, S. Joint beamforming and transmit power control for maximizing uplink communication rate of active IRS-assisted ISAC system. Scientific Reports **15**, https://doi.org/10.1038/s41598-025-16037-x (2025).
24. Ouyang, Q., Ye, N. & An, J. On the Vulnerability of Mega-Constellation Networks Under Geographical Failure. *IEEE Transactions on Networking***33**, 2049–2062. https://doi.org/10.1109/TON.2025.3551700 (2025).
25. Mekikis, P.-V., Antonopoulos, A., Kartsakli, E., Alonso, L. & Verikoukis, C. Connectivity Analysis in Wireless-Powered Sensor Networks with Battery-Less Devices. In 2016 IEEE Global Communications Conference (GLOBECOM), 1–6, https://doi.org/10.1109/GLOCOM.2016.7841682 (2016).
26. Wan, W., Chen, W., Wang, S., Li, G. Y. & Ai, B. Deep Plug-and-Play Prior for Multitask Channel Reconstruction in Massive MIMO Systems. *IEEE Transactions on Communications***72**, 4149–4162. https://doi.org/10.1109/TCOMM.2024.3369702 (2024).
27. Alkhateeb, A. DeepMIMO: A Generic Deep Learning Dataset for Millimeter Wave and Massive MIMO Applications. arXiv preprint https://doi.org/10.48550/arXiv.1902.06435 (2019).
28. Li, Q. et al. Validation of a Geometry-Based Statistical mmWave Channel Model Using Ray-Tracing Simulation. In 2015 IEEE 81st Vehicular Technology Conference (VTC Spring), 1–5, https://doi.org/10.1109/VTCSpring.2015.7146155 (2015).
29. Faragallah, O. S., El-Sayed, H. S. & El-Mashed, M. G. Performance Enhancement of MmWave MIMO Systems Using Deep Learning Framework. *IEEE Access***9**, 92460–92472. https://doi.org/10.1109/ACCESS.2021.3092709 (2021).
30. Huang, Y., Liu, C., Song, Y. & Yu, X. DFT codebook-based hybrid precoding for multiuser mmWave massive MIMO systems. *EURASIP Journal on Advances in Signal Processing***2020**, 11. https://doi.org/10.1186/s13634-020-00669-4 (2020).
31. Nguyen, D. H. N., Le, L. B. & Le-Ngoc, T. Hybrid MMSE precoding for mmWave multiuser MIMO systems. In 2016 IEEE International Conference on Communications (ICC), 1–6, https://doi.org/10.1109/ICC.2016.7510844 (2016).
32. He, K., Zhang, X., Ren, S. & Sun, J. Deep Residual Learning for Image Recognition. In 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 770–778, https://doi.org/10.1109/CVPR.2016.90 (2016).
33. Ioffe, S. & Szegedy, C. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. arXiv preprint https://doi.org/10.48550/arXiv.1502.03167 (2015).
34. Hinton, G. et al. Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research***15**, 1929–1958 (2014).
35. Gordon Rodriguez, E., Loaiza-Ganem, G., Pleiss, G. & Cunningham, J. Uses and Abuses of the Cross-Entropy Loss: Case Studies in Modern Deep Learning. arXiv preprint https://doi.org/10.48550/arXiv.2011.05231 (2020).

Author contributions

A.M. conceived the simulations, E.G. and T.K. conducted the simulations, E.G. and A.A. analysed the results, writing-reviewing and editing; E.G., T.K. and A.A. All authors reviewed the manuscript and have read and agreed to the published version of the manuscript.

Funding

Open access funding provided by The Science, Technology & Innovation Funding Authority (STDF) in cooperation with The Egyptian Knowledge Bank (EKB). This research received no external funding.

Declarations

Competing interests

The authors declare no competing interests.

Ethical approval

This article does not contain any studies involving animals performed by any authors. Also, this article does not contain any studies involving human participants performed by any of the authors.

Additional information

Correspondence and requests for materials should be addressed to T.M.

Reprints and permissions information is available at www.nature.com/reprints.

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.