# Limited Feedback Double Directional Massive MIMO Channel Estimation: From Low-Rank Modeling to Deep Learning

Haoran Sun[1], Ziping Zhao[2], Xiao Fu[3], and Mingyi Hong[1]
[1] Dept. of Elect. & Compt. Eng., University of Minnesota, Minneapolis, MN 55455, USA.
[2] Dept. of Elect. & Compt. Eng., The Hong Kong University of Science and Technology, Kowloon, Hong Kong.
[3] School of Elect. & Compt. Sci., Oregon State University, Corvallis, OR 97331, USA.
Email: sun00111@umn.edu, ziping.zhao@connect.ust.hk, xiao.fu@oregonstate.edu, mhong@umn.edu

***Abstract*—In frequency division duplex massive MIMO systems, one critical challenge is that the mobiles need to feed back a large downlink channel matrix to the base station, creating large signaling overhead. Estimating a large downlink channel matrix at the mobile may also be costly in terms of power and memory consumption. Prior work addresses these issues using appropriate angle parameterization and compressed sensing techniques, but this approach involves solving a challenging, and sometimes extremely large, sparse inverse problem—which is difficult to solve to global optimality, and often leads to unaffordable memory and computational costs.**

**In this work, we propose an alternative framework that explores the fact that double directional channels for mmWave massive MIMO usually have low rank. The base station estimates the downlink channel via recovering a low-rank matrix, utilizing samples of the channel matrix compressed and fed back from the mobiles. This way, the mobile users can avoid performing resource-consuming tasks. In addition, the number of feedback measurements can be much smaller than the size of the channel matrix without losing channel recovery guarantees. Further, the low-rank estimation problem at the base station has a manageable size that scales gracefully with the channel size. Based on the new model, we propose two methods for channel estimation, which are based on iterative optimization and deep learning, respectively. Compared with the state-of-the-art, the optimization method obtains $10$x improvement and the deep learning approach achieves up to $1000$x improvement in computational complexity, while achieving high estimation quality in very low sample region.**

## I. Introduction

For frequency division duplex (FDD) massive MIMO systems, the forward and reverse links generally have highly uncorrelated channels and the channel reciprocity does not hold [1]. Then one serious challenge is that uplink/downlink channel estimation may incur large overhead since the mobile users need to feedback the channel matrix [2]. To address this issue, vector quantization (VQ) and codebook-based method are proposed [1]. However, to obtain high accuracy for massive MIMO systems, the codebook size can be huge [3]. Besides the potentially large feedback overhead, estimating a large downlink channel matrix itself may also be costly in terms of training overhead and computational resources (e.g., power and memory) at the mobile side [4]. Moreover, when it comes to three dimensional (3D) beamforming [5], the curse of dimensionality easily arises.

Prior work addresses these issues using angle parameterization of the channel matrix and compressed sensing techniques. The idea is to work with the double directional channel model that is often adopted for mmWave massive MIMO, which represents the channel matrix using the directions of arrivals (DoAs), directions of departures (DoDs), and pathlosses of a small number of paths. A sparse optimization formulation is then built by representing the vectorized channel matrix in an over-complete dictionary consisting of atoms that are associated with discretized angles and antenna array geometry. A sparse inverse problem is then solved either at the mobile station (MS) [6], [7] or at the base station (BS) [8] to recover the downlink channel. However, these methods involve solving difficult, and sometimes impossibly huge inverse problems, resulting in huge memory requirement and long processing time. This is especially true in estimating high-dimensional channels like 3D channels. Besides the high dimensionality issue, the dictionary is also heavily affected by the array geometries of the mobile users, which may change significantly from mobile to mobile, causing flexibility issues.

In this work, we propose a novel channel estimation approach based on a new low-rank model of the double-directional massive MIMO channel. The idea is to model the downlink channel with a few paths as a low-rank channel, and to recover such channels at the BS by using a small number of scalar measurements sent from the mobile terminals. In the proposed framework, the mobile users only perform random compression of the received pilot signals and send back to the BS—which is very economical in terms of resource consumption at the mobile end. Instead of resorting to angle parameterization, what we utilize is merely the low-rank structure of the double directional channel model, which is flexible enough to cover different array geometries and both the 2D and 3D channels, *in a unified way without changing the problem size*. This is very different from the angle parameterization based approaches in [6]–[8].

Further, to solve the low-rank model at the BS, we propose two machine learning based approaches, one based on the popular low-rank matrix sensing technique that has been often used in machine learning (but it is yet to be popularized in wireless communications), and the other is based on deep

978-1-5386-3512-4/18/$31.00 ©2018 IEEE

learning (which has been widely used for processing images, video, speech, and audio [9]). Unlike many existing learning-based approaches in communications which aim at learning an algorithms [10] or a classifier [11], [12], in this work, we propose to use a deep neural network (DNN) to directly learn a *complex-valued*, *low-rank* model. The main idea is that if a DNN can be effectively trained to learn the nonlinear inverse relationship from the feedback samples to the true (complex) low-rank channel, then in the testing stage, such a network can be directly used to produce high-quality channel estimate, with significant speed-up compared with traditional optimization-based approach. Further, it is important to note that, rather than learning the behavior of a given optimization method as has been proposed in [10], our approach directly trains a DNN from the ground truth. The channel estimators obtained in this way exhibit very different characteristics compared with the optimization-based approaches – indeed, we found that compared with the low-rank matrix sensing based algorithm, the DNN based algorithm is able to accurately estimate the channel with much smaller number of samples.

## II. Limited feedback channel estimation

### A. System Model

Consider a double directional downlink massive MIMO system [5], which consists of one BS with a uniform rectangular array (URA) of size $M_t^x \times M_t^y$ and one MS with a URA of size $M_r^x \times M_r^y$. There are in total $M_t = M_t^x M_t^y$ transmit antennas for BS and $M_r = M_r^x M_r^y$ receive antennas for MS.

The limited feedback-based channel estimation model can be described as follows. The BS first sends a number of pilot signals $\mathbf{s}_i \in \mathbb{C}^{M_t}, i = 1 \dots N$ to the MS and the received signal $\mathbf{y}_i \in \mathbb{C}^{M_r}$ at MS is given as follows:

$$\mathbf{y}_i = \mathbf{H}\mathbf{s}_i + \mathbf{n}_{t,i}, \quad i = 1, \dots, N \tag{1}$$

where $\mathbf{H} \in \mathbb{C}^{M_r \times M_t}$ denotes the channel matrix and $\mathbf{n}_{t,i} \in \mathbb{C}^{M_r}$ is the noise, which is assumed to be the additive white Gaussian noise (AWGN).

Many existing works propose to use the MS to estimate $\mathbf{H}$ and send back the estimated $\mathbf{H}$ through uplink signaling channels [6]. This could be very costly in two aspects: First, estimating a large $\mathbf{H}$ could be energy and memory consuming for the mobiles. Second, the uplink overhead can be large, depending on which parametrization of the channel is adopted. In this work, we consider a channel estimation strategy in which the mobile performs very simple operations in order to save resources (a strategy that has a similar design goal can be seen in [8]). Specifically, after receiving the pilot signal, the MS just compresses the signal $\mathbf{y}_i$ to a scalar measurement $z_i$, and sends it back to the BS. The compressed signal $z_i$ is given by

$$z_i = \mathbf{a}_i^T \mathbf{H}\mathbf{s}_i + \mathbf{a}_i^T \mathbf{n}_{t,i} + \mathbf{n}_{r,i} := \mathbf{a}_i^T \mathbf{H}\mathbf{s}_i + \mathbf{n}_i, \tag{2}$$

where $\mathbf{a}_i \in \mathbb{C}^{M_r}$ denotes the $i$-th compression vector, $\mathbf{n}_{r,i} \in \mathbb{C}$ denotes the AWGN, and $\mathbf{n}_i := \mathbf{a}_i^T \mathbf{n}_{t,i} + \mathbf{n}_{r,i}$. The BS then collects a number of $z_i$'s to estimate the double directional downlink channel $\mathbf{H}$. An illustration of the above process is given in Fig. 1.

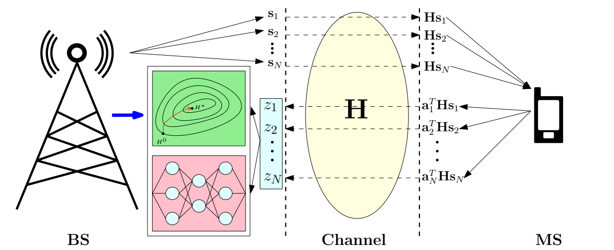


Fig. 1. The limited feedback channel estimation model. The yellow part represents the MIMO channel under consideration; the green and red boxes represent two algorithms proposed in this work, and both runs at the BS side.

In this work, we use the 3D double directional channel model, which contains $K$ paths from the transmitter to receiver array of antennas, where we assume that $K \ll \min\{M_r, M_t\}$ [8], [13]. Based on this assumption, the channel model can be expressed as follows [5]:

$$\mathbf{H} = \mathbf{A}_r(\boldsymbol{\theta}_r, \boldsymbol{\phi}_r)\mathrm{diag}(\boldsymbol{\beta})\mathbf{A}_t^H(\boldsymbol{\theta}_t, \boldsymbol{\phi}_t), \tag{3}$$

where $\boldsymbol{\theta}_r$ and $\boldsymbol{\phi}_r$ denote the elevation and azimuth angle vector of the DoAs, respectively; $\boldsymbol{\theta}_t$ and $\boldsymbol{\phi}_t$ denote the DoDs; $\mathbf{A}_r(\boldsymbol{\theta}_r, \boldsymbol{\phi}_r) \in \mathbb{C}^{M_r \times K}$ and $\mathbf{A}_t(\boldsymbol{\theta}_t, \boldsymbol{\phi}_t) \in \mathbb{C}^{M_t \times K}$ denote the receive antenna manifold and the transmit antenna manifold respectively; the path loss matrix $\mathrm{diag}(\boldsymbol{\beta})$ is a diagonal matrix with $\boldsymbol{\beta} = [\beta_1, \beta_2, \cdots, \beta_K]^T \in \mathbb{C}^K$ being its diagonal. It is easy to see that $\mathbf{H}$ is low-rank with rank($\mathbf{H}$) $= K$.

The receive antenna manifold $\mathbf{A}_r(\boldsymbol{\theta}_r, \boldsymbol{\phi}_r)$ is generated by a URA with matrices $\mathbf{A}_r^x(\boldsymbol{\theta}_r, \boldsymbol{\phi}_r) \in \mathbb{C}^{M_r^x \times K}$ and $\mathbf{A}_r^y(\boldsymbol{\theta}_r, \boldsymbol{\phi}_r) \in \mathbb{C}^{M_r^x \times K}$ as follows:

$$\mathbf{A}_r(\boldsymbol{\theta}_r, \boldsymbol{\phi}_r) = \mathbf{A}_r^y(\boldsymbol{\theta}_r, \boldsymbol{\phi}_r) \odot \mathbf{A}_r^x(\boldsymbol{\theta}_r, \boldsymbol{\phi}_r), \tag{4}$$

where "$\odot$" denotes the Khatri-Rao product (i.e., column-wise Kronecker product "$\otimes$"). For the $i$-th column of $\mathbf{A}_r(\boldsymbol{\theta}_r, \boldsymbol{\phi}_r)$ denoted by $\mathbf{a}_{r,i}(\boldsymbol{\theta}_r, \boldsymbol{\phi}_r)$, we have $\mathbf{a}_{r,i}(\boldsymbol{\theta}_r, \boldsymbol{\phi}_r) = \mathbf{a}_{r,i}^y(\boldsymbol{\theta}_r, \boldsymbol{\phi}_r) \otimes \mathbf{a}_{r,i}^x(\boldsymbol{\theta}_r, \boldsymbol{\phi}_r)$, where $\mathbf{a}_{r,i}^x(\boldsymbol{\theta}_r, \boldsymbol{\phi}_r)$ and $\mathbf{a}_{r,i}^y(\boldsymbol{\theta}_r, \boldsymbol{\phi}_r)$ are given by

$$\begin{aligned}
&\mathbf{a}_{r,i}^x(\boldsymbol{\theta}_r, \boldsymbol{\phi}_r) \\
&= [1, e^{j\pi \sin(\theta_{r,i})\cos(\phi_{r,i})}, \dots, e^{j\pi(M_r^x - 1)\sin(\theta_{r,i})\cos(\phi_{r,i})}]^T, \\
&\mathbf{a}_{r,i}^y(\boldsymbol{\theta}_r, \boldsymbol{\phi}_r) \\
&= [1, e^{j\pi \sin(\theta_{r,i})\sin(\phi_{r,i})}, \dots, e^{j\pi(M_r^y - 1)\sin(\theta_{r,i})\sin(\phi_{r,i})}]^T,
\end{aligned}$$

Similarly, the transmit antenna manifold $\mathbf{A}_t(\boldsymbol{\theta}_t, \boldsymbol{\phi}_t)$ can be generated by $\mathbf{A}_t^x(\boldsymbol{\theta}_t, \boldsymbol{\phi}_t)$ and $\mathbf{A}_t^y(\boldsymbol{\theta}_t, \boldsymbol{\phi}_t)$ with the DoD parameters $\boldsymbol{\theta}_t$ and $\boldsymbol{\phi}_t$.

### B. Prior Art: The Sparsity-Based Approach

In the literature, the structure of the channel matrix has been leveraged by transforming it to certain sparse models [8]. The idea can be summarized as follows. Suppose that the channel $\mathbf{H}$ takes the form in (3), then it can be approximated by

$$\mathbf{H} \approx \mathbf{D}_r \mathbf{G} \mathbf{D}_t^H. \tag{5}$$

In the above expression, $\mathbf{D}_r \in \mathbb{C}^{M_r \times Q^2}$ and $\mathbf{D}_t \in \mathbb{C}^{M_t \times Q^2}$ stand for an over-complete dictionary for the matrices $\mathbf{A}_r(\boldsymbol{\theta}_r, \boldsymbol{\phi}_r)$ and $\mathbf{A}_t(\boldsymbol{\theta}_t, \boldsymbol{\phi}_t)$ in the discretized angle space, which comply with geometries of the receive and transmit arrays, respectively; $\mathbf{G} \in \mathbb{R}^{Q^2 \times Q^2}$ is the matrix specifying active paths. The dictionary matrices are given by

$$\mathbf{D}_r = \mathbf{D}_r^y \otimes \mathbf{D}_r^x, \quad \text{and} \quad \mathbf{D}_t = \mathbf{D}_t^y \otimes \mathbf{D}_t^x, \tag{6}$$

where $\mathbf{D}_t^x \in \mathbb{C}^{M_t^x \times Q}$, $\mathbf{D}_t^y \in \mathbb{C}^{M_t^y \times Q}$, $\mathbf{D}_r^x \in \mathbb{C}^{M_r^x \times Q}$, and $\mathbf{D}_r^y \in \mathbb{C}^{M_r^y \times Q}$ are also over-complete quantized dictionary matrices for $\mathbf{A}_t^x(\boldsymbol{\theta}_t, \boldsymbol{\phi}_t)$, $\mathbf{A}_t^y(\boldsymbol{\theta}_t, \boldsymbol{\phi}_t)$, $\mathbf{A}_r^x(\boldsymbol{\theta}_r, \boldsymbol{\phi}_r)$, and $\mathbf{A}_r^y(\boldsymbol{\theta}_r, \boldsymbol{\phi}_r)$, respectively. In the above expressions $Q$ stands for the total number of quantization bins for both elevation and azimuth angles. The channel estimation problem at the BS can be formulated as

$$\begin{aligned} \underset{\mathbf{G}}{\text{minimize}} \quad & \sum_{i=1}^{N} |z_i - \mathbf{a}_i^T \mathbf{D}_r \mathbf{G} \mathbf{D}_t^H \mathbf{s}_i|^2 \\ \text{subject to} \quad & |\text{supp}(\mathbf{G})| \le L, \end{aligned} \tag{7}$$

where $L$ denotes the desired sparsity level of $\mathbf{G}$. Such a problem can be solved by many algorithms such as the orthogonal matching pursuit (OMP) method [14] and the iterative hard thresholding algorithm [15]; see [8] for how these methods can be applied to solve (7) [1].

However, to have a good approximation resolution, the dictionary matrices are usually of very high dimension. For example, a one degree quantization (ranging from $-90°$ to $90°$), $Q = 180$, and the matrix $\mathbf{D}_t \in \mathbb{C}^{M_t \times 32{,}400}$, $\mathbf{D}_r \in \mathbb{C}^{M_r \times 32{,}400}$, and $\mathbf{G} \in \mathbb{C}^{32{,}400 \times 32{,}400}$ are all of large size. In [8], the OMP method is used to solve the problem which requires a much larger dictionary of size $\mathbb{C}^{N \times 1{,}049{,}760{,}000}$, making the sparse recovery problem very hard. Additionally, since different mobile phones may have very different array geometries, the corresponding dictionary matrices could be different across the mobiles. This fact can further complicate the implementation of the above algorithm.

## III. The Low-rank sensing-based approach

In this section, we propose a low-rank based channel estimation approach to resolve the issue of curse of dimensionality arising from the sparse models. Directly leveraging on the low-rank property of the channel matrix (3), we formulate a limited feedback low-rank matrix sensing problem as below:

$$\begin{aligned} \underset{\mathbf{H}}{\text{minimize}} \quad & \sum_{i=1}^{N} |z_i - \mathbf{a}_i^T \mathbf{H} \mathbf{s}_i|^2 \\ \text{subject to} \quad & \text{rank}(\mathbf{H}) \le K, \end{aligned} \tag{8}$$

which is a mean squared error (MSE) minimization problem with a rank constraint.

Compared to problem (7), our new formulation is more natural and practical because it requires no prior knowledge on the array geometries and can avoid the issues resulted from the fixed dictionary, because the problem size no longer grow with improved quantization accuracy of the model (in fact, no quantization is needed). Moreover, it gets rid of the storage and computational burdens incurred by the high-dimensional dictionary matrices. Note that according to the low-rank matrix sensing theory [16], the number of feedback measurements can be substantially smaller than $M_r M_t$ without losing recovery guarantees of $\mathbf{H}$ [specifically, the number of feedback measurements for guaranteed channel recovery is in the order of $\mathcal{O}((M_r + M_t)K)$, which can be much smaller than $M_r M_t$ since the number of significant paths is usually rather small], and thus economical feedback is within reach.

At first glance, problem (8) is hard since it has a non-convex rank constraints. However, since the pilot and compressing vectors are under the designer's control, i.i.d. zero-mean and unit-variance Gaussian $\mathbf{a}_i$ and $\mathbf{s}_i$ can be used. This way, the problem falls into the category of low-rank sensing with the compressing operator satisfying the restricted isometry property (RIP). Therefore many efficient and globally optimal algorithms can be applied to solve problem (8) and its convex counterparts [17].

We propose to use the projected gradient (PGD) method [18], which is presented in Algorithm 1. This algorithm has many favorable properties. First, it guarantees the recovery of the ground truth $\mathbf{H}$ with high probability when $\mathcal{O}(MK \log M)$ measurements are used (assuming $M = M_r = M_t$) and converges at a linear rate to the desired solution; see details in [18]. It is also a first-order method that is relatively easy to implement (only matrix-vector multiplication and SVDs are involved). It is important to note that the fact problem (8) can be indeed solved to *global optimality* is an exciting feature of the proposed model. This is in stark contrast to the existing limited feedback sparse modeling like those in [8], which do not have any theoretical guarantee of the solution quality, due to highly structured dictionaries which lacks any randomness.

**Algorithm 1** PGD for Low-Rank Channel Estimation (8)

**Require:** $z_i, \mathbf{a}_i, \mathbf{s}_i$ ($i = 1, 2, \dots, N$), and $K$
1: $\mathbf{P} = [\mathbf{s}_1 \otimes \mathbf{a}_1; \mathbf{s}_2 \otimes \mathbf{a}_2; \dots; \mathbf{s}_N \otimes \mathbf{a}_N]$
2: $\alpha = 1/\sigma_{\max}^2(\mathbf{P})$
3: set $k = 0$ and $\mathbf{H}^0$
4: **repeat**
5: $\quad \tilde{\mathbf{H}}^{k+1} = \mathbf{H}^k + 2\alpha \sum_i^N \mathbf{a}_i^*(z_i - \mathbf{a}_i^T \mathbf{H}^k \mathbf{s}_i)\mathbf{s}_i^H$
6: $\quad [\mathbf{U}, \boldsymbol{\Sigma}, \mathbf{V}] = \text{svd}(\tilde{\mathbf{H}}^{k+1})$
7: $\quad \mathbf{H}^{k+1} = \mathbf{U}(:, 1:K)\boldsymbol{\Sigma}(1:K, 1:K)\mathbf{V}(:, 1:K)^H$
8: $\quad k \leftarrow k + 1$
9: **until** convergence

## IV. The DNN-based approach

In the previous section, a low-rank sensing based problem is formulated for the limited feedback channel estimation problem, and then efficiently solved by a numerical optimization algorithm. However, there are two potential drawbacks: i) Fundamental limit on sample complexity: the estimation performance could be bad when there are insufficient samples; ii) Computational complexity: potentially a large number of iterations are needed in order to get a desired solution, which makes the optimization procedure time-consuming.

To address these issues, a deep learning based method is further proposed. The DNN can potentially learn a mapping from the measurements to the channel matrix off-line, so during the on-line testing stage it is possible to use a small

[1] Note that the work in [8] considered an even harder problem than the above, where only the sign information of $\mathbf{a}_i^T \mathbf{D}_r \mathbf{G} \mathbf{D}_t^H \mathbf{s}_i$ is fed back to the BS for further reduction of the overhead. However, we leave this out of this paper and focus on the basic ideas.

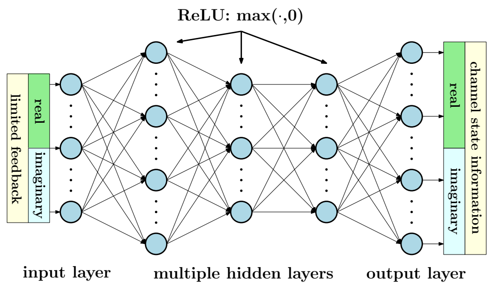


Fig. 2. The DNN structure used in this work.

number of samples to obtain highly accurate solutions. The philosophy behind this method is that by directly training the DNN using the ground-truth channel matrices and its random compressions using massive amount of training samples, the network is expected to "discover" the underlying relationship between the observations and the ground truth channel matrix. This could help substantially reduce the number of measurements needed in the testing stage (as long as the observations and ground truth channels come from a similar distribution as in the training), compared to the optimization-based approaches. The details of the DNN-based modeling are described as follows.

### A. Network Structure and Training Method

Our proposed approach uses a fully connected neural network with one input layer, multiple hidden layers, and one output layer, with the ReLU function $\max(\cdot, 0)$ as the activation function for the hidden layers, and the Linear Unit for the output layer. Given all the observations $\mathbf{z}$, our network takes the concatenated vector $\widetilde{\mathbf{z}} = [\mathrm{Re}(\mathbf{z}), \mathrm{Im}(\mathbf{z})]$ as its input, and it outputs a vector $\hat{\widetilde{\mathbf{h}}}$ which approximates the ground truth channel $\widetilde{\mathbf{h}} = [\mathrm{Re}(\mathbf{h}), \mathrm{Im}(\mathbf{h})]$, where $\mathbf{h}$ is the vectorized version of channel matrix, i.e., $\mathbf{h} = \mathrm{vec}(\mathbf{H})$.

In the training stage, a training set $\mathcal{T}$ and a validation set $\mathcal{V}$ containing sufficient $\{\widetilde{\mathbf{z}}, \widetilde{\mathbf{h}}\}$ pairs are first generated following the distributions described in Section V-A. Then they are used to optimize the weights of the neural network based on minimizing the mean squared error cost between the ground truth $\widetilde{\mathbf{h}}$ and the output of the network $\hat{\widetilde{\mathbf{h}}}$. The optimization algorithm we use is an efficient implementation of a mini-batch stochastic gradient descent algorithm called the RMSprop algorithm [19]. We also initialize the network weights with truncated normal distribution normalized by input size as described in [20]. Other algorithm parameters like the learning rate and batch size are all chosen based on validation set performance [10].

### B. Testing Stage

In the testing stage, we first generate multiple channel realizations following the same distribution as the training stage. For each realization, we pass the vectorized feedback $\widetilde{\mathbf{z}}^{(i)}$ through the trained network and collect the vectorized CSI prediction $\hat{\widetilde{\mathbf{h}}}^{(i)}$, where the superscript $(i)$ is used to denote the index of the testing sample. Then we reformulate the CSI to its matrix form $\hat{\mathbf{H}}^{(i)}$ and compute the estimation performance compared with the ground truth $\mathbf{H}^{(i)}$.

## V. Numerical Simulations

In this section, the effectiveness of our proposed low-rank and learning based models compared to the traditional sparse recovery method are demonstrated.

### A. Simulation Setup

The simulations are carried out on a workstation with two Intel Xeon E5-2640 2.4 GHz Processors (CPU), two Nvidia Quadro P6000 Graphical Processing Units (GPUs), and 64 GB of memory. The sparsity-based and low-rank sensing-based optimization algorithms are implemented on MATLAB R2017a, while the DNN approach is implemented in Python 3.6 with TensorFlow 1.5 where the GPUs are used only in the training stage to reduce the training time, but are not used in the testing stage.

Two different problem settings are considered: i) $M_t = 16, M_r = 16, K = 2$, and ii) $M_t = 64, M_r = 64, K = 5$. For every case, we use a DNN with 3 hidden layers and choose neurons for each layer according to the input and output size. For example, for case i) with 100 observations (i.e., the input size is 200), we choose 256, 512, and 1024 neurons for the 1st, 2nd, and 3rd hidden layers, respectively. Monte Carlo simulations are carried out and for each channel realization $\mathbf{H}$, the elevation and azimuth angle for DoAs and DoDs are generated as uniformly distributed random variables, where the elevation angle of $\boldsymbol{\theta}_t$ and $\boldsymbol{\theta}_r$ fall in the range $[0°, 90°]$, and the azimuth angle $\boldsymbol{\phi}_t$ and $\boldsymbol{\phi}_r$ belong to $[-90°, 90°]$. The elements in the vectors $\mathbf{a}_i$'s and $\mathbf{s}_i$'s are generated from the distribution $\mathcal{CN}(0, 1)$, while the path loss coefficient $\beta_i$'s are generated from $\mathcal{CN}(0, 1/\sqrt{2})$. The channel noise $\mathbf{n}_i$'s are assumed to be AWGN with 20 dB signal-to-noise ratio. To implement the traditional sparse recovery method, we use OMP algorithm to solve a vectorized version of problem (7), see [8] for details. The quantization level is chosen to $Q = 60$ for case i) and $Q = 32$ for case ii), in both case the dictionary matrices take about 40 GB memory space.

### B. Performance

To evaluate the estimation performance, the following performance metrics are adopted in our simulations. The normalized mean squared error (NMSE) is given by

$$\mathrm{NMSE}(\mathbf{H}) := \mathsf{E}\Big[||\mathbf{H} - \hat{\mathbf{H}}||_F^2 / ||\mathbf{H}||_F^2\Big]. \tag{9}$$

The signal-to-reconstruction-error (SRE) ratio (in dB) can be expressed as

$$\mathrm{SRE}(\mathbf{H}) := \mathsf{E}\Big[10\log_{10}\big(||\mathbf{H}||_F^2 / ||\mathbf{H} - \hat{\mathbf{H}}||_F^2\big)\Big]. \tag{10}$$

The average beamforming gain (BGain) is given by

$$\mathrm{BGain}(\mathbf{H}) := \mathsf{E}\Big[|\mathrm{Tr}(\hat{\mathbf{H}}^H \mathbf{H})|^2 / ||\hat{\mathbf{H}}||_F^2\Big]. \tag{11}$$

We also compute the CPU time (in seconds) to compare the computational efficiency of the proposed estimation method.

Different performance metrics are reported in Fig. 3, Fig. 4 and Table. I. It can be observed that when we have sufficient observations, both proposed methods can easily achieve very high accuracy, outperforming the sparse-based algorithm. Specifically, for $M_t = M_r = 16$ case, when the number of observations becomes larger than 150, PGD algorithm can achieve more than 20 dB SRE and DNN method can achieve

TABLE I
ESTIMATION AND COMPUTATIONAL PERFORMANCE COMPARISONS FOR A LARGE ANTENNA CASE ($M_t = 64$, $M_r = 64$, AND $K = 5$).

| | NMSE | | | SRE (in dB) | | | BGain ($\times 10^4$) | | | CPU time (in sec.) | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| # of samples | OMP | PGD | DNN | OMP | PGD | DNN | OMP | PGD | DNN | OMP | PGD | DNN |
| 3200 | 0.4706 | 0.0027 | 0.1524 | 3.2735 | 25.6864 | 8.6572 | 1.7334 | 3.8491 | 3.9449 | 193.0590 | 6.1419 | 0.2809 |
| 1600 | 0.5802 | 0.0082 | 0.1903 | 2.3642 | 20.8619 | 7.7301 | 1.5126 | 3.6388 | 3.9323 | 67.7541 | 5.5186 | 0.1532 |
| 800 | 0.6162 | 0.1608 | 0.3112 | 2.1027 | 7.9371 | 5.4465 | 1.4028 | 3.5692 | 2.8953 | 38.7693 | 4.9726 | 0.1209 |
| 400 | 0.6852 | 1.1680 | 0.4221 | 1.6418 | −0.6744 | 3.9292 | 1.1597 | 0.8948 | 2.3033 | 26.1790 | 2.4516 | 0.0920 |

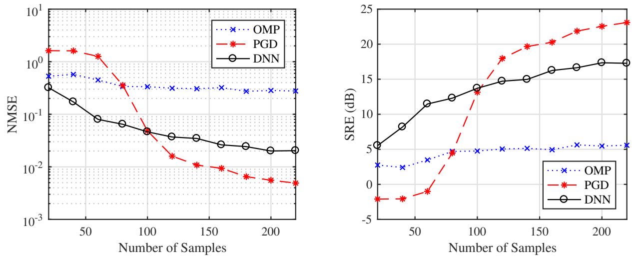


Fig. 3. Estimation performance comparisons, based on NMSE and SRE ($M_t = 16$, $M_r = 16$, and $K = 2$).

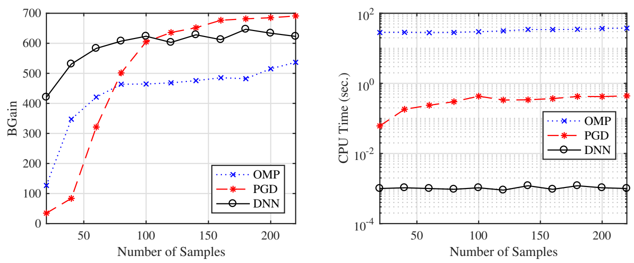


Fig. 4. Estimation and computational performance comparisons, based on BGain and CPU time ($M_t = 16$, $M_r = 16$, and $K = 2$).

more than 15 dB, both outperform the sparse-based algorithm with 5 dB performance.

Surprisingly, when the number of observations decreases, the DNN approach still maintains high performance, and it outperforms both the PGD and OMP algorithms, regardless of the fundamental limit of the sample complexity. For example, when there are only 20 samples, DNN still achieves around 400 BGain, while both PGD and OMP fall below 150.

Further, another key observation is that both the DNN and PGD methods outperform the OMP in terms of computational time. In particular the PGD achieves about 100x and 10x speed up for the small antenna case and big antenna case, respectively, and the DNN achieves more than 1000x speed up for the $M_t = M_r = 16$ case. Also note that it will be difficult to further improve the quantization accuracy for the sparse model in which the OMP method is based on, because the dictionary already takes around 40 GB of memory space.

## VI. CONCLUSION

In this work, we have proposed a low-rank sensing framework to handle the limited feedback based downlink channel estimation problem in massive MIMO systems. The main contribution of this work is two-fold: First, we propose the first low-rank sensing method for this problem, which is very appealing in a number of aspects such as flexibility, provable optimality, and scalability. Second, we propose a deep learning based scheme for real-time channel recovery, which leverages the learning power of DNNs to achieve further reduced feedback overhead.

## REFERENCES


[1] D. Love, R. Heath, V. Lau, D. Gesbert, B. Rao, and M. Andrews, "An overview of limited feedback in wireless communication systems," *IEEE Journal on Selected Areas in Communications*, vol. 26, no. 8, 2008.

[2] S. Lu and Z. Wang, "Joint optimization of power allocation and training duration for uplink multiuser MIMO communications," in *Proceedings of the IEEE Wireless Communications and Networking Conference*, 2015, pp. 322–327.

[3] Z. Gao, L. Dai, Z. Wang, and S. Chen, "Spatially common sparsity based adaptive channel estimation and feedback for FDD massive MIMO," *IEEE Transactions on Signal Processing*, vol. 63, no. 23, pp. 6169–6183, 2015.

[4] J. Choi, D. Love, and P. Bidigare, "Downlink training techniques for FDD massive MIMO systems: Open-loop and closed-loop training with memory," *IEEE Journal of Selected Topics in Signal Processing*, vol. 8, no. 5, pp. 802–814, 2014.

[5] A. Kammoun, H. Khanfir, Z. Altman, M. Debbah, and M. Kamoun, "Preliminary results on 3D channel modeling: From theory to standardization," *IEEE Journal on Selected Areas in Communications*, vol. 32, no. 6, pp. 1219–1229, 2014.

[6] J. Mo, P. Schniter, N. Prelcic, and R. Heath, "Channel estimation in millimeter wave MIMO systems with one-bit quantization," in *Proceedings of the IEEE Asilomar Conference on Signals, Systems and Computers*, 2014, pp. 957–961.

[7] R. Méndez-Rial, C. Rusu, N. González-Prelcic, A. Alkhateeb, and R. Heath, "Hybrid MIMO architectures for millimeter wave communications: Phase shifters or switches?," *IEEE Access*, vol. 4, pp. 247–267, 2016.

[8] P. Alevizos, X. Fu, N. Sidiropoulos, Y. Ye, and A. Bletsas, "Limited feedback channel estimation in massive MIMO with non-uniform directional dictionaries," *arXiv preprint arXiv:1712.10085*, 2017.

[9] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436, 2015.

[10] H. Sun, X. Chen, Q. Shi, M. Hong, X. Fu, and N. Sidiropoulos, "Learning to optimize: Training deep neural networks for wireless resource management," *arXiv preprint arXiv:1705.09412*, 2017.

[11] N. Samuel, T. Diskin, and A. Wiesel, "Deep MIMO detection," *arXiv preprint arXiv:1706.01151*, 2017.

[12] N. Farsad and A. Goldsmith, "Detection algorithms for communication systems using deep learning," *arXiv preprint arXiv:1705.08044*, 2017.

[13] W. Bajwa, J. Haupt, A. Sayeed, and R. Nowak, "Compressed channel sensing: A new approach to estimating sparse multipath channels," *Proceedings of the IEEE*, vol. 98, no. 6, pp. 1058–1076, 2010.

[14] J. Tropp and A. Gilbert, "Signal recovery from random measurements via orthogonal matching pursuit," *IEEE Transactions on Information Theory*, vol. 53, no. 12, pp. 4655–4666, 2007.

[15] T. Blumensath and M. Davies, "Iterative hard thresholding for compressed sensing," *Applied and Computational Harmonic Analysis*, vol. 27, no. 3, pp. 265–274, 2009.

[16] B. Recht, "A simpler approach to matrix completion," *Journal of Machine Learning Research*, vol. 12, no. Dec, pp. 3413–3430, 2011.

[17] E. Candès and B. Recht, "Exact matrix completion via convex optimization," *Foundations of Computational Mathematics*, vol. 9, no. 6, pp. 717, 2009.

[18] P. Jain, R. Meka, and I. Dhillon, "Guaranteed rank minimization via singular value projection," in *Advances in Neural Information Processing Systems*, 2010, pp. 937–945.

[19] G. Hinton, N. Srivastava, and K. Swersky, "Lecture 6a overview of mini–batch gradient descent," *Coursera Lecture slides*, 2012, [Online]. Available: https://class.coursera.org/neuralnets-2012-001/lecture.

[20] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," in *Proceedings of the International Conference on Artificial Intelligence and Statistics*, 2010, pp. 249–256.