Integrated Probing-beam Pattern Learning and Beam Prediction for mmWave Massive MIMO

Qiulin Xue, Alessandro Nordio, *Member, IEEE,* Kai Niu, *Member, IEEE,* Chao Dong, *Member, IEEE,*
Carla Fabiana Chiasserini, *Fellow, IEEE*

***Abstract*—With the widespread adoption of massive multiple-input multiple-output (MIMO) and millimeter wave (mmWave) communication techniques, the overhead of beam measurement and the complexity of beam management become even more severe issues due to the dramatic increase of the number of beams. Traditional methods often overlook the selection of optimal probing beams, thus limiting beam prediction performance. Additionally, existing solutions based on deep learning exhibit high complexity, which often hinders their practical deployment. In this work, we propose a lightweight, integrated neural network approach tailored for joint probing-beam pattern selection and beam prediction. Specifically, our solution includes two main components. First, formulating the selection of probing beams as a sampling operation, we envision a sampling network where, to enable gradient back-propagation of network parameters in spite the non-differentiable nature of sampling, the standard sampling function is approximated with a fitting function. Then, drawing inspiration from the physical structure of antenna arrays and 3D beam formation, we develop a beam-prediction network based on convolutional neural networks and self-attention mechanisms. Experimental results demonstrate that, thanks to the learned pattern, our proposed scheme achieves very good prediction performance (exceeding the state of the art by over 15% in top-1 accuracy), using a neural network with almost 90% less parameters than existing machine learning-based solutions.**

***Index Terms*—Millimeter-wave, massive MIMO, beam management, probing beams selection, deep learning**

I. INTRODUCTION

The proliferation of applications such as consumer and industrial Internet of Things (IoT) has introduced diverse and challenging demands on wireless communication systems, including low latency, high coverage, low energy consumption, and high data rates [1], [2]. Transmission technologies are therefore evolving to fulfill such requirements, with wireless communication standards adapting accordingly [3]. In particular, millimeter-wave (mmWave) massive multiple-input multiple-output (MIMO) technology and beamforming techniques have emerged as promising enablers for next-generation wireless communications, as they can harness abundant spectrum resources, thus enhancing data transmission rates. Furthermore, they allow for significant beamforming gains, which is crucial to compensate for the severe path loss characterizing mmWave propagation. However, to fully exploit these high beamforming gains, the beams generated by transmitting and receiving antennas (usually uniform planar or linear arrays (UPA/ULA)) must be precisely aligned. Importantly, addressing this challenge requires careful beam management (BM) as channel quality varies over time.

It is worth noting that BM is quite a complex process, as described in the current release of 5G, involving several procedures, namely, beam sweeping, beam measurement, beam reporting, beam determination, beam maintenance, and potential beam failure recovery [4], [5]. Such a complexity increases with the number of antennas, since the number of beams surges and beams become narrower. Indeed, this requires more frequent beam sweeping and measurements, and, especially when the channel is rapidly changing, it implies very high pilot overhead and significant, often unacceptable, delays [6].

To reduce beam sweeping complexity, hierarchical beam search approaches [6], [7] have been proposed. These methods use multi-tier beamforming codebooks at both the transmitter and receiver, starting with wider beams to iteratively refine the search space for the optimal narrow beam. While hierarchical beam searching reduces the number of beam sweeps to some extent, its accuracy is limited by the characteristics of the beamforming codebooks, particularly when errors occur in selecting the wider beams.

Recently, artificial intelligence (AI) or deep learning (DL)-enabled BM has emerged as a promising approach to solve this issue, thanks to its ability to extract nonlinear features inherent in the angular and time domains and to learn high-dimensional features of propagation scenarios [8]–[10]. Moreover, 3GPP has identified AI-enabled BM as a representative use case among the AI-based mechanisms for New Radio air interface [11].

DL-based BM solutions can be broadly categorized into context information-based solutions [12]–[18] and probing beam measurement-based solutions [11], [19]–[26]. Considering that the optimal beam direction is generally relative to the specific propagation environment, context information-based DL solutions aim to achieve precise beam alignment by leveraging high-dimensional environmental features from side information, such as locations, orientations, and images obtained from cameras. For instance, [12] formulates mmWave beam selection as a multi-class classification problem based on situational awareness, including receiver vehicle locations and types, and neighboring vehicles. Additionally, orientation

---

This work is supported by the National Natural Science Foundation of China (No. 62071058) and the China Scholarship Council for 1 year study at Politecnico di Torino (No. 202306470009). *(Corresponding author: Kai Niu.)*

Qiulin Xue, Kai Niu and Chao Dong are with the Key Laboratory of Universal Wireless Communication, Ministry of Education, Beijing University of Posts and Telecommunications, Beijing, China (e-mail: xue666@bupt.edu.cn; niukai@bupt.edu.cn; dongchao@bupt.edu.cn).

Alessandro Nordio is with CNR-IEIIT, Torino, Italy.

Carla Fabiana Chiasserini is with Politecnico di Torino, Torino, Italy and is also with CNR-IEIIT, Torino, Italy.

information of the receiver is used in deep neural network (DNN)-based beam selection methods in [14], [15]. Furthermore, a vision image-assisted BM solution is explored in [13], proposing a three dimension (3D) scene-based beam prediction framework. To address the requirement for extensive datasets to tune numerous trainable parameters, [16] exploits transfer learning techniques. A federated Lidar-aided beam selection method for vehicle-to-infrastructure mmWave communication is introduced in [17], where connected vehicles collaborate to train a shared DNN using locally available Lidar data. The work in [18], instead, employs a multi-armed bandit framework to develop online learning algorithms for beam pair selection, departing from traditional supervised learning approaches.

Probing beam measurement-based solutions, however, are typically preferred in existing communication systems, as they do not require prior knowledge. The key idea is to predict the optimal beam from the measurements (e.g., the reference signal receiving power (RSRP)) of a limited number of probing beams, leveraging on the spatial and temporal domain correlations between beams performance. Specifically, DL-based temporal beam prediction has been proposed in [19]–[21] for high-speed scenarios. In these works, historical probing beam measurements are used to predict optimal beam pairs in future time instances. Spatial beam prediction, instead, leverages probing beam measurements taken at the current time instance as input to a DNN. Examples of such techniques can be found in [22], [23], where wide beams are used as probing beams. The measurement of wide beams is regarded as a low-resolution image, and image super-resolution approaches can be used to predict the high-resolution measurement of narrow beams. In addition, several studies have attempted to use DL to design high-performance probing beam codebooks [27]. However, these methods involve a frequent update of the codebook, which requires significant system adaptations and, hence, is often unsuitable for practical applications. To address the above issues, studies such as [24]–[26] select partial narrow-beam pairs from the pre-defined codebook as probing beam pairs[1]. Specifically, different DNN-based schemes (such as BsNet [24], AMPBML [25] and visual geometry group based network (VGG-BMNet) [26]) have been proposed to predict the RSRP of all beam pairs or the optimal beam pair, based on the measurement of probing beam pairs.

Despite the extensive research on DL-based BM schemes and the significant progresses in reducing BM overhead, several challenges still need to be addressed. *Firstly*, solutions relying on context information require the knowledge of data (e.g., location, orientation, and vision images) that may not always be available. *Secondly*, in most of the research works based on measuring the performance of a probing beam set, the impact of the choice of the probing beam set on prediction performance has not been adequately addressed. Typically, spatially regular sampling or random sampling of the beam codebook are used, while the selection of the optimal probing beam pattern has not been sufficiently investigated. To the best of our knowledge, only one study [24] tackled this point by proposing the use of non-uniform sampling to extract Eigenbeams. However, this study focuses on the multiple-input single output (MISO) transmission scenario and employs just a simple ranking algorithm on beam energies for probing beams selection, without jointly optimizing the selection of probing beams and beam prediction performance. *Thirdly*, existing DNN models fail to fully exploit the spatial characteristics of beams, such as the features inherent in the UPA structure. This leads to a limited beam prediction performance. *Finally*, the issue of designing lightweight DNN models has not been adequately studied, with most of the existing models turning out to be unsuitable for practical sustainable deployment.

To solve the above challenges, we adopt a DL-based spatial BM approach, focusing, for simplicity, on a single-cell scenario. The main aim of our solution is to improve beam prediction performance by properly learning probing beam patterns, while striving for performance enhancement and lightweight design of the beam prediction mechanism. Specifically, our main contributions are as follows:

- We envision a DL-enabled scheme for joint probing beam pattern learning and beam prediction, named DeePP (Deep Pattern learning and beam Prediction). To this end, we develop an integrated neural network to optimize beam prediction performance and learn optimal probing beam patterns at the same time, which is achieved by designing a dual-purpose loss function. This mechanism can effectively adapt to various scenarios without requiring prior knowledge. Using the learned probing beam pattern, our approach minimizes the overhead and latency and enhances beam prediction performance.
- We design a sub-neural network, named sampling network, to facilitate the learning of the optimal probing beam pattern. The selection of probing beams is formulated as a sampling function, and the sampling network is trained to fit such a function. To overcome the issue of the non-differentiable standard sampling function, we incorporate a learnable sampling matrix and a modified Sigmoid function into the sampling network to approximate the standard sampling function and enable gradient back-propagation.
- We develop a lightweight and efficient beam prediction network. Inspired by the spatial characteristics of UPA and 3D beams formation, our work innovatively reshapes all the beam pairs into a multi-channel image, organizing BS-side measurements based on beam spatial positions, with receiver-side beams serving as channels. Convolutional layers and self-attention mechanisms are incorporated for extracting spatial beam correlation features and learning the optimal beam pair distribution.
- We assess the performance of DeePP in an indoor factory scenario. First, we construct a dataset by implementing an indoor factory channel model that is compliant with the 3GPP specification. Then we adopt various metrics

---

[1] Notably, in most of the existing systems and current 3GPP proposals, it is commonly assumed that probing beam pairs are directly selected from a predefined beam codebook. In our work, we thus align with this assumption and employ the term “probing beam pattern” to denote the indices of the selected beams or beam pairs. Importantly, we then formulate the problem of selecting beam pairs as a sampling operation.

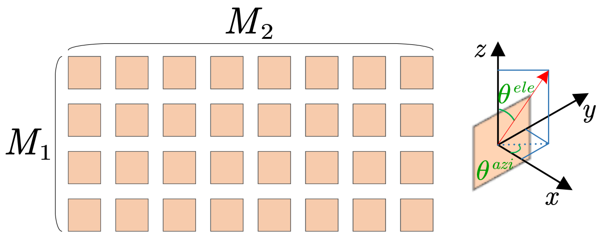


Fig. 1. Structure of a UPA panel and the associated local coordinate system.

and benchmarks for performance comparison. Simulation results show that, for a fixed number of probing beams, DeePP consistently outperforms our state-of-the-art benchmark using traditional probing beam pattern by more than 15% in top-1 accuracy. Importantly, such a gain in performance is obtained using a neural network with almost 90% less parameters than existing DL-based solutions.

The remainder of this paper is organized as follows. Section II describes the mmWave communication system model and the AI-based BM problem. Subsequently, Section III introduces the DL-enabled integrated probing beam pattern learning and beam prediction scheme. Section IV presents our performance evaluation; finally, Section V draws our conclusions.

II. SYSTEM MODEL AND PROBLEM FORMULATION

We consider a mmWave MIMO downlink communication system with a base station (BS) and multiple user equipments (UEs). We assume that both the BS and the UE are equipped with a UPA, where M_t and M_r antenna elements are used at BS and each UE, respectively.

A UPA panel, composed of $M_1 \times M_2$ elements, is depicted in Fig. 1. To each panel is associated a local coordinate system (LCS) whose x axis aligns with the broadside direction. The azimuth (i.e., horizontal) angle, $\theta^{\text{azi}} \in (-\pi, \pi)$, is measured counterclockwise from the x axis, whereas the zenith (i.e., vertical) angle, $\theta^{\text{ele}} \in (0, \pi)$, is measured from the z axis. Hence, points on the horizon have $\theta^{\text{ele}} = \pi/2$.

To generate a beam pointing to a direction specified by angles $(\theta^{\text{azi}}, \theta^{\text{ele}})$, the panel applies to its elements the following weight vector:

$$\begin{aligned}\mathbf{w}\left(\theta^{\text{azi}}, \theta^{\text{ele}}\right) = \tfrac{1}{\sqrt{M_1 M_2}} &\left[\mathrm{e}^{-\mathrm{j}2\pi d \sin\left(\theta^{\text{azi}}\right)\sin\left(\theta^{\text{ele}}\right)\mathbf{m}_2/\lambda}\right] \\ &\otimes \left[\mathrm{e}^{-\mathrm{j}2\pi d \cos\left(\theta^{\text{ele}}\right)\mathbf{m}_1/\lambda}\right],\end{aligned} \tag{1}$$

where θ^{azi} and θ^{ele} represent the beam angle along the horizontal and vertical directions, respectively. The vectors $\mathbf{m}_1 = [1, \ldots, M_1]^T$ and $\mathbf{m}_2 = [1, \ldots, M_2]^T$ denote the indices of the horizontal and vertical antenna elements in the UPA. The vectors $\mathrm{e}^{-\mathrm{j}2\pi d \sin\left(\theta^{\text{azi}}\right)\sin\left(\theta^{\text{ele}}\right)\mathbf{m}_2/\lambda}$ and $\mathrm{e}^{-\mathrm{j}2\pi d \cos\left(\theta^{\text{ele}}\right)\mathbf{m}_1/\lambda}$ are obtained by element-wise exponentiation of $\mathbf{m}_2$ and $\mathbf{m}_1$, respectively, where d and λ indicate the element spacing and signal wavelength, respectively. The symbol $\otimes$ denotes the Kronecker product.

Notice that (1) generically defines the beamforming vector in a panel. In the downlink system under consideration, we use superscripts (t) and (r) to denote the beamforming vectors, respectively, at the BS and at the UE. Both the BS and the UE can radiate beams in various spatial directions by adopting different combinations of azimuth (θ^{azi}) and elevation (θ^{ele}) angles in their respective beamforming weight vectors, $\mathbf{w}^{(t)}$ and $\mathbf{w}^{(r)}$.

As typically assumed and done in practical systems, we consider the beamforming codebooks to be predefined. Specifically, by choosing a set of azimuth angles and a set of elevation angles at both BS and UE, we can obtain the beamforming codebooks at the BS and the UE as: $\mathcal{C}_t = \left\{\mathbf{w}_1^{(t)}, \ldots, \mathbf{w}_{C_t}^{(t)}\right\}$ and $\mathcal{C}_r = \left\{\mathbf{w}_1^{(r)}, \ldots, \mathbf{w}_{C_r}^{(r)}\right\}$, of size of C_t and C_r, respectively. Notice that C_t consists of $N_t^v \times N_t^h$ beams and C_r consists of $N_r^v \times N_r^h$ beams, with N_t^v and N_r^v (N_t^h and N_r^h) denoting the number of beams along the vertical (horizontal) dimension.

Next, let us consider that the BS transmits the zer-mean unit energy pilot signal u with the j-th beams of its codebook and the UE receives it with the i-th beams of its codebook. Then, denoting the downlink channel matrix between the antenna elements of the BS and those of the UE with $\mathbf{H} \in \mathbb{C}^{M_r \times M_t}$, the received signal of beam pair (i, j) at the UE can be written as:

$$y_{i,j} = \sqrt{P_s}\mathbf{w}_i^{(r)H}\mathbf{H}\mathbf{w}_j^{(t)}u + n_{i,j}, \tag{2}$$

where $\mathbf{w}_i^{(r)} \in \mathbb{C}^{M_r}$ and $\mathbf{w}_j^{(t)} \in \mathbb{C}^{M_t}$ are the beamforming vectors at the UE and the BS, respectively, the superscript H denotes conjugate transpose, P_s indicates the transmit power, and $n_{i,j} \in \mathbb{C}$ is the addictive white Gaussian noise with zero mean and variance σ^2.

The RSRP of beam pair (i, j), computed by averaging over a sufficiently large number of pilot symbols, u, is given by $P_s|\mathbf{w}_i^{(r)H}\mathbf{H}\mathbf{w}_j^{(t))}|^2 + \sigma^2$. Intuitively, the best beam pair can be determined by exhaustive searching through $C_r C_t$ beam pairs. However, due to the large number of beams, such computation method results to be exceedingly complex. DNN-based solutions for the estimation of the optimal beam pair index have therefore emerged, which require measuring just a limited number of beam pairs. Notably, narrowing down the range of beam sweepings greatly reduces the beam training overhead.

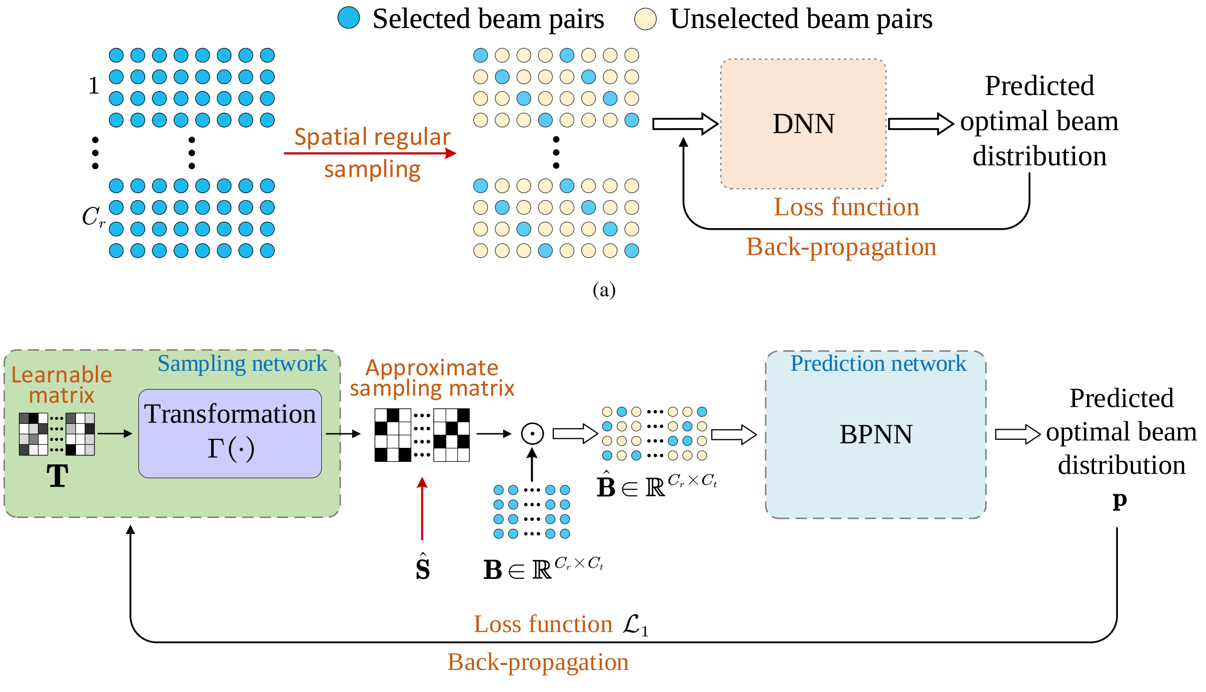
Let $\mathcal{M}$ denote the set of measured RSRP of the selected probing beam pairs. By indicating with $\text{BPNN}(\cdot)$ the beam prediction neural network (BPNN) function, and with $\mathbf{p} = \text{BPNN}(\mathcal{M})$ its output, the DNN-enabled spatial BM problem can be formulated as:

$$\ell = \arg\max_i \, p_i, \quad i \in \{1, 2, \ldots, C_r C_t\}, \tag{3}$$

where the output of the BPNN, $\mathbf{p} = [p_1, p_2, \ldots, p_{C_r C_t}]^T \in \mathbb{R}^{C_r C_t}$, is the estimated distribution vector of the optimal beam pair index and ℓ is the predicted index of the optimal beam pair.

III. DEEPP: A SPATIAL-DOMAIN BEAM PREDICTION

In this section, we introduce the motivation behind the proposed DeePP scheme and present its design and implementation.



(a)

(b)

Fig. 2. (a) Traditional DNN-based spatial beam prediction. (b) DeePP integrated neural network for joint probing-beam pairs selection and beam prediction.

A. Motivation and Framework Architecture

Let us consider the scheme depicted in Fig. 2(a), representing the approach used by the existing DNN-based beam prediction schemes. Such an approach involves offline training and online inference. For brevity, Fig. 2(a) only shows the offline-training operation: the gradient of a loss function is used to update the network parameters through back-propagation. In particular, this approach typically employs a predefined spatial pattern to select the probing beam pairs and then uses a DNN to estimate the optimal beam pair. According to most of the existing solutions, the predefined probing beam pattern commonly uses a spatial regular pattern. The input pattern depicted in Fig. 2(a) exemplifies this method, with blue (yellow) circles corresponding to selected (unselected) beam pairs. Circles are arranged in $N_t^v \times N_t^h$ rectangles, and there are a total of C_r rectangles, one for each receive beam. In this example, all C_r receive beams are selected, and $K_t{=}8$ transmit beams are sampled from a pool of $N_t^v \times N_t^h{=}4\times 8$ transmit beams in a systematic manner.

The spatial regular pattern typically adopted by the existing schemes is a practical, yet, in general, not optimal approach, as an optimal sampling pattern, e.g., the optimal indices of probing beam pairs, strongly depend on the characteristics of a specific scenario. In our study, we aim at addressing this issue, which, has been largely overlooked by prior work. We do so by leveraging deep learning, as the complex relationship between probing-beam pairs and operational scenarios prevents the use of analytical methods.

Similarly, our DeePP approach involves an offline-training stage and an online inference stage. During training, our primary objective is to learn the sampling pattern and train the BPNN, while in the inference stage, we use the trained BPNN for beam prediction. We assume that a dataset, $\mathcal{D}$, with each sample comprising the RSRP of all beam pairs, has been built in advance. The RSRP of all beam pairs and the corresponding index of the optimal beam pair are thus available in the training stage, while only the RSRP of the probing beam pairs can be acquired and used by the BPNN in the inference stage.

The offline training process of the proposed DeePP scheme consists of an end-to-end training of the integrated neural network and of a fine-tuning of the BPNN. The end-to-end training process, illustrated in Fig. 2(b), utilizes a versatile loss function, $\mathcal{L}_1$, for gradient computation and parameter updates. The number of probing beam pairs K is predetermined and incorporated into $\mathcal{L}_1$. Through the first-stage training, we obtain a preliminarily trained BPNN. Subsequently, fine-tuning of the BPNN is conducted using the loss function $\mathcal{L}_2$ (further details are provided in Section III-D).

Upon completing the two-stage offline-training, we acquire the learned probing beam pattern and the trained beam prediction network. As shown in Fig. 3, in the online inference stage, we deploy the BPNN at UE. After reference signals are transmitted by the BS and received by the UE through the learned probing beam pairs, the measured RSRP of these beam pairs serves as input to the deployed BPNN for predicting the index of the optimal beam pair. The pseudocode of the proposed DeePP scheme is presented in Algorithm 1.

The overall architecture of the proposed integrated neural network, including the sampling network and the BPNN, is shown in Fig. 2(b). As mentioned, the sampling network is trained to learn a probing beam pattern or, in other words, to fit a sampling function, while the BPNN is trained to learn the optimal beam pair from the sampled beam information to accomplish the beam prediction task.

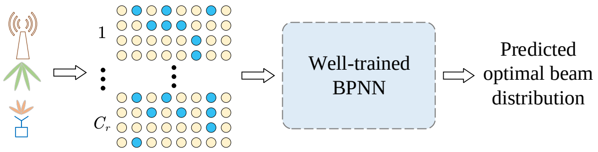


Fig. 3. DNN-based online spatial beam prediction using the learned pattern.

Algorithm 1 The DeePP algorithm

```
Input:
    The size of probing beam pairs, $K$
    The training dataset, $\mathcal{D}$
Output:
    Sampling matrix for probing beam pairs, $\mathbf{S}$
    Trained BPNN for online beam prediction
First training stage: Train the integrated neural network with an overall loss function $\mathcal{L}_1$
Obtain the sampling matrix $\mathbf{S}$ and the indices of probing beam pairs
Second training stage: Fine-tune the beam prediction network with loss function $\mathcal{L}_2$
Obtain the trained BPNN
Online inference stage: Use the above BPNN for optimal beam pair prediction from the measurement of probing beam pairs
```

B. Sampling Network

The selection of probing beam pairs can be regarded as a sampling operation. To each beam pair can be associated a binary variable, $s_{i,j}$, with $s_{i,j}{=}1$ indicating that the beam pair (i,j) is selected, and $s_{i,j}{=}0$ otherwise. These binary variables, for $i{=}1,\ldots,C_r$ and $j{=}1,\ldots,C_t$ can be arranged into the matrix $\mathbf{S}{\in}\mathbb{R}^{C_r\times C_t}$.

To facilitate the learning of the optimal probing beam pattern, we design the sampling network to fit the sampling function. To represent the sampling operation of beam pairs, our solution leverages the element-wise multiplication of the matrix $\mathbf{S}$ with the matrix $\mathbf{B}{\in}\mathbb{R}^{C_r\times C_t}$ containing the RSRP of all beam pairs. The sampling operation can be expressed as

$$\hat{\mathbf{B}} = \mathbf{S} \odot \mathbf{B}, \tag{4}$$

where $\hat{\mathbf{B}}{\in}\mathbb{R}^{C_r\times C_t}$ denotes the RSRP matrix after sampling, with non-zero elements indicating the selected beam pairs. The symbol $\odot$ denotes an element-wise multiplication.

The sampling network is expected to learn a sampling matrix $\mathbf{S}$ alongside the BPNN to ensure optimal beam prediction performance. Unfortunately, the binary constraint of each element of $\mathbf{S}$ makes the sampling function non-differentiable, posing challenges for learning the sampling matrix through gradient back-propagation. To circumvent this problem, we smooth-approximate the standard sampling function using a modified Sigmoid function $\mathcal{G}(\cdot)$:

$$\mathcal{G}(t) = \frac{\exp(t/\delta)}{1+\exp(t/\delta)}, \tag{5}$$

where t is a real-valued variable and δ is a real-valued constant. As depicted in Fig. 4, the smaller δ is, the closer the function resembles a step function, akin to the standard sampling function. This approach enables us to approximate the non-differentiable sampling function with a differentiable one, facilitating gradient back-propagation and the network parameters update.

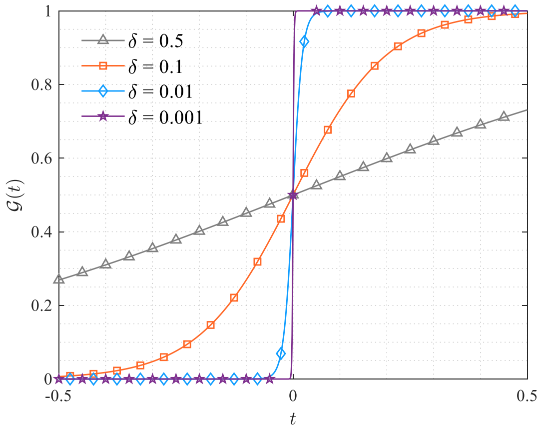


Fig. 4. Modified Sigmoid function for different values of δ.

The complete process of the sampling network is as follows. First, a learnable parameter matrix $\mathbf{T}{\in}\mathbb{R}^{C_r\times C_t}$ is defined, with each element of the matrix being initialized to a random value and then updated through gradient back-propagation. A Transformation block, denoted with $\Gamma(\cdot)$, is then used to process this matrix into an approximate sampling matrix $\hat{\mathbf{S}}$:

$$\hat{\mathbf{S}} = \Gamma(\mathbf{T}). \tag{6}$$

Following the Transformation block, an element-wise multiplication is performed between the resulting output $\hat{\mathbf{S}}$ and the RSRP matrix of all beam pairs $\mathbf{B}$, i.e., $\hat{\mathbf{B}}{=}\hat{\mathbf{S}} \odot \mathbf{B}$. The sampled values serve as input to the subsequent beam prediction network.

It is worth noting that the Transformation block can incorporate various DNN modules, such as fully connected layers and convolution layers. In the proposed solution, we opt to incorporate only the modified Sigmoid function $\mathcal{G}(\cdot)$ into the Transformation block with no additional modules; it can be written as $\hat{\mathbf{S}}{=}\Gamma(\mathbf{T}){=}\mathcal{G}(\mathbf{T})$, where $\mathcal{G}(\mathbf{T})$ denotes element-wise computation on $\mathbf{T}$.

During the end-to-end training stage, the learnable parameter matrix $\mathbf{T}$ and the approximate sampling matrix $\hat{\mathbf{S}}$ are continuously updated through end-to-end gradient back-propagation. It is important to highlight that the approximate sampling matrix $\hat{\mathbf{S}}$ acquired through the sampling network is not strictly constrained to discrete values of 0 or 1. Rather, by leveraging the modified Sigmoid function and an appropriate value of δ, each element in the learned approximate sampling matrix lies between 0 and 1, while closely approximating either 0 or 1. Therefore, after the end-to-end training stage, we can obtain the desired sampling matrix $\mathbf{S}$ by enforcing a binary constraint on the learned matrix $\hat{\mathbf{S}}$. The binary constraint involves sorting the elements of the learned matrix $\hat{\mathbf{S}}$, selecting the top K elements with the largest values and setting them to 1, while setting the remaining ones to zero. Consequently,

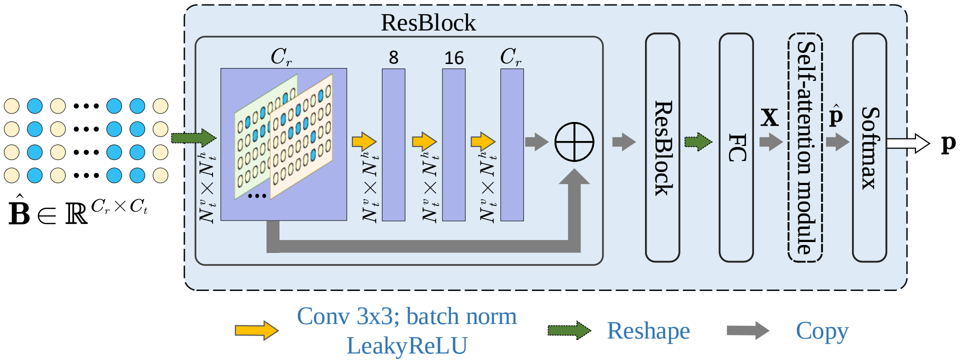


Fig. 5. Structure of the proposed BPNN.

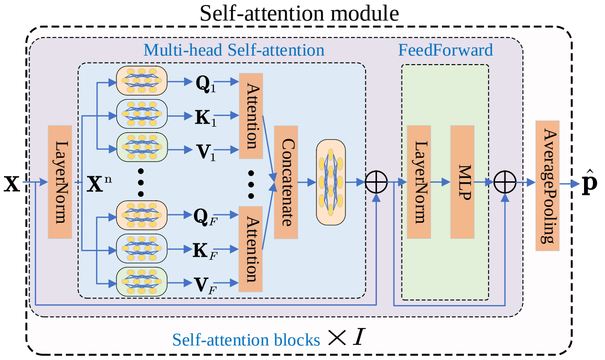


Fig. 6. Structure of the self-attention module.

fine-tuning the BPNN is imperative, after the completion of the end-to-end training of the integrated network.

C. Beam Prediction Network

We propose a novel architecture for the BPNN, as illustrated in Fig. 5. Drawing inspiration from the spatial characteristics of UPA and 3D beam formation, our proposed BPNN initially reshapes the sampled RSRP matrix $\hat{\mathbf{B}}\in\mathbb{R}^{C_r\times C_t}$ into a matrix with dimensions $C_r\times N_t^v\times N_t^h$ (as an example, N_t^v=4 and N_t^h=8 in Fig. 5), which can be viewed as a multi-channel image. In this reshaped format, the receive beams serve as channels, while the transmit beams are organized into a matrix of dimensions $N_t^v\times N_t^h$ based on their spatial positions.

In the proposed BPNN, two Convolutional Neural Networks (CNN) and self-attention mechanisms are employed subsequently. Initially, the two CNNs are used to extract shallow spatial relationships between different beams pairs, leveraging the strong correlations present in adjacent beam measurements, which resemble image-like structures. Subsequently, self-attention mechanisms are employed to capture deeper relationships between different beam pairs, determining the importance of each beam pair and outputting a weight vector $\mathbf{p}$ representing the optimal beam pair distribution. The adoption of self-attention mechanisms is motivated by their effectiveness in both capturing correlations between features and focusing attention on important features.

Specifically, two CNN-based Residual Blocks (ResBlocks) are used initially. Each ResBlock comprises three convolutional layers for feature extraction, including 8, 16, and C_r filters, respectively; all filters have size 3×3. These specific choices of convolutional layers are designed to progressively increase the depth and abstraction of the features being extracted. Starting with a smaller number of filters allows the network to capture basic features, while, thanks to the higher number of filters, the subsequent layers enable the network to learn more complex relationship and hierarchical representations of the measurements of all beam pairs. A residual unit is employed at the end of each ResBlock to prevent gradient explosion.

Following these two CNN-based Resblocks, we obtain hidden layer features with dimensions of $C_r\times N_t^v\times N_t^h$, which are reshaped into $C_r\times N_t^v N_t^h$ and fed into fully connected (FC) layers. The resulting hidden features are denoted by $\mathbf{X}\in\mathbb{R}^{C_r\times C_t C_r}$, where the dimension of each feature is equal to the total number of beam pairs. Subsequently, a self-attention module is applied to the hidden features $\mathbf{X}$.

Fig. 6 presents the detail of the self-attention module. We cascade several (namely, I) self-attention blocks in the self-attention module to improve the network performance. The detailed structure of a self-attention block is also depicted in Fig. 6. Specifically, in each block, a multi-head self-attention mechanism followed by a feedforward module is implemented. For the sake of brevity, we describe the operations inside the first block as an example; the operations in the other blocks are identical.

First, after the LayerNorm operation, we obtain $\mathbf{X}^{\mathrm{n}}=\mathrm{LayerNorm}\left(\mathbf{X}\right)$; then a multi-head (namely, F-head) self-attention mechanism is performed; each head has dimension d_h.

For each $i = 1,\ldots,F$ the self-attention mechanism is characterized by the query matrix $\mathbf{Q}_i$, the key matrix $\mathbf{K}_i$, and the value matrix $\mathbf{V}_i$ [28]. These matrices are calculated as follows:

$$\begin{aligned} \mathbf{Q}_i &= \mathbf{X}^{\mathrm{n}}\mathbf{W}_i^Q \\ \mathbf{K}_i &= \mathbf{X}^{\mathrm{n}}\mathbf{W}_i^K, \quad i=1,\ldots,F \\ \mathbf{V}_i &= \mathbf{X}^{\mathrm{n}}\mathbf{W}_i^V \end{aligned} \tag{7}$$

where $\mathbf{W}_i^Q\in\mathbb{R}^{C_t C_r\times d_h}$, $\mathbf{W}_i^K\in\mathbb{R}^{C_t C_r\times d_h}$ and $\mathbf{W}_i^V\in\mathbb{R}^{C_t C_r\times d_h}$ denote the linear transformation matrices with trainable weight elements.

The output of the single-head self-attention mechanism can be obtained as:

$$\mathbf{D}_i = \mathrm{Attention}(\mathbf{Q}_i, \mathbf{K}_i, \mathbf{V}_i) = \mathrm{Softmax}\left(\frac{\mathbf{Q}_i\mathbf{K}_i^T}{\sqrt{C_tC_r}}\right)\mathbf{V}_i, \tag{8}$$

where $\sqrt{C_tC_r}$ is the scaling factor that keeps the gradient stable. Each row of $\frac{\mathbf{Q}_i\mathbf{K}_i^T}{\sqrt{C_tC_r}}$ is the raw attention scores obtained from the dot product of a query vector (one row vector in $\mathbf{Q}_i$) and key vectors (row vectors in $\mathbf{K}_i$). The Softmax function is used to transform each attention scores vector into a vector of probabilities. This probability distribution assigns larger attention weights to more relevant values and smaller weights to less relevant ones.

Multi-head attention mechanism allows the model to extract features in different subspaces, where the results of different attention heads are concatenated and then projected, as shown in Fig. 6. The output of the multi-head attention mechanism is computed as:

$$\mathrm{MultiHead}(\mathbf{X}^{\mathrm{n}}) = [\mathbf{D}_1, \ldots, \mathbf{D}_F]\mathbf{W}^O, \tag{9}$$

where $\mathbf{W}^O \in \mathbb{R}^{Fd_h \times C_tC_r}$ is the linear transformation matrix with trainable weight elements.

Subsequently, the scheme uses a residual unit and a feed-forward module, including a LayNorm operation and a multi-layer perceptron (MLP) with one hidden layer containing 128 neurons. For the i-the self-attention block, assume $\mathbf{X}_i$ as the input, and $\mathbf{X}_1 = \mathbf{X}$, then we have:

$$\begin{aligned} &\mathbf{X}_i^{\mathrm{n}} = \mathrm{LayerNorm}\left(\mathbf{X}_i\right) \\ &\mathbf{Y}_i = \mathrm{MultiHead}(\mathbf{X}_i^{\mathrm{n}}) + \mathbf{X}_i^{\mathrm{n}} \qquad\qquad i = 1, \ldots, F\,, \\ &\mathbf{X}_{i+1}^{\mathrm{n}} = \mathrm{MLP}(\mathrm{LayerNorm}(\mathbf{Y}_i)) + \mathbf{Y}_i \end{aligned} \tag{10}$$

Following I self-attention blocks, we obtain an output of size $C_r \times C_tC_r$. The average along the first dimension is computed by the AveragePooling block to get a vector $\hat{\mathbf{p}}$ of size C_tC_r, which, as mentioned, is then passed through a Softmax layer to yield a probability vector $\mathbf{p} \in \mathbb{R}^{C_rC_t}$, representing the probability distribution of the optimal beam pair index.

Notably, the number of the self-attention blocks, I, and the number of the attention heads, F, in the multi-head attention mechanisms affect the complexity of the neural network. To balance model complexity and performance, we set $I{=}2$ and $F{=}32$ (unless otherwise specified). In Section IV-B, we assess the DeePP performance for different levels of model complexity.

D. Loss Function

In the end-to-end training stage, the probing beam pattern is learned simultaneously with the preliminary training of the beam prediction neural network. The entire network thus undergoes end-to-end gradient back-propagation using a designed multi-objective loss function, $\mathcal{L}_1$, defined as:

$$\mathcal{L}_1 = H(\mathbf{z}, \mathbf{p}) + \left|\sum_{i=1}^{C_r}\sum_{j=1}^{C_t}\hat{s}_{i,j} - K\right|^2, \tag{11}$$

where the first term represents the prediction performance loss, while the second term imposes a constraint on the size of the sampled beam pair set. We recall that $\mathbf{p}$ is the output of the BPNN and denotes the predicted probability distribution of the optimal beam pair, whereas we denote by $\mathbf{z} \in \mathbb{R}^{C_rC_t}$ the ground-truth optimal beam pair distribution vector. All elements of $\mathbf{z}$ are zero except for the one corresponding to the optimal beam pair which is valued at 1. Such element is denoted by $z_{\mathrm{opt}}{=}1$. In the implementation, $\mathbf{z}$ can be obtained by sorting the RSRP of all beam pairs and setting the element corresponding to the optimal beam pair to 1, while setting all other elements to 0.

Since the beam prediction task can be modeled as a multi-class classification problem, we use the classic cross-entropy loss function $H(\mathbf{z}, \mathbf{p})$ as the first term. Given the structure of the vector $\mathbf{z}$, the categorical cross entropy loss reduces to:

$$H(\mathbf{z}, \mathbf{p}) = -\sum_{i=1}^{C_rC_t} z_i \cdot \log(p_i) = -\log(p_{\mathrm{opt}}), \tag{12}$$

where p_{opt} represents the probability predicted by the model for the actual optimal beam pair. In the second term of (11), we use the mean square error (MSE) between the sum of the elements in the learned approximate sampling matrix $\hat{\mathbf{S}}$ and the predefined size of probing beam pairs to ensure that the number of the learned probing beam pairs approximates the preset value.

Following the first-stage training, the sampling network learns an approximate sampling matrix $\hat{\mathbf{S}}$, i.e., we obtain a trained BPNN. As mentioned in Section III-B, fine-tuning of the BPNN is however required after the completion of end-to-end training of the integrated network. In the fine-tuning stage, the cross-entropy function is used as loss function, denoted by $\mathcal{L}_2$:

$$\mathcal{L}_2 = H(\mathbf{z}, \mathbf{p})\,. \tag{13}$$

For the sake of clarity, the equations representing the parameter updating based on the loss function are reported in the Appendix.

IV. PERFORMANCE EVALUATION

In this section, we evaluate the performance of the proposed DeePP solution in indoor scenarios. We start by using a 3GPP-compliant indoor factory channel model to generate the dataset to be employed in our simulations and show the superiority of DeePP in beam prediction with learned probing beam patterns, relatively to traditional schemes using spatial-regular beam patterns. Then we validate the effectiveness of the proposed scheme by leveraging an open-source indoor dataset. Importantly, we also demonstrate the lower complexity of our solution compared to state-of-the-art approaches.

A. Simulation Settings and Dataset Construction

We consider the single-cell indoor factory scenario specified by 3GPP [29] and depicted in Fig. 7. Also, to generate our dataset, we adopt the parameters settings and modeling procedures specified by the 3GPP standard for an indoor mmWave channel (referred to as Indoor Factory (INF) channel). The parameter configuration is outlined in Table I.

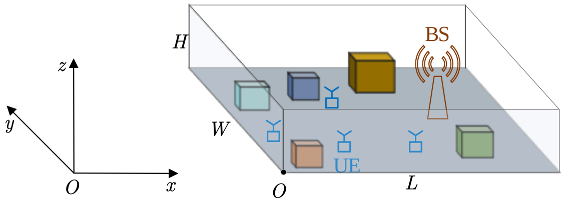


Fig. 7. Single-cell indoor factory scenario and associated global coordinate system.

TABLE I
PARAMETER SETTINGS

Parameter	**Value**
Frequency Range	FR2 @ 30 GHz
Indoor Factory size	(L,W,H)=(100,40,8) m
Density of clutters	40%
Deployment	A BS fixed at (80,20,6), UEs are randomly deployed
UE Speed	10 km/h
BS Antenna Configuration	UPA (M_1, M_2)=$(4, 8)$, d=$\frac{\lambda}{2}$
BS Antenna radiation pattern	3GPP model, TR 38.802 Table A.2.1-6, A.2.1-7
UE Antenna Configuration	UPA (M_1, M_2)=$(1, 4)$, d=$\frac{\lambda}{2}$
UE Antenna radiation pattern	Isotropic
BS Antenna bearing angle	-90 degrees w.r.t. GCS
UE Antenna bearing angle	90 degrees w.r.t. GCS
BS Antenna height	6 m
UE Antenna height	1 m
BS beams setup	Vertical 4 beams {9π/16, 11π/16, 13π/16, 15π/16} Horizontal 8 beams {-7π/16, -5π/16, -3π/16, -1π/16, 1π/16, 3π/16, 5π/16, 7π/16}
UE beams setup	Horizontal 4 beams {-3π/8, -1π/8, 1π/8, 3π/8}

As depicted in Fig. 7, the single-cell indoor factory scenario refers to a factory hall of size (L, W, H)=$(100, 40, 8)$ m, where L, W, and H denote the hall length, width and height, respectively. The hall contains metallic machinery of small to medium size, along with various irregular objects collectively referred to as clutters in the INF channel model. The density of clutters is set to 40%, indicating that clutters occupy 40% of the floor area, resulting in a high probability of Non-Line-of-Sight (NLOS) transmissions.

As depicted in Fig. 7, a global coordinate system (GCS) is defined, with the origin of the GCS positioned at the bottom left corner of the hall, while the hall length, width and height (L, W and H) are measured along the x, y, and z axes, respectively. In our experiments, a BS is located at (x, y, z)=$(80, 20, 6)$ m, while UEs (such as robots, sensors, or handsets) are randomly deployed. Notice that, since we focus on an indoor indoor scenario, the UE speed is set to a low value (namely, 10 km/h).

At the BS, a UPA with $4{\times}8$ elements is deployed, while each UE is equipped with a UPA featuring $1{\times}4$ elements[2]. The tilt and slant angles of both the BS and UE panels are set to zero, indicating that the panels are perpendicular to the floor. The bearing angles for both the BS and UE panels are -90 degree and 90 degree w.r.t GCS, respectively .

With regard to the beam configuration, C_t=32 transmit beams are employed at the BS with N_t^v=4 zenith beam angles {9π/16, 11π/16, 13π/16, 15π/16} and N_t^h=8 azimuth beam angles {-7π/16, -5π/16, -3π/16, -1π/16, 1π/16, 3π/16, 5π/16, 7π/16} w.r.t LCS, while C_r=4 receive beams are used at each UE with horizontal beam angles {-3π/8, -1π/8, 1π/8, 3π/8} w.r.t LCS (see Fig. 1 for the LCS associated with each panel and the beam angles).

For dataset generation, we randomly deploy one UE for N_{sample} times, performing beam sweeping and measurement for each UE deployment. This procedure allows for the collection of RSRP values of all beam pairs for every UE. Each sample has dimensions of $C_r{\times}C_t$=4×32, and the the resulting dataset is $\mathcal{D}{\in}\mathbb{R}^{N_{\text{sample}}\times 4\times 32}$. The dataset included N_{sample}=$100,000$ samples, 80% of which are used for training, 10% for validation, and 10% for testing. Training is conducted on an NVIDIA RTX 2080Ti GPU, employing the Adam optimizer with a batch size of 256. A cosine annealing learning rate schedule is implemented, varying from 0.001 to 0.0001 over 100 epochs.

B. Performance Metrics, Benchmarks, and Numerical Results

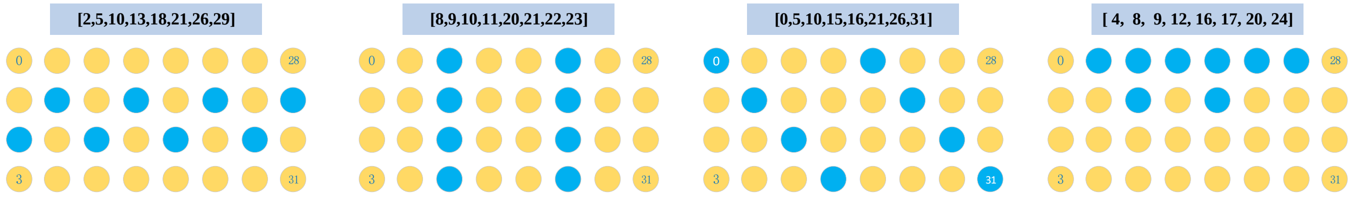
Using the constructed dataset, we conduct a thorough evaluation of the beam prediction performance. To ensure a comprehensive performance evaluation, we adopt the key performance indicators (KPIs) recommended by 3GPP for evaluating AI-based spatial beam prediction. Specifically, we evaluate the following KPIs:

- **KPI-1: Top-Q Beam Prediction Accuracy (percentage)**, quantifying the percentage of cases where the top-1 genie-aided beam (or beam pair) based on genie-aided RSRPs, is among the top-Q predicted beams (or beam pairs) based on the predicted optimal beam distribution vector;
- **KPI-2: Marginal Beam Prediction Accuracy (Percentage)**, which indicates the percentage of instances where the genie-based RSRP of the top-1 predicted beam (or beam pair) falls within a specified difference threshold (in terms of dB), compared to the genie-aided RSRP of the top-1 genie-aided beam (or beam pair);
- **KPI-3: Distribution of RSRP Differences**, expressing the distribution of differences between the RSRP of the genie's best beam (or beam pair) and the predicted best beam (or beam pair).

The traditional probing beam sampling scheme primarily adopts spatial-regular sampling. Within the realm of spatial-regular sampling, there exist several alternative patterns, three of which we enumerate and denote as pattern P1, pattern P2, and pattern P3. Notably, given the limited number of antenna elements and beams at the UE, existing work typically selects a limited number of transmit beams while all receiver beams are chosen. Following this convention, in our experiment we adopt the same approach, e.g., K_t probing beams are sampled from the C_t transmit beams while K_r=C_r receive beams are

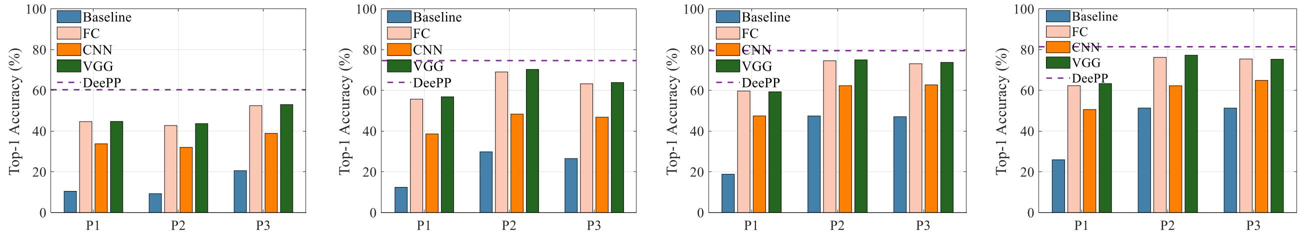
---

[2] Although in our experiments a ULA is deployed at each UE, without loss of generality we refer to both settings as UPA for simplicity, as a ULA can be regarded as a special case of a UPA.



(a) (b) (c) (d)

Fig. 8. Learned probing-beam pattern vs. the traditional ones for K_t=8. (a) Traditional beam pattern P1; (b) Traditional beam pattern P2; (c) Traditional beam pattern P3; (d) Beam pattern learned by DeePP.



(a) (b) (c) (d)

Fig. 9. Top-1 beam prediction accuracy comparison between DeePP and its benchmark schemes, under different sizes of probing-beam pairs. (a) $K_r \times K_t$=4×4; (b) $K_r \times K_t$=4×8; (c) $K_r \times K_t$=4×12; (d) $K_r \times K_t$=4×16.

selected[3], thus resulting in $K = K_rK_t = C_rK_t$. As an example, the sampling patterns of transmit beams, i.e., P1, P2 and P3 are illustrated in Fig. 8(a), Fig. 8(b), and Fig. 8(c), respectively, for a number of probing transmit beams K_t=8 (in the figures, the rows and columns of each rectangle represent the transmit beams in the vertical and horizontal directions, respectively; and the blue (yellow) circles correspond to selected (unselected) beams). The order of the transmit beams is illustrated in Fig. 8. Based on these traditional probing beam patterns, we compare DeePP against the following benchmarks:

- **Baseline scheme:** A non-ML method which predicts the beam pair with the largest RSRP among the probing beam pairs as the optimal one;
- **FC-based scheme:** A beam prediction network based on fully-connected (FC) neural networks;
- **CNN-based scheme:** A beam prediction network based on CNN;
- **VGG-based scheme** [26]: Visual Geometry Group (VGG) based network VGG-BMNet, recognized as the state-of-the-art (SOTA) scheme for spatial beam prediction.

Fig. 9 compares the top-1 beam prediction accuracy of DeePP and its alternatives under different sizes of probing beam pairs, specifically $K_r \times K_t$=4×4, $K_r \times K_t$=4×8, $K_r \times K_t$=4×12, and $K_r \times K_t$=4×16. The bars in Fig. 9 illustrate the prediction accuracy of various benchmark schemes under traditional probing beam patterns, while the line represents the performance of our solution. It can be observed that DeePP consistently achieves the highest prediction accuracy across varying probing beam sizes. This superiority can be attributed to the fact that traditional benchmark approaches, relying on regular sampling of beam patterns, often fail to select the most crucial beams for accurate prediction.

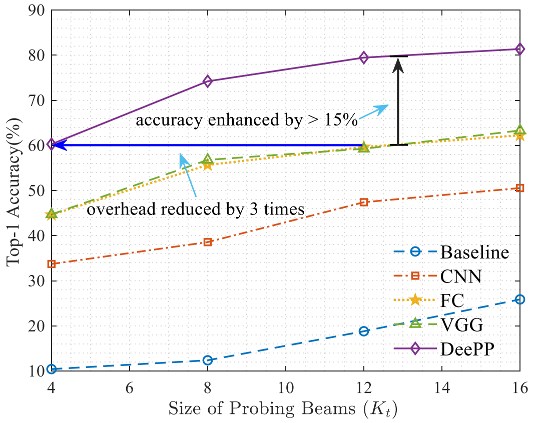


Fig. 10. Top-1 accuracy against the number of the probing beams K_t under different schemes.

To shed further light on DeePP advantages, Fig. 10 compares the top-1 beam prediction accuracy of DeePP with that of its alternatives versus the size of probing beams K_t, using P1. For a fixed number of probing beams, we observe that DeePP consistently outperforms its benchmarks by more than 15% in top-1 accuracy. Further, for a given value of top-1 accuracy, e.g., 60%, DeePP just requires 4×4 probing beam pairs, while the best-performing benchmark requires 4×12 pairs. In other words, for a 60% top-1 accuracy and under P1, DeePP reduces the probing overhead by at least three times, along with the corresponding latency, when compared

---

[3] Our proposed scheme is a general method. By setting each row vector in the learnable parameter matrix $\mathbf{T}$ to be the same, it learns the sampling pattern solely for the transmitting beams.

TABLE II
PROBING-BEAM PATTERNS FOR DIFFERENT SCHEMES

Pattern	$K_t = 4$	$K_t = 8$
P1	[10,13,18,21]	[2,5,10,13,18,21,26,29]
P2	[9,10,21,22]	[8,9,10,11,20,21,22,23]
P3	[8,13,18,23]	[0,5,10,15,16,21,26,31]
Learned	[4, 8,12,16]	[4,8, 9,12,16,17,20,24]
	$K_t = 12$	
P1	[2,5,6, 9,10,13,18,21,22,25,26,29]	
P2	[4,5,6, 7,12,13,14,15,20,21,22,23]	
P3	[0,5,8,10,13,15,16,18,21,23,26,31]	
Learned	[0,4,8, 9,12,14,16,17,20,22,23,24]	
	$K_t = 16$	
P1	[1,2,5,6,9,10,13,14,17,18,21,22,25,26,29,30]	
P2	[0,1,2,3,8, 9,10,11,16,17,18,19,24,25,26,27]	
P3	[0,3,5,6,9,10,12,15,16,19,21,22,25,26,28,31]	
Learned	[0,1,4,6,8, 9,12,14,15,16,17,18,20,22,23,24]	

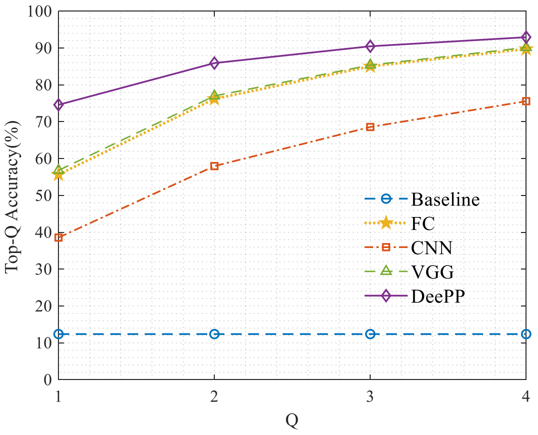


Fig. 11. Top-Q beam prediction accuracy for $K_r \times K_t$=4×8, when DeePP and its benchmarks are evaluated under pattern 1.

to the state-of-the-art beam prediction scheme. This clearly demonstrates the effectiveness of the proposed solution.

Table II presents the traditional patterns and the pattern learned by DeePP for different sizes of probing beam pairs. Additionally, Fig. 8 compares the learned probing beam pattern against the traditional patterns when $K_r \times K_t$=4×8; the learned pattern is depicted in Fig. 8(d). Notably, the pattern learned by DeePP exhibits a non-regular sampling. This observation aligns with our modeling approach, where the selection of probing beams is treated as a fitting problem of a non-regular sampling function. Moreover, it underscores the effectiveness of the proposed sampling network, which can discern more critical beam pairs by learning the features of a specific environment.

Furthermore, Fig. 11, Fig. 12, and Fig. 13 respectively present the evaluation results for the proposed three KPIs: Top-Q beam prediction accuracy, marginal beam prediction accuracy, and RSRP difference distribution. The evaluation is conducted for $K_r \times K_t$=4×8, under P1. Notably, the proposed scheme exhibits significant advantages in terms of all evaluated KPIs.

Next, to further verify its superiority, we assess DeePP performance using a publicly available dataset for an indoor office scenario [30]. The scenario includes 64 transmit beams and 4 receive beams. We compare DeePP against the baseline beam prediction as well as several ML-based schemes, using a probing beam size K_t=8. Table III presents the top-Q beam prediction accuracy for the different schemes, while Table IV provides the results of marginal beam prediction accuracy

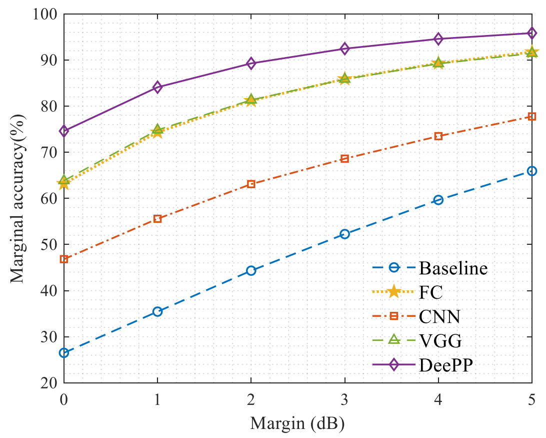


Fig. 12. Marginal accuracy of DeePP and its alternatives, for $K_r \times K_t$=4×8, under pattern 1.

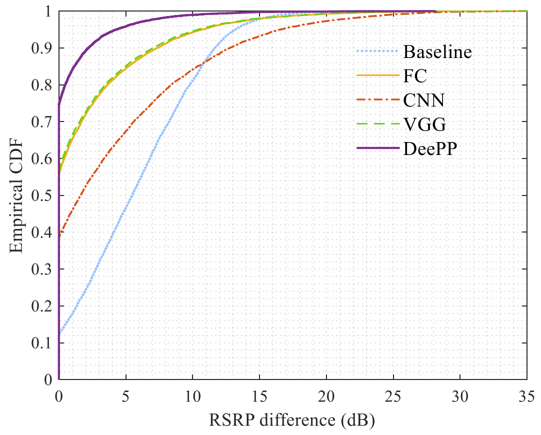


Fig. 13. CDF of RSRP difference for DeePP and its alternatives, when $K_r \times K_t$=4×8 and under pattern 1.

TABLE III
TOP-Q BEAM PREDICTION ACCURACY USING THE OPEN-SOURCE INDOOR DATASET

Scheme	**Pattern**	**Top-Q accuracy**			
		Q=1	**Q=2**	**Q=3**	**Q=4**
Baseline	P1	0.2397	0.2397	0.2397	0.2397
	P2	0.1865	0.1865	0.1865	0.1865
	P3	0.1692	0.1692	0.1692	0.1692
FC	P1	0.7913	0.9153	0.9508	0.9648
	P2	0.7618	0.8790	0.9223	0.9400
	P3	0.7455	0.8833	0.9298	0.9530
CNN	P1	0.4965	0.6973	0.7913	0.8498
	P2	0.3643	0.5318	0.6285	0.6980
	P3	0.4473	0.6388	0.7345	0.7920
VGG	P1	0.8125	0.9310	0.9620	0.9725
	P2	0.7895	0.9033	0.9360	0.9500
	P3	0.7740	0.8960	0.9380	0.9558
DeePP	**Learned**	**0.8553**	**0.9520**	**0.9715**	**0.9763**

TABLE IV
MARGINAL BEAM PREDICTION ACCURACY USING THE OPEN-SOURCE INDOOR DATASET

Scheme	**Pattern**	**Marginal beam prediction accuracy**					
		0dB	**1dB**	**2dB**	**3dB**	**4dB**	**5dB**
Baseline	P1	0.2397	0.2758	0.3065	0.3400	0.3792	0.4120
	P2	0.1865	0.1945	0.2010	0.2100	0.2183	0.2295
	P3	0.1692	0.1953	0.2198	0.251	0.2815	0.3103
FC	P1	0.7913	0.8553	0.8863	0.9075	0.922	0.9318
	P2	0.7618	0.8133	0.8370	0.8475	0.8590	0.8650
	P3	0.7455	0.7950	0.8150	0.8313	0.8418	0.8473
CNN	P1	0.4965	0.5603	0.6075	0.6480	0.6850	0.7100
	P2	0.3643	0.4020	0.4245	0.4363	0.4520	0.4650
	P3	0.4473	0.4893	0.5193	0.5448	0.5643	0.5800
VGG	P1	0.8125	0.8728	0.9008	0.9158	0.9288	0.9373
	P2	0.7895	0.8375	0.8565	0.8658	0.8750	0.8798
	P3	0.7740	0.8230	0.8438	0.8563	0.8653	0.8713
DeePP	**Learned**	**0.8553**	**0.9053**	**0.9268**	**0.9355**	**0.9450**	**0.9518**

TABLE V
PREDICTION ACCURACY UNDER DIFFERENT BEAM PREDICTION APPROACHES

Model	**Learned**	**P1**	**P2**	**P3**	**Parameters**	**FLOPs**
FC	0.7389	0.5567	0.6900	0.6317	1181824	1179648
CNN	0.6787	0.3857	0.4832	0.4682	11907	1544832
VGG	0.746	0.5679	0.7025	0.6383	3107456	20596736
DeePP-S	**0.7280**				109748	502144
Proposed	**0.7459**				**219368**	**1004288**
DeePP-L	**0.7564**				1589456	6610432

against the RSRP margin. All the above results confirm the superiority of our scheme, highlighting a 4% to 8% gain in terms of top-1 accuracy and a 3% to 8% improvement in terms of 1 dB marginal beam prediction accuracy, relatively to the best performing benchmark (i.e., VGG).

Finally, considering that the beam prediction network is typically deployed at the UEs, which have limited storage and computational capabilities, using a lightweight model is of paramount importance. We therefore conducted a comparative analysis of the performance and complexity of the proposed beam prediction network against the considered benchmarks, as presented in Table V. The results demonstrate that not only does the proposed approach outperform its alternatives in terms of beam prediction accuracy, but it also significantly reduces the number of network parameters and computational complexity. For example, compared to the SOTA scheme VGG, DeePP reduces the number of network parameters by over 90% and decreases floating-point operations (FLOPs) by 20 times. This is due to the design of the BPNN based on CNNs and self-attention mechanisms, which account for the structure of a UPA and spatial positions of corresponding beams, thereby enhancing the efficiency of neurons in BPNN.

Notably, Table V also presents both small-scale and large-scale versions of our proposed BPNN, referred to as "DeePP-S" and "DeePP-L", respectively. Surprisingly, even with half the parameters, the proposed small-scale model achieves excellent prediction performance, which further underscores the practicality and superiority of our approach.

V. CONCLUSIONS

We designed a joint probing-beam pattern learning and beam prediction scheme, named DeePP, which leverages a lightweight, integrated neural network. Such network includes a sampling network that has been designed by treating the selection of probing beam pairs as a sampling operation and introducing a modified sigmoid function to handle the non-differentiable characteristic of sampling, thus enabling the back-propagation of the network parameter gradients. Furthermore, inspired by the physical structure of antenna arrays and 3D-beam formation, we designed a highly efficient and effective BPNN. Therein CNNs are used for shallow features extraction and a subsequent self-attention mechanism for deeper feature extraction, which resulted into a high-performance beam distribution learning.

Experimental results clearly demonstrate the superiority of DeePP in enhancing beam prediction accuracy while simultaneously reducing model complexity. When compared to the SOTA benchmark using a traditional probing beam pattern (i.e., P1), DeePP yields more than 15% improvement in in top-1 accuracy, while using a neural network with almost 90% less parameters than existing DL-based solutions. Our approach not only contributes insights on ML-based beam management but also provides practical solutions for lightweighting ML-enabled beam management. Future work will then extend our approach to ML-based spatial and temporal beam management in multi-cell scenarios.

APPENDIX

Based on a loss function $\mathcal{L}$ ($\mathcal{L}_1$ for the first training stage and $\mathcal{L}_2$ for the fine-tuning stage), the update formula for the network parameters is given by:

$$\boldsymbol{\theta}_{t+1} = \boldsymbol{\theta}_t - \eta \nabla_{\boldsymbol{\theta}} \mathcal{L}, \tag{14}$$

where the vector $\boldsymbol{\theta}$ contains the network parameters, $\boldsymbol{\theta}_{t+1}$ represents the updated parameters and $\boldsymbol{\theta}_t$ denotes the current parameters. η is the learning rate and $\nabla_{\theta}\mathcal{L}$ is the gradient of the loss function, $\mathcal{L}$, with respect to each element of $\boldsymbol{\theta}$. Note that in the end-to-end training stage, the parameter set contains the parameters of the sampling network (e.g., the elements of $\mathbf{T}$) and the parameters characterizing the BPNN (e.g., ResBlocks and Self-attention module). In the fine-tuning stage, the parameter set only contains the parameters in the BPNN.

REFERENCES

[1] D. C. Nguyen, M. Ding, P. N. Pathirana, A. Seneviratne, J. Li, D. Niyato, O. Dobre, and H. V. Poor, “6G Internet of Things: A Comprehensive Survey,” *IEEE Internet Things J.*, vol. 9, no. 1, pp. 359–383, Jan. 2022.

[2] A. Mahmood, L. Beltramelli, S. Fakhrul Abedin, S. Zeb, N. I. Mowla, S. A. Hassan, E. Sisinni, and M. Gidlund, “Industrial IoT in 5G-and-Beyond Networks: Vision, Architecture, and Design Trends,” *IEEE Trans. Ind. Informat.*, vol. 18, no. 6, pp. 4122–4137, Jun. 2022.

[3] C.-X. Wang, X. You, X. Gao, X. Zhu, Z. Li, C. Zhang, H. Wang, Y. Huang, Y. Chen, H. Haas, J. S. Thompson, E. G. Larsson, M. D. Renzo, W. Tong, P. Zhu, X. Shen, H. V. Poor, and L. Hanzo, “On the Road to 6G: Visions, Requirements, Key Technologies, and Testbeds,” *IEEE Commun. Surveys Tuts.*, vol. 25, no. 2, pp. 905–974, 2023.

[4] M. Giordani, M. Polese, A. Roy, D. Castor, and M. Zorzi, “A Tutorial on Beam Management for 3GPP NR at mmWave Frequencies,” *IEEE Commun. Surveys Tuts.*, vol. 21, no. 1, pp. 173–196, 2019.

[5] Y.-N. R. Li, B. Gao, X. Zhang, and K. Huang, “Beam Management in Millimeter-Wave Communications for 5G and Beyond,” *IEEE Access*, vol. 8, pp. 13 282–13 293, 2020.

[6] L. Wei, Q. Li, and G. Wu, “Exhaustive, Iterative and Hybrid Initial Access Techniques in mmWave Communications,” in *2017 IEEE Wireless Communications and Networking Conference (WCNC)*, Mar. 2017, pp. 1–6.

[7] M. Giordani, M. Mezzavilla, C. N. Barati, S. Rangan, and M. Zorzi, “Comparative analysis of initial access techniques in 5G mmWave cellular networks,” in *2016 Annual Conference on Information Science and Systems (CISS)*. IEEE, Mar. 2016, pp. 268–273.

[8] S. Han, T. Xie, C.-L. I, L. Chai, Z. Liu, Y. Yuan, and C. Cui, “Artificial-Intelligence-Enabled Air Interface for 6G: Solutions, Challenges, and Standardization Impacts,” *IEEE Commun. Mag.*, vol. 58, no. 10, pp. 73–79, Oct. 2020.

[9] K. Ma, Z. Wang, W. Tian, S. Chen, and L. Hanzo, “Deep Learning for mmWave Beam-Management: State-of-the-Art, Opportunities and Challenges,” *IEEE Wireless Commun.*, vol. 30, no. 4, pp. 108–114, Aug. 2023.

[10] M. Qurratulain Khan, A. Gaber, P. Schulz, and G. Fettweis, “Machine Learning for Millimeter Wave and Terahertz Beam Management: A Survey and Open Challenges,” *IEEE Access*, vol. 11, pp. 11 880–11 902, 2023.

[11] RP-213599, “New SI: Study on Artificial Intelligence (AI)/Machine Learning (ML) for NR Air Interface,” in *3GPP TSG RAN Meeting # 94e*, Dec. 2021.

[12] Y. Wang, A. Klautau, M. Ribero, A. C. K. Soong, and R. W. Heath, “MmWave Vehicular Beam Selection With Situational Awareness Using Machine Learning,” *IEEE Access*, vol. 7, pp. 87 479–87 493, 2019.

[13] W. Xu, F. Gao, S. Jin, and A. Alkhateeb, “3D Scene-Based Beam Selection for mmWave Communications,” *IEEE Wireless Commun. Lett.*, vol. 9, no. 11, pp. 1850–1854, Nov. 2020.

[14] S. Rezaie, C. N. Manchón, and E. de Carvalho, “Location- and Orientation-Aided Millimeter Wave Beam Selection Using Deep Learning,” in *ICC 2020 - 2020 IEEE International Conference on Communications (ICC)*, Jun. 2020, pp. 1–6.

[15] K. N. Nguyen, A. Ali, J. Mo, B. L. Ng, V. Va, and J. C. Zhang, “Beam Management with Orientation and RSRP using Deep Learning for Beyond 5G Systems,” in *2022 IEEE International Conference on Communications Workshops (ICC Workshops)*, May 2022, pp. 133–138.

[16] S. Rezaie, A. Amiri, E. de Carvalho, and C. N. Manchón, “Deep Transfer Learning for Location-Aware Millimeter Wave Beam Selection,” *IEEE Commun. Lett.*, vol. 25, no. 9, pp. 2963–2967, Sep. 2021.

[17] M. B. Mashhadi, M. Jankowski, T.-Y. Tung, S. Kobus, and D. Gündüz, “Federated mmWave Beam Selection Utilizing LIDAR Data,” *IEEE Wireless Commun. Lett.*, vol. 10, no. 10, pp. 2269–2273, Oct. 2021.

[18] V. Va, T. Shimizu, G. Bansal, and R. W. Heath, “Online Learning for Position-Aided Millimeter Wave Beam Training,” *IEEE Access*, vol. 7, pp. 30 507–30 526, 2019.

[19] S. H. Lim, S. Kim, B. Shim, and J. W. Choi, “Deep Learning-Based Beam Tracking for Millimeter-Wave Communications Under Mobility,” *IEEE Trans. Commun.*, vol. 69, no. 11, pp. 7458–7469, Nov. 2021.

[20] R. Wang, C. She, Y. Li, and B. Vucetic, “Attention Mechanism for Beam Prediction in mmWave Communications with Mission-Critical Applications,” in *2022 IEEE Globecom Workshops (GC Wkshps)*, Dec. 2022, pp. 1454–1459.

[21] Q. Li, P. Sisk, A. Kannan, T. Yoo, T. Luo, G. Shah, B. Manjunath, C. Samarathungage, M. T. Boroujeni, H. Pezeshki, and H. Joshi, “Machine Learning Based Time Domain Millimeter-Wave Beam Prediction for 5G-Advanced and Beyond: Design, Analysis, and Over-The-Air Experiments,” *IEEE J. Sel. Areas Commun.*, vol. 41, no. 6, pp. 1787–1809, Jun. 2023.

[22] H. Echigo, Y. Cao, M. Bouazizi, and T. Ohtsuki, “A Deep Learning-Based Low Overhead Beam Selection in mmWave Communications,” *IEEE Trans. Veh. Technol.*, vol. 70, no. 1, pp. 682–691, Jan. 2021.

[23] S.-C. Fan, H.-Y. Chang, C.-Y. Wang, and W.-H. Chung, “Super Resolution-Based Beam Selection With Hierarchical Codebook in mmWave Communication,” *IEEE Wireless Commun. Lett.*, vol. 11, no. 5, pp. 967–971, May 2022.

[24] C.-H. Lin, W.-C. Kao, S.-Q. Zhan, and T.-S. Lee, “BsNet: A Deep Learning-Based Beam Selection Method for mmWave Communications,” in *2019 IEEE 90th Vehicular Technology Conference (VTC2019-Fall)*, Sep. 2019, pp. 1–6.

[25] W. Ma, C. Qi, and G. Y. Li, “Machine Learning for Beam Alignment in Millimeter Wave Massive MIMO,” *IEEE Wireless Commun. Lett.*, vol. 9, no. 6, pp. 875–878, Jun. 2020.

[26] J. Zuo, J. Zhang, Y. Cao, X. Chen, F. Wang, N. Hu, and X. Xu, “Artificial Intelligence-Based Spatial Domain Beam Prediction for 5G Beyond,” in *2022 IEEE Globecom Workshops (GC Wkshps)*, Dec. 2022, pp. 1460–1465.

[27] Y. Heng, J. Mo, and J. G. Andrews, “Learning Site-Specific Probing Beams for Fast mmWave Beam Alignment,” *IEEE Trans. Wireless Commun.*, vol. 21, no. 8, pp. 5785–5800, Aug. 2022.

[28] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, “Attention is All you Need,” in *Advances in Neural Information Processing Systems*, vol. 30. Curran Associates, Inc., 2017.

[29] 3rd Generation Partnership Project, “Study on Channel Model for Frequencies from 0.5 to 100 GHz (Release 16),” in *3GPP Rep. TR 38.901 V16.1.0*, 2020.

[30] OPPO, “Dataset for AI-based beam management,” 2023. [Online]. Available: https://wireless-intelligence.com/