Deep Learning at the Edge for Channel Estimation in Beyond-5G Massive MIMO

Mauro Belgiovine, Kunal Sankhe, Carlos Bocanegra, Debashri Roy, and Kaushik R. Chowdhury

Abstract

Massive multiple-input multiple-output (mMIMO) is a critical component in upcoming 5G wireless deployment as an enabler for high data rate communications. mMIMO is effective when each corresponding antenna pair of the respective transmitter-receiver arrays experiences an independent channel. While increasing the number of antenna elements increases the achievable data rate, at the same time computing the channel state information (CSI) becomes prohibitively expensive. In this article, we propose to use deep learning via a multi-layer perceptron architecture that exceeds the performance of traditional CSI processing methods like least square (LS) and linear minimum mean square error (LMMSE) estimation, thus leading to a beyond fifth generation (B5G) networking paradigm wherein machine learning fully drives networking optimization. By computing the CSI of all pairwise channels simultaneously via our deep learning approach, our method scales with large antenna arrays as opposed to traditional estimation methods. The key insight here is to design the learning architecture such that it is implementable on massively parallel architectures, such as GPU or FPGA. We validate our approach by simulating a 32-element array base station and a user equipment with a 4-element array operating on millimeter-wave frequency band. Results reveal an improvement up to five and two orders of magnitude in BER with respect to fastest LS estimation and optimal LMMSE, respectively, substantially improving the end-to-end system performance and providing higher spatial diversity for lower SNR regions, achieving up to 4 dB gain in received power signal compared to performance obtained through LMMSE estimation.

Introduction

Large antenna arrays are revolutionizing wireless communications and sensing, with manifestations in programmable surfaces, gesture monitoring, and high rate data delivery through incorporation in the form of massive multiple-input multiple-output (mMIMO) systems. Already envisaged as a key component of 5G, mMIMO utilizes a number of antennas that can be one to two orders of magnitude higher than the classical MIMO WiFi access points and LTE base stations (BSs) available today. However, despite the significant advances in edge computing capabilities, there are practical challenges in processing needs associated with such large antenna arrays. This article is motivated by our desire to decouple the scale of deployment with the limits of classical processing, especially as it pertains to the task of understanding the channel between a given antenna-receiver antenna-element pair for millimeter-wave (mmWave) communication. We accomplish this via training a deep learning (DL) architecture that offers the ability to produce a robust and high fidelity channel matrix between the mobile user and the mMIMO BS in a single forward pass. Since the overhead of the DL-based channel estimation becomes irrespective of the size of the antenna array, we believe this approach will enable a fundamental leap toward beyond 5G (B5G) standards where thousands of coordinated antennas will become the new norm. Emerging B5G networks are envisioned to support edge computing, which will enable rapid optimization and reconfiguration of the network architecture. This is a critical first step toward supporting requirements of emerging high-bandwidth and low-latency applications. Machine learning (ML) and artificial intelligence (AI) algorithms running at the edge computing servers help to (i) scale the optimization problem without proportional increase in complexity and (ii) enable fast response close to the BS, thus meeting strict demands of a time-varying wireless channel. We believe our use case of DL-enabled mmWave mMIMO demonstrates the need for tightly integrating AI into emerging wireless standards, which remains a gap even in the ongoing 5G rollout today.

Challenge in Channel Estimation

Channel estimation is the first step in the larger processing chain associated with decoding the data packet. Its objective is to identify the complex signal transformation imposed on the emitted wireless signal by the channel, and this is inferred via special information bits embedded in the packet preamble. For a spatially multiplexed system, this complex transformation is captured via the so-called channel state information (CSI). Knowing the CSI allows the transmitter to perform additional precoding functions that maximize the signal energy in the direction of interest. Thus, delayed computation of CSI, or worse, an incorrect computation can quickly degrade the performance in systems like mMIMO, where the CSI computation needs to be repeated several dozen times.

In the context of the B5G use case we explore in this article, we consider time-division duplexing (TDD) for mMIMO and assume that the channel varies slowly (coherence time of 10–100 ms [1]).

The authors are with Northeastern University, Boston.

Digital Object Identifier:
10.1109/MWC.001.2000322

1536-1284/21/$25.00 © 2021 IEEE

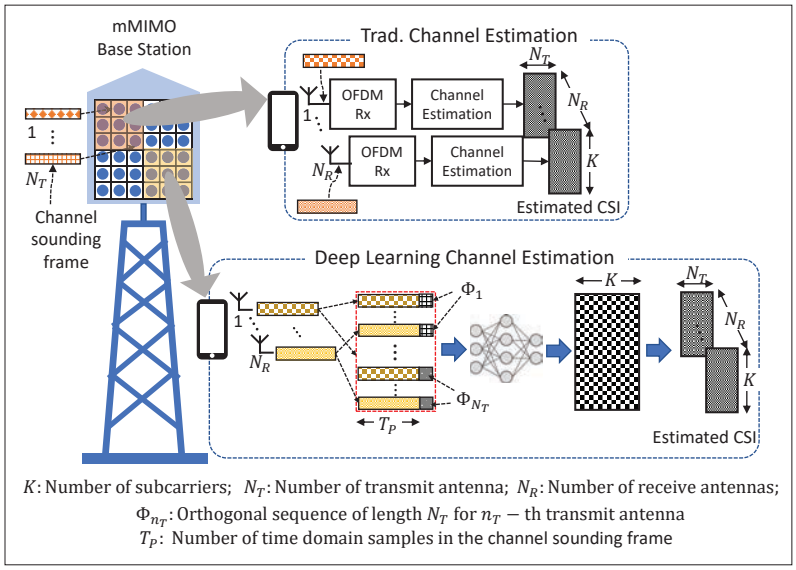

FIGURE 1. Overview of deep-learning-based channel estimation for B5G massive MIMO.

In this regime of operation, two phases involving the BS and user equipment (UE) precede downlink transmissions: *Channel Sounding*, in which the UE performs CSI estimation for the complete MIMO channel and sends it back to the BS, and *Data Transfer*, in which the BS uses the received CSI estimation to compute precoding weights for directional beams. Thus, the CSI estimation must be completed quickly in order to allow both the Channel Sounding and Data Transfer phases to be completed within the channel coherence time. Such a hard threshold on timeliness ensures that the BS can turn around its radio front-end and leverage channel reciprocity for the downlink transmission. Furthermore, by focusing on reducing the overhead associated with the CSI estimation step, it may be possible to reduce the Channel Sounding phase. This in turn will allow more data to be transferred in the given channel coherence time, ultimately increasing the overall throughput of the system.

Solution Overview

Our proposed approach of using DL aims to address the above issues by constructing a channel estimator that is able to obtain the complete MIMO channel matrix by processing the incoming preambles in a single forward pass, irrespective of the number of antenna elements involved in the system. For downlink, the BS sounds the channel by using a reference transmission, which allows the UE to estimate the channel using the proposed DL block. The UE transmits the channel estimation information back to the BS for calculation of the precoding needed for the subsequent data transmission. We generate the dataset in MATLAB, which we also release along with the simulation code to accelerate further research on this topic.

The Benefit of Deep Learning

Our goal is to leverage the massively parallel nature of a type of DL called deep neural networks (DNNs). Specifically, the key idea behind our proposed method is to estimate each of the sub-channels in the mMIMO channel matrix independent from each other. We do so by exploiting similarities in channel dynamics across spatial dimension and using an efficiently tuned DNN model whose weights are trained in order to be shared across the entire antenna array. Thus, we aim to retrieve the complete three-dimensional CSI matrix, where each dimension corresponds to the number of receiver antennas, the number of transmitter antennas, and the number of usable sub-carriers, by grouping all the received preambles in a single batch and processing it in a single forward step, as shown in Fig. 1. We design a compact multi-layer perceptron (MLP) with only three hidden layers to jointly exploit the hierarchical representational power of DNNs while keeping the execution time associated to its forward step low. To further reduce the computational burden associated with channel estimation, we train our model by taking as input the received time-domain preamble sequence, avoiding completely the prior demodulation step in orthogonal frequency-division multiplexing (OFDM) systems. The model is trained in a regression fashion in order to predict for each mMIMO sub-channel the CSI in the frequency domain for the complete set of OFDM pilot and data sub-carriers. This allows learning directly a mapping from the time-domain signal to its correspondent CSI in the frequency domain. The proposed DNN model architecture is presented later.

By training the model on true CSI values obtained at high signal-to-noise ratio (SNR) level, we observe that the proposed method generalizes well for low SNR scenarios and outperforms the practical least square (LS) estimation in terms of accuracy, while approaching or exceeding performance of linear minimum mean square error (LMMSE) and improving the end-to-end system performance in low SNR regimes, critical for frequencies above 6 GHz band such as mmWave or THz bands.

Moreover, to fully take advantage of this data-driven approach and increase robustness of the DL pipeline, we add a denoising training step, in which we apply controlled additional white Gaussian noise on the training samples.

Summary of Contributions

- We propose a deep-learning-based CSI estimation method for mMIMO that incurs a fixed computational cost, irrespective of the number of antenna elements, by exploiting the inherently parallel nature of DNNs.
- We discuss the limitations of traditional estimation techniques and compare the inference time complexity of the state of the art in DL-based channel estimation with the proposed approach, demonstrating its suitability for edge applications.
- We validate the performance of CSI estimation by simulating downlink transmissions between a BS with N_T = 32 uniform rectangular array (URA) antennas and a single UE equipped with N_R = 4 uniform linear array (ULA) antennas.
- By focusing on low SNR conditions, our denoising training approach allows better accuracy for CSI estimation, approaching or exceeding the end-to-end performance of an LMMSE estimator. Thus, our method matches one of the most accurate estimators for this problem, but eliminates the computational burden that limits the deployment of LMMSE.

Technology Limitations for B5G mMIMO

CSI characterizes how signals propagate through a wireless channel between the transmitter and receiver [2]. Thus, CSI matrices used in mMIMO capture the channel variations in the time and frequency domains. We consider an mMIMO-OFDM system intended for mmWave communications. The mMIMO channel is computed not only for each of the $N_R \times N_T$ pairs, but also for every sub-carrier, during the explicit Channel Sounding stage provisioned within the 5G standard. Incorrect computation of CSI matrices can degrade the beams formed between the mMIMO BS and UEs, resulting in increased bit error rate (BER) during data transmission [2]. Moreover, if CSI matrices are not computed in a timely manner (i.e., within the channel coherence time), it will adversely impact the following data transfer because the channel coefficients used for beamforming are already outdated.

Using hybrid mMIMO beamforming [3], the BS transmits channel sounding frames in parallel over all the N_T transmitter antennas. Each channel sounding frame, within the long-training field (LTF) sequence of the preamble, spans over L OFDM symbols with additional orthogonal mapping sequences employed to avoid interference. The receiver estimates the CSI matrix using the received signal, after OFDM demodulation and orthogonal demapping, using either LS estimation or LMMSE. LS estimation is a widely adopted channel estimator, as it requires only $\mathcal{O}(N_T N_R K)$ element-wise divisions for all antenna pairs, where K is the number of sub-carriers, and its computation is dominated by the OFDM demodulation step, which relies on fast Fourier transform (FFT) operation having complexity $\mathcal{O}(K \log K)$. Unfortunately, LS estimation suffers from noise distortion and high mean squared error (MSE), particularly at low SNR. LS estimation can be refined by computing the LMMSE [4] estimation, although it requires prior knowledge of channel and noise statistics and solving a linear system whose complexity grows as much as $\mathcal{O}(N_T N_R K^3)$ for MIMO systems due to a matrix inversion step performed on the channel correlation matrix. Therefore, finding fast and accurate ways to perform CSI estimation is crucial in mMIMO, especially as the number of antennas may grow to the order of thousands in B5G networks.

Related Works for DL in mMIMO

While ML- and DL-based architectures have been traditionally deployed in the image, video, speech, natural language processing, and healthcare [5] domains, there have also been efforts in solving challenging tasks in the RF domain, such as modulation recognition, radio identification [6], and network resource allocation. In the area of channel estimation, [7] presents an end-to-end OFDM symbol decoding method using MLP by treating a single-input single-output (SISO) channel model as a black box. In the context of mMIMO, [8] proposes a compressive method for generating CSI feedback based on encoder-decoder DL architecture.

Applying DL-based approaches for CSI estimation in mMIMO is still at a nascent stage. Due to the high dimensionality in mMIMO, especially when involving OFDM techniques, the majority of existing solutions use complex and deep architectures to estimate large channel matrices. These solutions treat the multi-dimensional input signal as a single entity and often require additional prior or post-estimation steps. Although use of very deep architectures is a growing trend, their complexity usually limits use in edge devices that are typically constrained in power and processing capability. Reference [9] uses convolutional neural networks (CNNs) to improve the quality of a coarse initial estimate of the channel matrix in a method called Tentative Estimation. To exploit adjacent sub-carrier frequency correlations, the coarse estimate matrices are concatenated in large input tensors and processed by a neural network consisting of 10 convolutional layers. Reference [10] proposes a 10-layer LDAMP architecture, based on the unfolding of an iterative D-AMP algorithm. As the estimated channel is treated as a noisy 2D image, each layer relies on an additional denoising CNN, which is 20 layers deep and used to update the channel estimated in the previous layer. Although CNNs are efficient in terms of number of parameters, the resulting complexity poses a challenge for deep architectures when deployed on edge devices. Therefore, the large CNNs in both [9, 10] have limitations in real-time implementations. In the context of single-carrier systems, [11] devises an uplink (UL) transmission for single-antenna users and multiple-antenna BSs using a six-layer MLP to first estimate direction of arrival (DoA) and then determine the channel for each user, by expressing the channel estimate as a function of DoA and solving an additional linear system of equations. Recently, [12] described an online training method based on the Deep Image Prior scheme, using a 6-layer architecture based on 1×1 convolutions and upsampling, which performs denoising of the received signal before a traditional LS estimation. Although the number of parameters is low, this method requires training the network during every transmission for thousands of epochs, without any guarantee that this step completes within the channel coherence time. For single-carrier solutions, K separate models should be trained and deployed to be applied in OFDM systems. Table 1 summarizes the time complexity of existing methods and compares how our proposed approach results in a much simpler model that is suitable for edge architectures.

Deep Learning Solution for mmWave mMIMO

Model Architecture

We design a compact DNN model to keep computation time low and train it to learn a joint approximation of OFDM demodulation and LS/LMMSE channel estimation methods. Figure 2 shows how the DNN model will replace the processing blocks associated with demodulation and channel estimation in a typical mMIMO Channel Sounding process. Different from the state of the art presented earlier, we design our training process to use the received time-domain waveform corresponding to the LTF obtained after synchronization as input to the model. This allows us to avoid completely the OFDM demodulation necessary for CSI estimation, further reducing the computation burden associated with this step for systems with large bandwidth that require a high number of sub-carriers. We let the model perform inference from the spectral components of the received time-domain signal, without performing demodulation explicitly. Through this approach, we design a DNN that learns the mapping from the time-domain LTF waveform to the desired CSI estimation in the frequency domain.

While ML- and DL-based architectures have been traditionally deployed in the image, video, speech, natural language processing, and healthcare domains, there have also been efforts in solving challenging tasks in the RF domain, such as modulation recognition, radio identification, and network resource allocation.

In order to reduce the size of the input and capture the effect of channel on amplitude and phase components of the received signals, we choose to treat independently the real and imaginary component of the input, and therefore we create two identical and independent specialized models accepting the real and imaginary components respectively, both in real valued format.

Method	Type of DL model	L	Inference complexity	OFDM	Additional comments
DOA estimation [11]	MLP	6	$\mathcal{O}(\sum_{l=1}^{L} N_l I_l + \mathcal{G})$	No	K models needed to operate on OFDM
Deep CNN [9]	CNN	10	$\mathcal{O}(K\mathcal{T} + N_T N_R \sum_{l=1}^{L} F_l N_{l-1} N_l)$	Yes†	
Beamspace mmWave [10]	LDAMP + CNN	10	$\mathcal{O}(\sum_{l=1}^{L} \mathcal{L} + L\sum_{c=1}^{20} W_c H_c F_c^2 N_{c-1} N_c)$	No	K models needed to operate on OFDM
Untrained DNN [12]	CNN + upsampling	6	$\mathcal{O}(E(W_1 H_1 N_0 N_k + \sum_{l=2}^{L} 2W_{l-1} 2H_{l-1} N_{l-1} N_k))$	Yes†	E has no upper bound
Proposed	MLP	3	$\mathcal{O}(\sum_{l=1}^{L} N_l I_l)$	Yes‡	

Notation: N_T = number of transmitter antennas, N_R = number of receiver antennas, K = number of sub-carriers, L = number of hidden layers, I_i = number of input features of layer i, N_i = number of neurons (or kernels, in the case of CNNs) in the ith layer, F_i = kernel size of the ith convolutional layer (assuming square kernels), W_i = width of input volume for the ith convolutional layer, H_i = height of input volume of the ith convolutional layer, E = number of epochs, $\mathcal{L}$ = complexity of the LDAMP layer (linear system) in [10], $\mathcal{T}$ = complexity of Tentative Estimation (linear system, including matrix multiplications and inversions) in [9], $\mathcal{G}$ = complexity of additional linear system needed to compute complex channel coefficients from DOA estimation (requires matrix inversion) in [11]. †: method requires OFDM demodulation; ‡: method does not require OFDM demodulation.

TABLE 1. A coarse computational complexity comparison between existing methods and proposed channel estimator.

In order to reduce the size of the input and capture the effect of channel on amplitude and phase components of the received signals, we choose to treat the real and imaginary component of the input independently; therefore, we create two identical and independent specialized models accepting the real and imaginary components, respectively, both in real valued format. The corresponding real valued output of each model is then recast back into a complex representation to produce the final channel estimation output. Since the two models are independent of each other, they could potentially run in parallel; hence, for the sake of simplicity, we refer to both models as incurring a common temporal processing overhead in the rest of the article. For our experiments, we rely on an MLP architecture that accepts an input $\mathcal{X}^{n_R,n_T}$ obtained by concatenating an LTF signal $\underline{y}^{n_R}$ arriving at a particular receiver antenna n_R, and the orthogonal coding sequence $\underline{\Phi}_{n_T}$, known to both transmitter and receiver, associated with a given transmit antenna n_T. The size of input tensor is $[T_P + T_C \times 1]$, where T_P is the number of symbols belonging to the LTF sequence in the time domain and T_C is the length of the coding sequence. The reason we concatenate these additional features to the input is as follows: For an LTF signal received at a given receiver antenna, we must recover all the channel states relative to each transmitter antenna. Without the orthogonal coding sequence, it would be impossible for the model to produce the channel states for a given n_R receiver antenna, as well as all the other n_T transmitter antenna pairs. This is because the input signal $\underline{y}^{n_R}$ would be completely identical for all these cases. On the other hand, the output of the proposed model $\underline{\hat{H}}_{DNN}^{n_R,n_T}$ is a tensor of size $[K \times 1]$, where K is the number of sub-carriers of the target system, and corresponds to the model prediction of the channel frequency response experienced between transmitter antenna n_T and receiver antenna n_R during propagation of the input LTF signal.

The configuration chosen for the MLP architecture has only 2 hidden layers, each with 1024 neurons and ReLU activation function, and an output layer using linear activation function for the regression task. We perform batch normalization after each hidden layer and add a Dropout layer with drop probability 15 percent between the first and second hidden layers to avoid overfitting. The proposed architecture is also depicted in Fig. 2. Therefore, to retrieve the complete MIMO channel matrix, it is possible to construct a single input batch of size $N_T \times N_R$ inputs to process in parallel all the necessary $\mathcal{X}^{n_R,n_T}$ inputs and produce as output the $K \times N_T \times N_R$ channel matrix.

The choice of MLP over other architectures, such as CNN, is due to its forward step reduced complexity, that is, it requires less operations.

For instance, if we consider each channel prediction independently, the computational complexity of a single fully connected layer is dominated by matrix-vector multiplication, which has a serial computation complexity of $\mathcal{O}(N_i I_i)$, where N_i is the number of neurons in the ith layer and I_i is the number of features input to it. On the other hand, convolution complexity is $\mathcal{O}(W_i H_i F_i^2 N_{i-1} N_i)$ (for notation, see Table 1) that, depending on input and output features arrangement, would incur a much higher number of operations and output features to be processed by a fully connected regression layer, as in our case. Although MLP has a simpler forward step, it can be further accelerated by taking advantage of massively parallel computing architectures — graphical processing units (GPUs) or field programmable gate arrays (FPGAs), so it is crucial to minimize N_i and I_i, besides the number of layers, in order to provide a fast and compact model.

Training Procedure

The MLP models are trained via a regression approach, using gradient descent optimization and MSE loss function in order to minimize the error between each individual CSI estimation predicted by the neural network and the *perfect* channel estimation for a given input signal, which we consider to be the output of a classic deterministic channel estimator (either LS or LMMSE) under ideal noiseless conditions.

The neural network is trained using the Adam optimization method with a learning rate of 10^{-4},

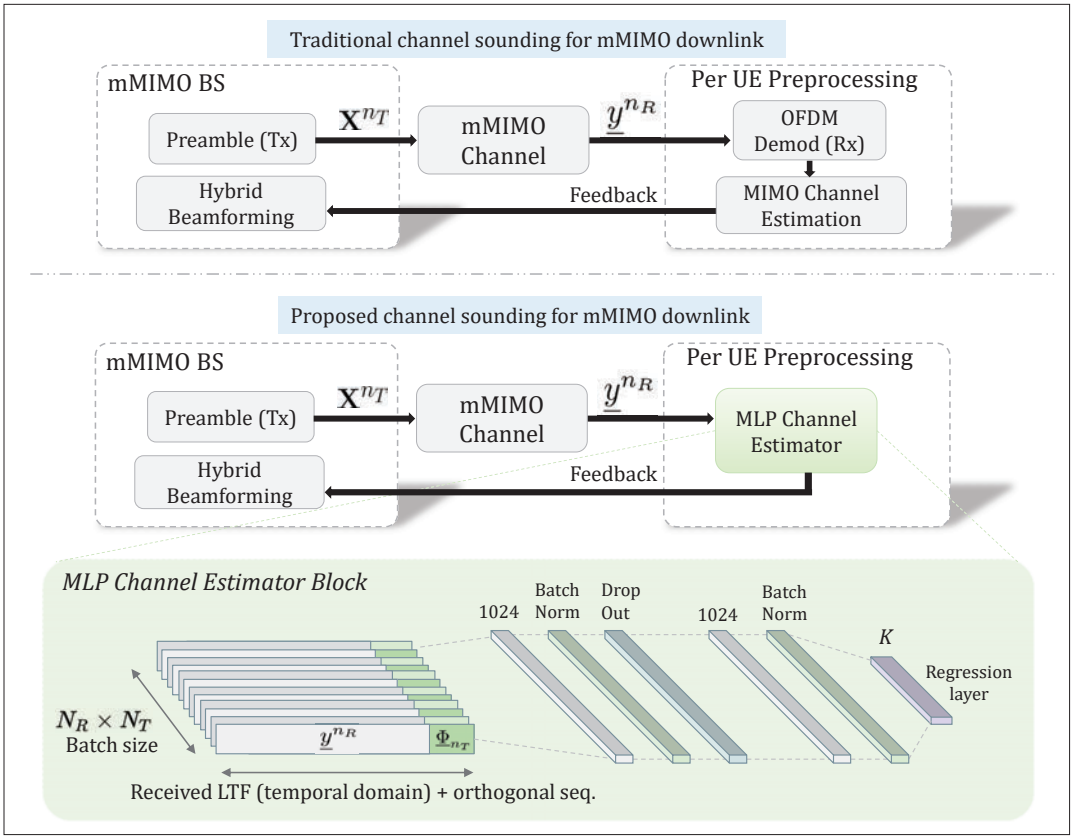

FIGURE 2. Classical and proposed channel sounding architectures for B5G mMIMO.

which is reduced by another factor of 10 when validation loss reaches a plateau for more than 15 epochs. Moreover, an early stopping criterion is adopted to terminate the training process if validation loss does not improve within the last 20 epochs.

Denoising Approach

As explained previously, since we want our model to be robust to noise variations at lower SNR levels, we incorporate a denoising approach. Specifically, during each training epoch, for each random mini-batch of input signals generated from the dataset, we augment the input data by applying additive white Gaussian noise (AWGN) with an increasing noise variance, reflecting the intended SNR range of the deployment scenario. For our experiments, we choose SNR levels of [–20, –10, 0, 10, 20, 30] dB during training. A predefined noise power is associated with each of these noise levels, based on the average power of all the received signals in the training dataset.

In this way, by only collecting low noise input samples, we are able to augment the data during training to effectively make the model more robust to different noise levels. The benefit of this approach is that we force our model to produce an output channel estimation that is close to the ideal noiseless one. This reduces errors in end-to-end transmissions due to poorly estimated channels under low SNR conditions.

Dataset Creation for Training/Testing

We use *Communication Toolbox* and *Phased Array Toolbox* within MATLAB™ to set up an mMIMO transmitter/receiver scenario. Specifically, we simulate a downlink end-to-end transmission from a BS equipped with N_T = 32 URA antennas and a UE with N_R = 4 ULA antennas, resulting in a 32 × 4 MIMO channel. Devices operate on a carrier frequency of 28 GHz, using 100 MHz bandwidth and FFT size of 256, resulting in 234 usable sub-carriers. We use a geometric scattering channel model without a line of sight (LoS) path with 100 scatterers that, for every transmission, are randomly placed on a spherical surface around the UE, which has a radius of 10 percent of the distance between UE and BS, while the position of UE and BS are assumed to be fixed, with a distance of 500 m. We generate Channel Sounding preamble frames, as explained earlier, having length $L = N_T$ OFDM symbols, and simulate transmission through a multi-path scattering channel model with N_s = 100 scatterers, as well as adding thermal noise. Since mmWave signals experience orders of magnitude more path loss than the microwave signals, the CSI computed at the receiver is used to compute precoding weights with orthogonal matching pursuit (OMP) [13], an algorithm that approximates optimal unconstrained precoders and combiners for a geometric scattering channel model, such that it can be implemented in low-cost RF hardware and operate under very low SNR scenarios.

Quadrature phase shift keying (QPSK) modulation is used at data transmission time. For training, we simulate 9000 complete transmissions, that is, including both Channel Sounding and Data Transfer phases, which are divided in 85 percent and 15 percent ratios for training and validation. In order to generate enough variation in channel realizations, we uniformly sample random seeds in the range $U[1, 10^7]$, used to generate unique channel states. For each transmission, we store the

Since we want our model to be robust to noise variations at lower SNR levels, we incorporate a denoising approach. Specifically, during each training epoch, for each random mini-batch of input signals generated from the dataset, we augment the input data by applying Additive White Gaussian Noise (AWGN) with an increasing noise variance, reflecting the intended SNR range of the deployment scenario.

The quality of CSI estimation, usually assumed perfect in the literature, impacts directly on the quality of the Data Transfer phase, as it forms the initial information from which the BS will compute precoding and combiner parameters. Hence, higher spatial diversity can be achieved when employing beamforming under accurate CSI estimation.

$\underline{y}^{nR}$ received LTF preambles, after the channel and noise application at each n_R receiver antenna and the relative channel estimation performed on the transmitter side, for all antennas and usable OFDM sub-carriers. In total, our training dataset consists of 1,152,000 LTF preambles.

To test our model under different noise conditions, we generate separate test datasets on a range of SNR levels, each composed of 500 transmissions. Due to the ability of the OMP precoding method to operate on extremely low SNR, we consider SNR ranging from −22 up to 10 dB. Since we want to assess the robustness of our model to noise variation, we add different levels of AWGN during data generation according to the desired SNR levels under which we want to test our model. The entire dataset and the software related to the data generation pipeline will be released for further use by the research community.

Performance Evaluation

In this section, we evaluate the performance of the proposed DL-based CSI estimation technique. First, the normalized MSE (NMSE) is used to measure the accuracy of the channel estimation. Second, the impact of model predictions is verified by means of bit error rate (BER) and beamforming gains, and later compared against traditional estimation techniques. We also present a discussion on how the proposed approach decouples the CSI channel estimation overhead from the antenna array dimensions.

Prediction Accuracy

First, we wish to assess the quality of the proposed channel estimator model under unseen channel conditions, and compare it to the ones obtained through traditional methods. Figure 3 depicts the NMSE of channel estimations for each method with respect to perfect (i.e., noiseless) estimation. Our method not only generalizes well on unseen channel conditions, but provides a high-quality estimation when the signal is corrupted by a large amount of noise power, approaching or even exceeding LMMSE accuracy in [−15, 10] dB SNR range, validating the robustness of our denoising training method against blocking and jamming effects.

The quality of CSI estimation, usually assumed perfect in the literature, impacts directly on the quality of the Data Transfer phase, as it forms the initial information from which the BS will compute precoding and combiner parameters. Hence, higher spatial diversity can be achieved when employing beamforming under accurate CSI estimation. Figure 4 shows how the proposed method, despite presenting presumably a higher NMSE in the lowest SNR regions, still outperforms optimal LMMSE estimation under all SNR levels considered, providing better quality channel estimation. Finally, Fig. 5 further proves how the channel estimation provided by the proposed approach is superior to optimal LMMSE in terms of end-to-end performance, providing zero BER starting from −19 dB, where LMMSE shows similar performance only starting from −17 dB.

Scalability

As stated before, with the proposed method every $N_T \times N_R$ channel estimation is performed independent from one another and using the same DNN model, by grouping the LTF incoming signal relative to each channel in a single input batch. This allows convenient scaling up to higher order mMIMO systems if massively parallel hardware accelerators are employed at inference time. Moreover, for those instances where a single forward pass cannot fit all the batch samples, our system allows the arrangement of smaller batches that could be processed in parallel using independent accelerators, that is, F accelerators provide a $\times F$ speedup increase compared to conventional streamlined systems. To give an idea of the effectiveness of the proposed method, our system estimates a full 32 × 4 mMIMO channel over 234 usable sub-carriers in $5.985 \cdot 10^{-4}$ s using an NVIDIA RTX 2080 Ti GPU, thus proving high accuracy and execution times below the envisioned channel coherence time for moderate mobility scenarios (i.e., 1–10 ms at 28 GHz frequency range).

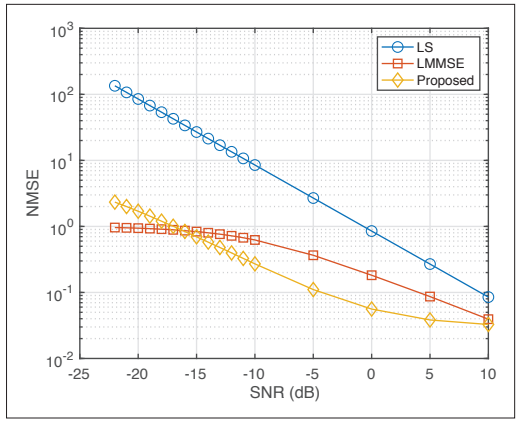

FIGURE 3. NMSE between each channel estimation method and ideal channel estimation. Note that as noise power increases, LMMSE output coefficients are close to 0 due to a large amount of noise corrupting the samples, so NMSE approaches 1 as SNR tends to −∞.

Open Research Challenges

Implementation of edge intelligence for emerging communication networks is still at the nascent stage with many open challenges:

- AI/ML-enabled channel estimation needs publicly available, representative datasets, where different types of pilots, channel conditions, antenna configurations, and scenarios are considered holistically. Thus, new tools are needed to generate and disseminate such datasets. The Platforms for Advanced Wireless Research (PAWR) [14] program has mMIMO BS installations that can be used for this.
- Real-time execution of channel estimation schemes need carefully designed edge computing architectures in the form of FPGAs. Thus, when limited training is also done on site using GPUs, there needs to be an automated pathway that takes the trained models to generate and test compatible FPGA code without human intervention.
- The design of deep architectures cannot be divorced from impact on inference time. Recently, several works on joint training and compression via pruning [15] have been demonstrated for RF applications. Furthermore, quantization of the weights speeds up FPGA processing.

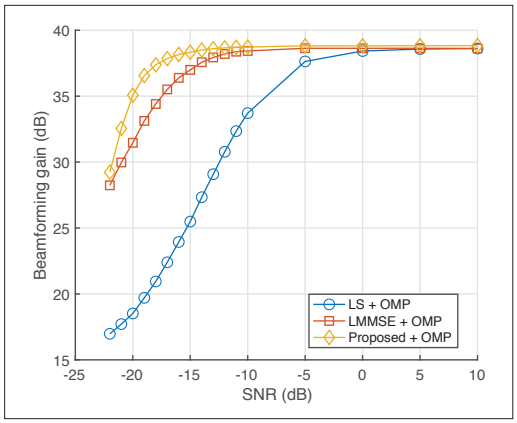

FIGURE 4. Gain in dB of received signal observed during Data Transfer phase (i.e., after beamforming) compared to signal power observed during Channel Sounding phase.

- The overall wireless environment changes with time and location. Hence, use of transfer learning and federated learning are needed to cope with scenarios not encountered at training time, including hardware updates like addition of more antennas, or software changes, such as new protocols that require different pilot arrangements.

Conclusion

We present a DL-based CSI estimation technique for massive MIMO antenna arrays, which will facilitate faster channel sounding for beyond 5G wireless networks. It will also achieve higher throughput for extremely low SNR scenarios, as is generally also applicable for mmWave and THz bands. The proposed DNN uses two hidden MLP layers and a linear output layer to jointly perform the task of OFDM demodulation and CSI matrix generation for mMIMO downlink transmission. We substantially improve the end-to-end system performance, achieving up to 5 and 2 orders of magnitude reduction in BER with respect to practical LS and optimal LMMSE, respectively, and higher spatial diversity for lower SNR regions, achieving up to 4 dB gain in received power signal compared to performance obtained through LMMSE estimation. Finally, we discuss the importance of model compression techniques to be applied on trained models in order to be easily deployed in edge devices, enabling higher data rates for edge computing over B5G mmWave communication.

Acknowledgments

This material is based on work supported by the Defense Advanced Research Projects Agency (DARPA) under Agreement No. HR00112090055 and the U.S. National Science Foundation under award no. CNS #1923789.

References

[1] S. Haghighatshoar and G. Caire, "Massive MIMO Channel Subspace Estimation from Low-Dimensional Projections," *IEEE Trans. Signal Processing*, vol. 65, no. 2, 2017, pp. 303–18.
[2] Y. Ma, G. Zhou, and S. Wang, "WiFi Sensing with Channel State Information: A Survey," *ACM Comp. Surv.*, vol. 52, no. 3, 2019.
[3] A. F. Molisch *et al.*, "Hybrid Beamforming for Massive MIMO: A Survey," *IEEE Commun. Mag.*, vol. 55, no. 9, Sept. 2017, pp. 134–41.
[4] V. Savaux and Y. Louët, "LMMSE Channel Estimation in OFDM Context: A Review," *IET Signal Processing*, vol. 11, no. 2, 2017, pp. 123–34.

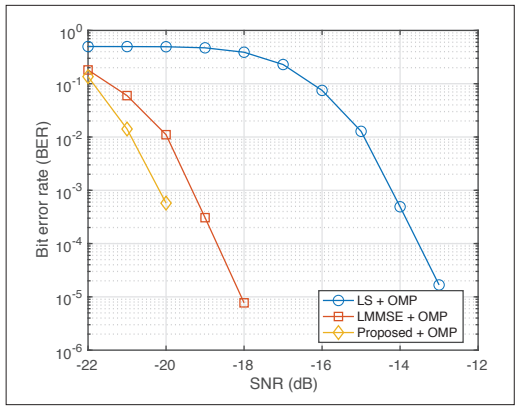

FIGURE 5. BER measured over different SNR level for LS estimation and proposed DNN-based method. Value not showed results in BER = 0.

[5] W. Taylor *et al.*, "An Intelligent Non-Invasive Real-Time Human Activity Recognition System for Next-Generation Healthcare," *Sensors*, vol. 20, no. 9, 2020, p. 2653.
[6] K. Sankhe *et al.*, "ORACLE: Optimized Radio clAssification through Convolutional NeuraL nEtworks," *IEEE INFOCOM 2019*, 2019, pp. 370–78.
[7] H. Ye, G. Y. Li, and B. Juang, "Power of Deep Learning for Channel Estimation and Signal Detection in OFDM Systems," *IEEE Wireless Commun. Letters*, vol. 7, no. 1, 2018, pp. 114–17.
[8] C. K. Wen, W. T. Shih, and S. Jin, "Deep Learning for Massive MIMO CSI feedback," *IEEE Wireless Commun. Letters*, vol. 7, no. 5, 2018, pp. 748–51.
[9] P. Dong *et al.*, "Deep CNN-Based Channel Estimation for mmWave Massive MIMO Systems," *IEEE J. Selected Topics in Signal Processing*, vol. 13, no. 5, 2019, pp. 989–1000.
[10] H. He *et al.*, "Deep Learning-Based Channel Estimation for Beamspace mmWave Massive MIMO Systems," *IEEE Wireless Commun. Letters*, vol. 7, no. 5, 2018, pp. 852–55.
[11] H. Huang *et al.*, "Deep Learning for Super- Resolution DOA Estimation in Massive MIMO Systems," *IEEE VTE-Fall 2018*, Aug. 2018, no. 9, pp. 8549–60.
[12] E. Balevi, A. Doshi, and J. G. Andrews, "Massive MIMO Channel Estimation with an Untrained Deep Neural Network," *IEEE Trans. Wireless Commun.*, vol. 19, no. 3, 2020, pp. 2079–90.
[13] O. E. Ayach *et al.*, "Spatially Sparse Precoding in Millimeter Wave MIMO Systems," *IEEE Trans. Wireless Commun.*, vol. 13, no. 3, 2014, pp. 1499–1513.
[14] A. Gosain, "Platforms for Advanced Wireless Research: Helping Define a New Edge Computing Paradigm," *Proc. 2018 Technologies for the Wireless Edge Wksp.*, 2018, p. 33.
[15] Z. Wang *et al.*, "Learn-Prune-Share for Lifelong Learning." *Proc. IEEE Int'l. Conf. Data Mining*, Nov. 2020.

Biographies

Mauro Belgiovine is pursuing his Ph.D. at the Electrical and Computer Engineering Department at Northeastern University, Boston, Massachusetts, under the guidance of Prof. Kaushik Chowdhury. His current research interests involve deep learning, wireless communication, and heterogeneous computing.

Kunal Sankhe is currently pursuing a Ph.D. degree in computer engineering at Northeastern University under the supervision of Prof. K. Chowdhury. His current research efforts are focused on implementing deep learning in the wireless domain and developing a cross-layer communication framework for the Internet of Things.

Carlos Bocanegra is a Ph.D. candidate working under the guidance of Prof. Kaushik R. Chowdhury at Northeastern University. He has experience in the areas of multi-antenna frameworks for centralized or distributed systems, machine learning for wireless applications, mobile sensing and computing, and coexistence of heterogeneous wireless systems.

Debashri Roy received her Ph.D. degree in computer science from the University of Central Florida. She is currently a postdoctoral fellow at Northeastern University. Her research interests are in the areas of experiential AI and ML in wireless communication.

Kaushik Roy Chowdhury [M'09, SM'15] is a professor at Northeastern University. His current research interests involve systems aspects of networked robotics, machine learning for agile spectrum sensing/access, wireless energy transfer, and large-scale experimental deployment of emerging wireless technologies.

The overall wireless environment changes with time and location. Hence, use of transfer learning and federated learning are needed to cope with scenarios not encountered at training time, including hardware updates like addition of more antennas, or software changes, such as new protocols that requires different pilot arrangements.