Deep Learning for Direct Hybrid Precoding in Millimeter Wave Massive MIMO Systems

Xiaofeng Li and Ahmed Alkhateeb

Arizona State University, Email: {xiaofen2, alkhateeb}@asu.edu

Abstract—This paper proposes a novel neural network architecture, that we call an *auto-precoder*, and a deep-learning based approach that jointly senses the millimeter wave (mmWave) channel and designs the hybrid precoding matrices with only a few training pilots. More specifically, the proposed machine learning model leverages the prior observations of the channel to achieve two objectives. First, it optimizes the compressive channel sensing vectors based on the surrounding environment in an unsupervised manner to focus the sensing power on the most promising spatial directions. This is enabled by a novel neural network architecture that accounts for the constraints on the RF chains and models the transmitter/receiver measurement matrices as two complex-valued convolutional layers. Second, the proposed model learns how to construct the RF beamforming vectors of the hybrid architectures directly from the projected channel vector (the received sensing vector). The auto-precoder neural network that incorporates both the channel sensing and beam prediction is trained end-to-end as a multi-task classification problem. Thanks to this design methodology that leverages the prior channel observations and the implicit awareness about the surrounding environment/user distributions, the proposed approach significantly reduces the training overhead compared to classical (non-machine learning) solutions. For example, for a system of 64 transmit and 64 receive antennas, with 3 RF chains at both sides, the proposed solution needs only 8 or 16 channel training pilots to directly predict the RF beamforming/combining vectors of the hybrid architectures and achieve near-optimal achievable rates. This highlights a promising solution for the channel estimation and hybrid precoding design problem in mmWave and massive MIMO systems.

Index Terms—Deep learning, channel estimation, hybrid precoding, millimeter wave, massive MIMO.

I. INTRODUCTION

Hybrid analog/digital architectures have attracted significant interest in the last few years thanks to their capability of achieving high data rates with energy-efficient hardware. To design the hybrid precoding matrices, however, an explicit estimation of the mmWave channel is normally required. This mmWave channel estimation is a challenging task because of the large numbers of antennas at both the transmitters and receivers, which result in high training overhead, and the strict hardware constraints on the RF chains [1], [2]. Leveraging the sparsity of the mmWave channels, several compressive sensing based channel estimation solutions have been proposed and showed promising performance [1], [3]–[5]. Essentially, these compressive sensing solutions for the mmWave channel estimation problem normally require an order of magnitude less training pilots compared to exhaustive search approaches [6]. But can we do better? In this paper, we show that machine learning tools can efficiently leverage the prior observations

about the channel estimates and the hybrid precoding designs to significantly reduce the training overhead associated with the mmWave channel training and precoding design problem.

Several channel estimation and hybrid precoding design approaches have been proposed in the last few years [1], [3], [4], [7]. Most of these approaches relied on leveraging the sparse nature of the mmWave channels and developed compressive sensing based solutions for the channel estimation. The developed solutions in [1], [3], [4], [7] generally show that compressive measurements/projections can efficiently sense and reconstruct the mmWave channels while requiring less training pilots compared to the exhaustive beam training techniques. These compressive sensing solutions, however, typically adopt random channel measurement vectors that distribute the sensing power in all spatial directions. For a given environment (for example, an outdoor street or an indoor room setting), though, one would expect that the base station/access point should focus the sensing power on the spatial directions that are likely to include the angles or arrival/departure of the channel paths. Intuitively, these directions depend on the given environment (geometry, materials, etc.) and the candidate user locations among other factors. Therefore, it is interesting to leverage any side information about the surrounding environment and user distribution in designing these sensing vectors.

In this paper, we propose a novel deep-learning based approach the jointly optimizes the channel measurement vectors and designs the hybrid beamforming vectors to achieve near-optimal data rates with negligible training overhead. More specifically, we develop a novel neural network architecture, that we call an *auto-precoder*, to achieve two main objectives: (i) It learns how to optimize the channel sensing vectors to focus the sensing power on the promising spatial directions (which implicitly adapt these measurement vectors to the surrounding environment/ user distributions) and (ii) it learns how to predict the hybrid beamforming vectors directly from the receive sensing vector (without the need to explicitly estimate/reconstruct the channel). Achieving these two objectives results in a promising channel sensing/precoding design approach that can predict near-optimal hybrid beamforming vectors while requiring negligible training overhead.

Notation: We use the following notation throughout this paper: $\mathbf{A}$ is a matrix, $\mathbf{a}$ is a vector, a is a scalar, and $\mathcal{A}$ is a set. $|\mathbf{A}|$ is the determinant of $\mathbf{A}$, whereas $\mathbf{A}^T$ and $\mathbf{A}^\mathsf{H}$ are its transpose and Hermitian (conjugate transpose). $\mathbf{A} \otimes \mathbf{B}$ is the Kronecker product of $\mathbf{A}$ and $\mathbf{B}$, and $\mathbf{A} \circ \mathbf{B}$ is their Khatri-Rao product. $\mathcal{N}_\mathbb{C}(\mathbf{m}, \mathbf{R})$ is a complex Gaussian random vector with mean $\mathbf{m}$ and covariance $\mathbf{R}$. $\mathbb{E}[\cdot]$ is used to denote expectation.
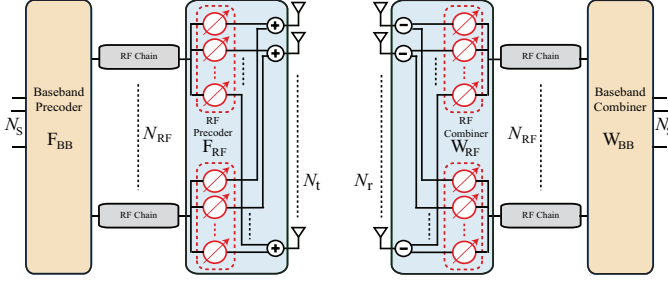
Fig. 1. A block diagram for the adopted hybrid analog/digital transceiver architecture with a transmitter employing N_t antennas and N_t^{RF} RF chains and a receiver employing N_r antennas and N_r^{RF} RF chains. The precoding/combining processing is divided between baseband precoding/combining $\mathbf{F}_{\mathrm{BB}}, \mathbf{W}_{\mathrm{BB}}$ and RF precoding/combining $\mathbf{F}_{\mathrm{RF}}, \mathbf{W}_{\mathrm{RF}}$.

II. SYSTEM AND CHANNEL MODELS

We consider the fully-connected hybrid analog/digital architecture depicted in Fig. 1, where a transmitter employing N_t antennas and N_t^{RF} RF chains is communicating via N_S streams with a receiver having N_r antennas and N_r^{RF} RF chains. The transmitter precodes the transmitted signal using an $N_t^{\mathrm{RF}} \times N_S$ baseband precoder $\mathbf{F}_{\mathrm{BB}}$ and an $N_t \times N_t^{\mathrm{RF}}$ RF precoder $\mathbf{F}_{\mathrm{RF}}$ while the receiver combines the received signal using the $N_r \times N_r^{\mathrm{RF}}$ RF combiner $\mathbf{W}_{\mathrm{RF}}$ and the $N_r^{\mathrm{RF}} \times N_S$ baseband combiner $\mathbf{W}_{\mathrm{BB}}$. Since the analog/RF precoders/combiners, $\mathbf{F}_{\mathrm{RF}}, \mathbf{W}_{\mathrm{RF}}$ are implemented in the analog domain using RF circuits, every entry of the RF precoders/combiners is assumed to have a constant-modulus, i.e., $[\mathbf{F}_{\mathrm{RF}}]_{n,m} = \frac{1}{\sqrt{N_t}} e^{j\Theta_{m,nt}}$ (and similarly for elements of $\mathbf{W}_{\mathrm{RF}}$). Further, the total power constraint is enforced by normalizing the baseband precoder $\mathbf{F}_{\mathrm{BB}}$ to satisfy $\|\mathbf{F}_{\mathrm{RF}} \mathbf{F}_{\mathrm{BB}}\|_F^2 = N_S$ [3].

For the channel between the transmit and receive antennas, we adopt the geometric channel model in [3] with L paths. In this model, the $N_r \times N_t$ channel matrix $\mathbf{H}$ is written as

$$\mathbf{H} = \sum_{\ell=1}^{L} \alpha_\ell \mathbf{a}_r (\phi_{r,\ell}, \theta_{r,\ell}) \mathbf{a}_t^{\mathsf{H}} (\phi_{t,\ell}, \theta_{t,\ell}) \qquad (1)$$

where α_ℓ denotes the complex path gain of the ℓth path (including the path loss). The angles $\phi_{r,\ell}, \theta_{r,\ell}$ represent the ℓth path azimuth and elevation angles of arrival (AoAs) at the receive antennas while $\phi_{t,\ell}, \theta_{t,\ell}$ represent the ℓth path angles of departure from the transmit array. Finally, $\mathbf{a}_t(.)$ and $\mathbf{a}_r(.)$ denote the transmit and receive array response vectors. The array response vectors for ULA and UPA arrays are defined in [8]. It is worth noting here that for mmWave frequencies, the measurements showed that the channels are typically sparse in the angular domain resulting in a small number of channel paths L (normally in the range of 3-5 paths) [9], [10].

III. PROBLEM DEFINITION

The general objective of this paper is to directly design the hybrid precoders/combiners to maximize the system achievable rate while minimizing the channel training overhead. Given the system and channel models in Section II, the achievable rate with the analog/digital precoders/combiners can be written as

$$R = \log_2 \left| \mathbf{I} + \mathbf{R}_n^{-1} \mathbf{W}^{\mathsf{H}} \mathbf{H} \mathbf{F} \mathbf{F}^{\mathsf{H}} \mathbf{H}^{\mathsf{H}} \mathbf{W} \right|, \qquad (2)$$

with $\mathbf{F} = \mathbf{F}_{\mathrm{RF}} \mathbf{F}_{\mathrm{BB}}$ and $\mathbf{W} = \mathbf{W}_{\mathrm{RF}} \mathbf{W}_{\mathrm{BB}}$. The matrix $\mathbf{R}_n = \frac{1}{\mathsf{SNR}} \mathbf{W}^{\mathsf{H}} \mathbf{W}$ represents the noise covariance matrix where $\mathsf{SNR} = \frac{P_T}{N_S \sigma_n^2}$, and with P_T and σ_n^2 denoting the total transmit power and noise power.

Next, we assume that the RF beamforming/combining vectors are selected from pre-defined quantized codebooks, i.e., $[\mathbf{F}_{\mathrm{RF}}]_{:,n_t} \in \mathcal{F}, \forall n_t$ and $[\mathbf{W}_{\mathrm{RF}}]_{:,n_r} \in \mathcal{W}, \forall n_r$. If the channel is known, the hybrid precoder/combiner design problem can then be written as

$$\{\mathbf{F}_{\mathrm{BB}}^\star, \mathbf{F}_{\mathrm{RF}}^\star, \mathbf{W}_{\mathrm{BB}}^\star, \mathbf{W}_{\mathrm{RF}}^\star\} =$$
$$\arg\max \quad \log_2 \left| \mathbf{I} + \mathbf{R}_n^{-1} \mathbf{W}^{\mathsf{H}} \mathbf{H} \mathbf{F} \mathbf{F}^{\mathsf{H}} \mathbf{H}^{\mathsf{H}} \mathbf{W} \right|, \quad (3)$$
$$\text{s.t} \quad \mathbf{F} = \mathbf{F}_{\mathrm{BB}} \mathbf{F}_{\mathrm{RF}}, \qquad (4)$$
$$\mathbf{W} = \mathbf{W}_{\mathrm{BB}} \mathbf{W}_{\mathrm{RF}}, \qquad (5)$$
$$[\mathbf{F}_{\mathrm{RF}}]_{:,n_t} \in \mathcal{F}, \forall n_t, \qquad (6)$$
$$[\mathbf{W}_{\mathrm{RF}}]_{:,n_r} \in \mathcal{W}, \forall n_r, \qquad (7)$$
$$\|\mathbf{F}_{\mathrm{RF}} \mathbf{F}_{\mathrm{BB}}\|_F^2 = N_S, \qquad (8)$$

Further, if the RF beamforming./combining codebooks consist of orthogonal vectors (such as the case in the widely-adopted DFT codebooks), then for any selected RF precoders and combiners, $\mathbf{F}_{\mathrm{RF}}, \mathbf{W}_{\mathrm{RF}}$, the optimal baseband precoders/combiners are defined as [11]

$$\mathbf{F}_{\mathrm{BB}}^\star = \left(\mathbf{F}_{\mathrm{RF}}^{\mathsf{H}} \mathbf{F}_{\mathrm{RF}}\right)^{-\frac{1}{2}} \left[\overline{\mathbf{V}}\right]_{:,1:N_S}, \qquad (9)$$
$$\mathbf{W}_{\mathrm{BB}}^\star = \left[\overline{\mathbf{U}}\right]_{:,1:N_S}, \qquad (10)$$

where $\overline{\mathbf{V}}$ and $\overline{\mathbf{U}}$ are the right and left singular vector matrices of the effective channel matrix $\overline{\mathbf{H}} = \mathbf{W}_{\mathrm{RF}}^{\mathsf{H}} \mathbf{H} \mathbf{F}_{\mathrm{RF}}$. This reduces the hybrid precoding design problem to the following exhaustive search problem over the RF precoding/combining matrices

$$\{\mathbf{F}_{\mathrm{RF}}^\star, \mathbf{W}_{\mathrm{RF}}^\star\} = \underset{\substack{[\mathbf{F}_{\mathrm{RF}}]_{:,n_t} \in \mathcal{F}, \forall n_t \\ [\mathbf{W}_{\mathrm{RF}}]_{:,n_r} \in \mathcal{W}, \forall n_r}}{\arg\max} \log_2 \left| \mathbf{I} + \mathsf{SNR} \mathbf{W}_{\mathrm{RF}}^{\mathsf{H}} \mathbf{H} \mathbf{F}_{\mathrm{RF}} \right.$$
$$\left. \times \left(\mathbf{F}_{\mathrm{RF}}^{\mathsf{H}} \mathbf{F}_{\mathrm{RF}}\right)^{-1} \mathbf{F}_{\mathrm{RF}}^{\mathsf{H}} \mathbf{H}^{\mathsf{H}} \mathbf{W}_{\mathrm{RF}} \right|, \qquad (11)$$

The optimization problem in (11) implies that the optimal hybrid precoders can be found via an exhaustive search over the candidate RF beamforming/combining vectors. The challenge however is that the channel is normally unknown and its explicit estimation requires very large training overhead in mmWave systems [1]. To address this challenge, our objective in this paper is to devise a solution that directly finds the RF precoding/combining vectors of the hybrid architecture that maximize (or approach) the optimal achievable rate while requiring low channel training overhead. In the following section, we show how machine/deep learning tools can provide an efficient solution to this problem.
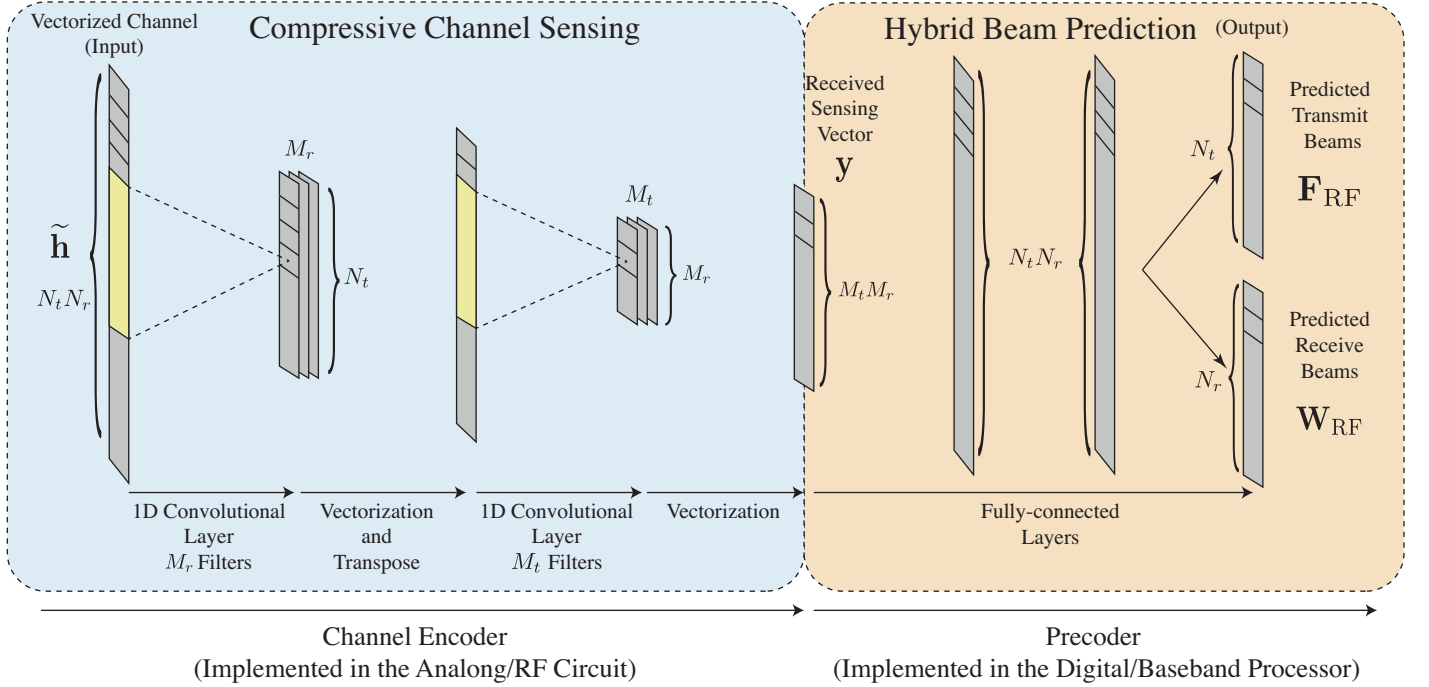
Fig. 2. The proposed *auto-precoder* neural network consists of two sections: the channel encoder and the precoder. The channel encoder takes the vectorized channel vector as an input and passes it through two complex-valued 1D-convolution layers that mimic the Kronecker product operation of the transmit and receive measurement matrices in (13). The output of the channel encoder (the received sensing vector) is then inputted to the precoder network. The precoder network consists of two fully connected layers and two output layers with the objective of predicting the indices of the N_t^RF RF beamforming vectors and N_r^RF RF combining vectors of the hybrid architecture. The dimensions of the two output layers are equal to the sizes of the RF beamforming and combining codebooks, $\mathbf{F}, \mathbf{W}$, i.e. the number of classes for the multi-label classification problem.

IV. MMWAVE AUTO-PRECODER: A NOVEL NEURAL NETWORK FOR DIRECT HYBRID PRECODING DESIGN

In classical (non machine learning) signal processing, the channel estimation and hybrid precoding design is normally done through three stages [2], [3]. First, leveraging the sparse nature of the mmWave channels, the channel is sensed using compressive measurements (that are normally random) [1], [3]. Then, the channel is reconstructed from the compressed measurements using, for example, basis pursuit algorithms. Finally, the constructed channel is used to design the hybrid precoding matrices. The main drawback of this approach is that it does not leverage the prior channel observations to reduce the training overhead associated with estimating the channels and designing the precoders. In this paper, we propose a novel neural network architecture that we call an 'auto-precoder' and a deep-learning approach that senses/compresses the channels and directly design the hybrid beamforming vectors from the compressed measurements. Our approach achieves multi-fold gains: (i) Different from the random measurements that are normally adopted in compressive channel estimation, our approach learns how to optimize the measurement (compressive sensing) vectors based on the user distribution and the surrounding environment to focus the measurement/sensing power on the most promising spatial directions. (ii) The deep learning model learns (and memorizes) how to predict the hybrid beamforming vectors directly from the compressed measurements. This design approach significantly reduces the training overhead while achieving near-optimal achievable

rates as will be shown in Section V. Next, we briefly explain the main concept of the proposed *auto-precoder* deep learning model in Section IV-A and then provide a detailed description of its two main stages, namely the channel sensing and hybrid beam prediction in Sections IV-B–IV-C.

A. mmWave Auto-Precoder: The Main Concept

In this paper, we propose the novel *auto-precoder* neural network architecture in Fig. 2. The auto-precoder neural network consists of two main sections: (i) the *channel encoder* which learns how to optimize the compressive sensing vectors to focus the sensing power on the most promising directions and (ii) the *precoder* which learns how to predict the RF beamforming/combining vectors of the hybrid architecture directly from the receive sensing vector; i.e., the output of the channel encoder. In order to achieve these objectives, the auto-precoder network is trained and used as follows.

- **Auto-Precoder Training:** In the training phase, the auto-precoder is trained end-to-end in a supervised manner. More specifically, a dataset of the mmWave channels and the corresponding RF beamforming/combining matrices are constructed and the auto-precoder is trained to be able to predict the indices of the RF precoding/combining vectors of the hybrid architecture given the input channel vector. In this paper, we use the near-optimal Gram-Schmidt hybrid beamforming algorithm in [11] to construct the RF beamforming/combining matrices. We note, however, that other hybrid beamforming algorithms can

be also adopted to construct the target precoders. It is important to mention here that through the end-to-end training of the auto-precoder model, the channel encoder in Fig. 2 learns in an unsupervised way how to optimize its compressive sensing vectors. This is thanks to the novel design of the channel encoder architecture that will be described in detail in Section IV-B.

- **Auto-Precoder Prediction:** After the auto-precoder network is trained, it is decoupled into two parts in the testing (prediction) phase: (i) the neural network of the channel encoder is directly implemented in the analog/RF circuits. More specifically, the weights of the two convolutional layers of the channel encoder network will be used as the weights of the analog/RF measurement matrices at both the transmitter and receiver. This is enabled by the specific design of the channel encoder network as will be explained in Section IV-B. These deep-learning optimized measurement matrices will be employed to sensing the unknown mmWave channel matrix. (ii) the output of the channel measurement, i.e., the receive sensing vector, will be inputted to the precoder network in the baseband/digital domain and used to directly predict the indices of the RF beamforming/combining vectors of the hybrid architecture.

It is worth mentioning here that we focus in this paper on predicting the RF precoding/combining matrices of the hybrid architecture as their design is what normally requires large training overhead. Once the RF precoders/combiners are designed, the effective channel will have small dimensions, $N_{\mathrm{r}}^{\mathrm{RF}} \times N_{\mathrm{t}}^{\mathrm{RF}}$, and can be easily trained with a few training pilots to design the baseband precoders/combiners. In the next two sections, we will describe the two components of the auto-precoders model in Fig. 2.

B. mmWave Compressive Channel Sensing

Leveraging the mmWave channel sparsity, [3] proposed to leverage compressive sensing tools to sense and reconstruct the mmWave channels with hybrid analog/digital transceivers. In this section, we develop a novel neural network architecture that implements the compressive sensing formulation in [3], and allows the neural network model to optimize the measurement vectors based on the surrounding environment and candidate user locations. More specifically, consider the system and channel models in Section II. Let $\mathbf{P}$ and $\mathbf{Q}$ denote the $N_{\mathrm{t}} \times M_{\mathrm{t}}$ and $N_{\mathrm{r}} \times M_{\mathrm{r}}$ channel measurement matrices adopted by both the transmitter and receiver to sense the channel $\mathbf{H}$, with M_{t} and M_{r} representing the number of transmit/receiver measurements. If the pilot symbols are equal to 1, then the received measurement matrix, $\mathbf{Y}$, can be written as [3]

$$\mathbf{Y} = \sqrt{P_{\mathrm{T}}} \mathbf{Q}^{\mathsf{H}} \mathbf{H} \mathbf{P} + \mathbf{Q}^{\mathsf{H}} \mathbf{V}, \qquad (12)$$

where $[\mathbf{V}]_{m,n} \sim \mathcal{N}_{\mathbb{C}}(0, \sigma_n^2)$ is the receive measurement noise. Now, when vectorizing this measurement matrix, $\mathbf{Y}$, we get

$$\mathbf{y} = \sqrt{P_{\mathrm{T}}} \left(\mathbf{P}^T \otimes \mathbf{Q}^{\mathsf{H}} \right) \mathbf{h} + \mathbf{v}_q, \qquad (13)$$

where $\mathbf{y} = \mathbf{vec}\left(\mathbf{Y}\right), \mathbf{v} = \mathbf{vec}\left(\mathbf{Q}^{\mathsf{H}}\mathbf{V}\right)$ and $\mathbf{h} = \mathbf{vec}\left(\mathbf{H}\right)$. In classical signal processing approaches (the do not lever-
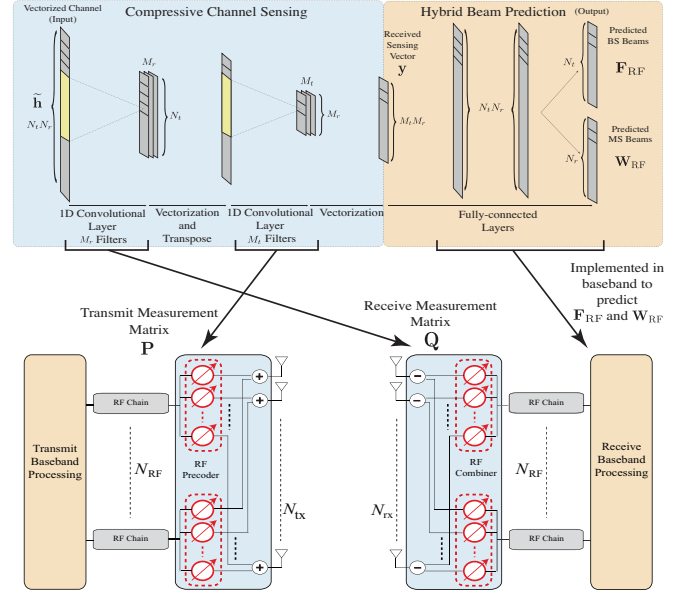

Fig. 3. This figure illustrates the key idea of the proposed joint channel sensing and hybrid beam prediction approach. The complex weights of the trained channel encoding network are used as the weights of the transmit/receive measurement vectors that are implemented in the RF (or using hybrid architecture). The precoder network is implemented in the baseband. It takes the output of the channel measurement vectors (the receive vector) and use it to predict the best transmit/receive beamforming vectors.

age machine learning), the transmitters and receivers do not normally make use of previous observations and ,therefore, they do not have knowledge about the most promising spatial directions for the channel measurements. As a result, classical compressive channel sensing approaches typically adopt random measurement vectors [4], [6]. Intuitively, however, given a certain environment (geometry, materials, user distribution, etc.), one would expect that these measurement vectors can be optimized based on the environment to improve the channel measurement performance. Next, we present a novel neural network architecture that mimics the joint transmitter/receiver channel sensing formulation in (13) and allows for an environment-based optimization of the transmitter and receiver measurement vectors.

Network Architecture of 'Channel Encoder': Looking at the channel sensing formulation in (13), we note that the product of the vectorized channel vector $\mathbf{h}$ and the two measurement matrices can be emulated by inputing this channel vector into a neural network consisting of two consecutive convolutional layers, as shown in Fig. 2. In this architecture, the first convolutional layer employs M_{r} kernels (filters). Each kernel has a size of N_{r} and a stride of N_{r}, and represent one receive measurement vector. More specifically, the weights of every kernel directly represent the entries of a receive measurement vector in $\mathbf{Q}$. The output of the first convolutional layer has M_r feature maps. The matrix of these feature maps is then vectorized and fed in to the second convolutional layer. Similar to the first layer, the second convolutional layer consists of M_{t} kernels implementing the transmit measurement matrix $\mathbf{P}$. It is important to note here that since the transmitter/receiver measurement weights are generally complex, we adopt the

complex-valued neural network implementation of the convolutional layers in [12].

It is important to note here that the end-to-end training of the auto-precoder (explained in Section IV-A) teaches this channel encoder in an unsupervised way to optimize its transmit/receive compressive channel measurement matrices (or kernel weights). Intuitively, this optimization adapts the measurement matrices to the surrounding environment and user distribution and focuses the sensing power on the promising spatial directions. After the model is trained, the kernels of the channel encoder will be directly employed as the measurement matrices by the transmitter and receiver to sense the unknown channels. The output of this channel measurement (the receive sensing vector $\mathbf{y}$) will be inputted to the precoder network in Fig. 2, which is the second section of the auto-precoder, to predict the RF beamforming/combining vectors.

C. Hybrid beam predictions

Given the received channel measurement vector $\mathbf{y}$, we leverage deep neural networks to learn the direct mapping function from the received vector $\mathbf{y}$ and the beamforming/combining vectors. For simplicity, we focus on predicting the RF beamforming/combining vectors $\mathbf{F}_{\mathrm{RF}}$ and $\mathbf{W}_{\mathrm{RF}}$. Note that finding these RF beamforming/combining vectors is what requires large training overhead in mmWave systems. Once the RF beamforming/combining vectors are designed, the low-dimensional effective channel $\mathbf{W}_{\mathrm{RF}}^{\mathsf{H}}\mathbf{H}\mathbf{F}_{\mathrm{RF}}$ can be easily estimated and used to construct the beseband precoders and combiners. Further, since the RF beamforming/combining vectors are selected from the quantized codebooks $\mathcal{F}, \mathcal{W}$, we formulate the problem of predicting the indices of the RF beamforming/combining vectors as a multi-label classification problem. Next, we briefly explain the adopted network architecture for the hybrid beam prediction.

Network Architecture of 'Precoder': To predict the indices of the RF beamforming/combining vectors from the receive measurement vector $\mathbf{y}$, we propose the 'precoder' neural network architecture in Fig. 2. This network consists of two fully connected layers and two output layers, and it is fed by the output of the 'channel encoder' network described in Section IV-B. Each fully-connected layer is followed by Relu activation and batch normalization. The network has two output layers; one predicts the indices of the transmit beamforming vectors and the other one predicts the indices of the receive combining vectors. The dimensions of the two layers are equal to the cardinalities of the transmit and receive codebooks, $\mathcal{F}, \mathcal{W}$, i.e., the number of candidate beamforming/combining vectors. Note that the number of candidate beams is also the number of classes for the multi-label classification problem.

The auto-precoder network is trained end-to-end in a Multi-Task Learning (MTL) manner [13], by considering the design of RF precoding and combining matrices of the hybrid architecture as two related tasks. This way, the network is trained to simultaneously optimize the two tasks which accounts for the dependence between the precoding and combining matrices.

The MTL strategy enables us to solve two related problems using one neural network.

D. Deep Learning Model Training and Prediction

As briefly highlighted in Sections IV-A-IV-C, the proposed auto-precoder network in Fig. 2 is trained end-to-end as a multi-task learning problem [13], which belongs to the supervised learning class. Essentially, we train the neural network based on a dataset of channel vectors and corresponding RF beamforming/combining vectors of the hybrid architecture. The target RF precoding/combining matrices are calculated based on the near-optimal Gram-Schmidt hybrid precoding algorithm in [11]. Following [14]–[17], the channel vectors are globally normalized such as the largest absolute value of the channel elements equals one. The labels for the transmit beamforming vectors (and similarly for the receive combining vectors) are modeled as $N_{\mathrm{t}}^{\mathrm{RF}}$-hot vectors, with ones at the locations that correspond to the indices of the target RF beamforming codewords (from the codebook $\mathbf{F}$).

For the loss function, we use the binary cross entropy for the multi-label classification that corresponds to each task (i.e., to the precoding and combining tasks). The total loss function is the arithmetic mean of the binary cross entropies of the two tasks since they are equally important for the entire hybrid precoding/combing design objective. To evaluation the prediction performance, we calculate the sample-wise accuracy of the predicted indices. Let $y_{\mathrm{true}}^{(i)}$ denote the set of label indices and $y_{\mathrm{pred}}^{(i)}$ denote the set of predicted indices. Then, the accuracy of sample i is defined as

$$a_i = \frac{\left| y_{\mathrm{true}}^{(i)} \cap y_{\mathrm{pred}}^{(i)} \right|}{\left| y_{\mathrm{true}}^{(i)} \right|}, \tag{14}$$

where $\cap$ represents the intersection of two sets and $\left| \cdot \right|$ is the cardinality of a set. Finally, the sample-wise accuracy is defined as

$$a = \frac{\sum_{i=1}^{N} a_i}{N_{\mathrm{samples}}}, \tag{15}$$

where N_{samples} is number of samples. Note that since the number of labels for each channel is a fixed number for both true labels and predicted labels, the two widely-adopted performance metrics in the multi-label classification problems, namely precision and recall, reduce to (14).

During this training process, the neural network architecture in Fig. 2 achieves two joint objectives: (i) It optimizes the transmitter/receiver measurement vectors (which are the weights of the two convolutional layers in the channel encoder network) in an unsupervised way to focus the sensing power on the most promising directions and (ii) it learns how to predict the RF beamforming/combining vectors of the hybrid architecture directly from the channel measurement vectors through the precoder network. Thanks to this design that leverages the prior observations in designing the channel measurement/sensing vectors and hybrid precoding/combining matrices, the proposed approach significantly reduces the required training overhead to design the hybrid precoding/combining matrices, as will be shown in the following section.

TABLE I
THE ADOPTED DeepMIMO DATASET PARAMETERS

DeepMIMO Dataset Parameters	Value
Activate BS	4
Activate users	From row R1200 to R1500
Number of BS Antennas	$M_x = 1$, $M_y = 64$, $M_z = 1$
Number of User Antennas	$M_x = 1$, $M_y = 64$, $M_z = 1$
Antenna spacing (in wavelength)	0.5
System bandwidth (in GHz)	0.5
Number of OFDM subcarriers	1024
OFDM sampling factor	1
OFDM limit	1
Number of paths	3

Fig. 4. The achievable rates of the proposed deep-leaning based joint channel sensing and hybrid precoding approach for different values of total transmit power. P_T. These rates are also compared to the upper bound that designs the hybrid precoding/combining matrices assuming perfect channel knowledge. This figure illustrates that the proposed solution can achieve near-optimal data rates while significantly reducing the training overhead, i.e. requiring a small number of channel measurements, M_t, M_r).

V. SIMULATION RESULTS

In this section, we evaluate the performance of the proposed deep-learning based direct hybrid precoding design approach using realistic 3D ray-tracing simulations.

A. Dataset and Training

Our dataset is generated using the publicly-available generic DeepMIMO [18] dataset with the parameters summarized in Table I. More specifically, we consider BS 4 in the street-level outdoor scenario 'O1' communicating with the mobile users from row R1200 to R1500 with an uplink setup. For simplicity, both the transmitter (mobile users) and receiver (base station) are assumed to employ 64 antennas with 3 RF chains each. For every user, we first construct the channel matrix using the DeepMIMO dataset generator. Then, random noise is added to the channel matrix. The noise power is calculated based on a bandwidth of 0.5GHz and receive noise figure of 5dB. The channel is also adopted to construct the target RF precoding/combining matrices of the hybrid architecture using the near-optimal Gram-Schmidt based hybrid precoding algorithm in [11]. the pair of the noisy channel and the corresponding codebook indices of the RF precoders/combiners is then considered as one data point in the dataset.

The constructed dataset is used to train the proposed auto-precoder neural network adopting the cross-entropy based loss function defined in Section IV-D. Our model is implemented using the Keras libraries [19] with a Theano [20] backend. We use the Adam optimizer with momentum 0.5, a batch size of 512, and a 0.005 learning rate.

B. Achievable Rates

To evaluate our deep-learning based direct hybrid precoding solution, we calculate the achievable rate using the predicted precoding/combining indices and compare it with the optimal rate. More specifically, for a given noisy channel measurement, we use the proposed auto-precoder neural network model in Section IV to predict the indices of the RF precoding/combining matrices of the hybrid architecture. Then, adopting the baseband precoding/combining design in (9)-(10) we calculate the achievable rate as defined in (2). This rate is then compared to the optimal rate that is achieved when the precoding/combining matrices are calculated as described in Section V-A with perfect channel knowledge.
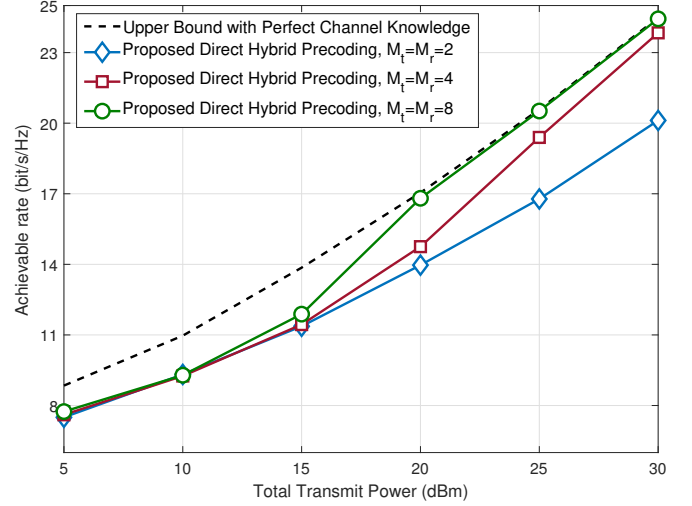
Fig. 4 illustrates the achievable rate of the proposed deep-learning based blind hybrid precoding approach and its upper bound versus different values of total transmit power. Further, to draw some insights into the required training overhead, we plot the achievable rate of the proposed direct hybrid precoding approach for three different values of the channel measurements, $M_\text{t} = M_\text{r} = 2, 4, 8$. This figure adopts the system and channel models described in Section V-A where both the transmitter (mobile user) and receiver (base station) employ a hybrid transceiver architecture with 64-element ULA and 3 RF chains. First, Fig. 4 shows that the performance of the proposed deep-learning based direct/blind hybrid precoding solution approaches the upper bound (which assumes perfect channel knowledge) at reasonable values of the transmit power. Further, and more interestingly, this figure illustrates the significant reduction in the required training overhead compared to classical (non-machine learning) solutions. For example, with only 4 channel measurements at both the transmitter and receiver, i.e., a total of $4 \times 4 = 16$ pilots, the proposed deep-learning based channel sensing/precoding solution achieves nearly the same spectral efficiency of the upper bound. This is instead of the $64 \times 64 = 4096$ pilots required for exhaustive search and almost one tenth of that, $\sim 300 - 400$, pilots needed by compressive sensing solutions [6]. **This significant reduction in the training overhead is thanks to the proposed deep learning model which leverages the prior observations to optimize the channel sensing based on the environment/user locations and to learn the mapping from the receive channel sensing vector and the optimal precoders/combiners.**

C. Sample-Wise Accuracy

In addition to the achievable rate (which is the main objective of this work), we also evaluate the performance

TABLE II
BEAM INDICES PREDICTION ACCURACY VERSUS SNR

P_T (dBm)	5	10	15	20	25	30
Tx acc. ($M_t=M_r=2$)	0.56	0.63	0.71	0.72	0.73	0.74
Rx acc. ($M_t=M_r=2$)	0.55	0.63	0.70	0.71	0.72	0.72
Tx acc. ($M_t=M_r=4$)	0.67	0.69	0.72	0.77	0.85	0.88
Rx acc. ($M_t=M_r=4$)	0.66	0.69	0.71	0.78	0.85	0.88
Tx acc. ($M_t=M_r=8$)	0.68	0.70	0.73	0.89	0.91	0.91
Rx acc. ($M_t=M_r=8$)	0.61	0.69	0.73	0.89	0.91	0.92

of the proposed deep learning model using the sample-wise accuracy defined in Section IV-D. The sample-wise accuracy evaluates the ability of the deep learning model in predicting the correct set of the RF beamforming/combining vectors for the hybrid architecture. In Table II, we adopt the same system and channel models considered in Fig. 4 and summarize the sample-wise accuracy of the transmit beamforming and receive combining beams for different values of total transmit power P_T. Similar to Fig. 4, Table II shows with only 4 or 8 channel measurements, the sample-wise accuracy of predicting the exact 3 transmit beams and 3 receive beams reaches nearly 90%. It is worth noting here that the accuracy for the transmit and receive predicted beams are almost the same since we treat them as equally important tasks when training the auto-precoder neural network model.

VI. CONCLUSION

In this paper, we developed a neural network architecture and a deep-learning approach for joint channel sensing and hybrid beamforming design in mmWave massive MIMO systems. The proposed neural network, that we called an auto-precoder, has two components: (i) The channel encoder which learns how to optimize the channel sensing vectors to focus the sensing power on the promising directions, and (ii) the precoder network which learns how to predict the RF beamforming/combining vectors of the hybrid architecture directly from the received sensing vector. Thanks to the specific design of the channel encoder that uses complex-valued neural networks and accounts for the constraint on the RF chains, the trained weights of the channel encoder are directly used as channel measurement vectors in the prediction phase. Simulation results showed that the proposed deep-learning based solution can successfully predict the hybrid beamforming vectors that achieve near-optimal data rates while requiring negligible training overhead compared to exhaustive search and classical compressive sensing solutions.

REFERENCES

[1] R. W. Heath, N. Gonzlez-Prelcic, S. Rangan, W. Roh, and A. M. Sayeed, "An overview of signal processing techniques for millimeter wave MIMO systems," *IEEE Journal of Selected Topics in Signal Processing*, vol. 10, no. 3, pp. 436–453, April 2016.

[2] A. Alkhateeb, J. Mo, N. Gonzalez-Prelcic, and R. Heath, "MIMO precoding and combining solutions for millimeter-wave systems," *IEEE Communications Magazine,*, vol. 52, no. 12, pp. 122–131, Dec. 2014.

[3] A. Alkhateeb, O. El Ayach, G. Leus, and R. Heath, "Channel estimation and hybrid precoding for millimeter wave cellular systems," *IEEE Journal of Selected Topics in Signal Processing*, vol. 8, no. 5, pp. 831–846, Oct. 2014.

[4] P. Schniter and A. Sayeed, "Channel estimation and precoder design for millimeter wave communications: The sparse way," in *the Asilomar Conference on Signals, Systems and Computers (ASILOMAR)*, Nov. 2014.

[5] J. Lee, G.-T. Gil, and Y. Lee, "Exploiting spatial sparsity for estimating channels of hybrid MIMO systems in millimeter wave communications," in *Global Communications Conference (GLOBECOM), 2014 IEEE*, Dec 2014, pp. 3326–3331.

[6] A. Alkhateeb, G. Leus, and R. Heath, "Compressed-sensing based multi-user millimeter wave systems: How many measurements are needed?" in *in Proc. of the IEEE International Conf. on Acoustics, Speech and Signal Processing (ICASSP), Brisbane, Australia, arXiv preprint arXiv:1505.00299*, April 2015.

[7] Y.-Y. Lee, C.-H. Wang, and Y.-H. Huang, "A hybrid RF/baseband precoding processor based on parallel-index-selection matrix-inversion-bypass simultaneous orthogonal matching pursuit for millimeter wave MIMO systems," *IEEE Transactions on Signal Processing*, vol. 63, no. 2, pp. 305–317, Jan 2015.

[8] O. El Ayach, S. Rajagopal, S. Abu-Surra, Z. Pi, and R. Heath, "Spatially sparse precoding in millimeter wave MIMO systems," *IEEE Transactions on Wireless Communications*, vol. 13, no. 3, pp. 1499–1513, Mar. 2014.

[9] T. Rappaport, S. Sun, R. Mayzus, H. Zhao, Y. Azar, K. Wang, G. Wong, J. Schulz, M. Samimi, and F. Gutierrez, "Millimeter wave mobile communications for 5G cellular: It will work!" *IEEE Access*, vol. 1, pp. 335–349, May 2013.

[10] G. MacCartney and T. Rappaport, "73 GHz millimeter wave propagation measurements for outdoor urban mobile and backhaul communications in New York City," in *Proc. of IEEE International Conference on Communications (ICC)*, June 2014, pp. 4862–4867.

[11] A. Alkhateeb and R. W. Heath, "Frequency selective hybrid precoding for limited feedback millimeter wave systems," *IEEE Transactions on Communications*, vol. 64, no. 5, pp. 1801–1818, May 2016.

[12] C. Trabelsi, O. Bilaniuk, Y. Zhang, D. Serdyuk, S. Subramanian, J. F. Santos, S. Mehri, N. Rostamzadeh, Y. Bengio, and C. J. Pal, "Deep Complex Networks," *arXiv e-prints*, p. arXiv:1705.09792, May 2017.

[13] S. Ruder, "An overview of multi-task learning in deep neural networks," *arXiv preprint arXiv:1706.05098*, 2017.

[14] A. Alkhateeb, S. Alex, P. Varkey, Y. Li, Q. Qu, and D. Tujkovic, "Deep learning coordinated beamforming for highly-mobile millimeter wave systems," *IEEE Access*, vol. 6, pp. 37 328–37 348, 2018.

[15] X. Li, A. Alkhateeb, and C. Tepedelenlioglu, "Generative adversarial estimation of channel covariance in vehicular millimeter wave systems," in *Proc of Asilomar Conference on Signals, Systems, and Computers*, Oct 2018, pp. 1572–1576.

[16] A. Taha, M. Alrabeiah, and A. Alkhateeb, "Enabling Large Intelligent Surfaces with Compressive Sensing and Deep Learning," *arXiv e-prints*, p. arXiv:1904.10136, Apr 2019.

[17] M. Alrabeiah and A. Alkhateeb, "Deep Learning for TDD and FDD Massive MIMO: Mapping Channels in Space and Frequency," *arXiv e-prints*, p. arXiv:1905.03761, May 2019.

[18] A. Alkhateeb, "DeepMIMO: A generic deep learning dataset for millimeter wave and massive MIMO applications," in *Proc. of Information Theory and Applications Workshop (ITA)*, San Diego, CA, Feb 2019, pp. 1–8.

[19] F. Branchaud-Charron, F. Rahman, and T. Lee, "Keras." [Online]. Available: https://github.com/keras-team

[20] J. Bergstra, F. Bastien, O. Breuleux, P. Lamblin, R. Pascanu, O. Delalleau, G. Desjardins, D. Warde-Farley, I. Goodfellow, A. Bergeron *et al.*, "Theano: Deep learning on gpus with python," in *NIPS 2011, BigLearning Workshop, Granada, Spain*, vol. 3. Citeseer, 2011, pp. 1–48.