A Learnable Optimization and Regularization Approach to Massive MIMO CSI Feedback

Zhengyang Hu, Guanzhang Liu, Qi Xie, Jiang Xue, Senior Member, IEEE, Deyu Meng, Member, IEEE, and Deniz Gündüz, Fellow, IEEE

Abstract—Channel state information (CSI) plays a critical role in achieving the potential benefits of massive multiple input multiple output (MIMO) systems. In frequency division duplex (FDD) massive MIMO systems, the base station (BS) relies on sustained and accurate CSI feedback from users. However, due to the large number of antennas and users being served in each over MIMO systems, feedback overhead can become a bottleneck. In this paper, we propose a model-driven deep learning method for CSI feedback, called learnable optimization and regularization algorithm (LORA). Instead of using $l_1$-norm as the regularization term, LORA introduces a learnable regularization module that adapts to characteristics of CSI automatically. The conventional Iterative Shrinkage-Thresholding Algorithm (ISTA) is unfolded into a neural network, which can learn both the optimization process and the regularization term by end-to-end training. We show that LORA improves the CSI feedback accuracy and speed. Besides, a novel learnable quantization method and the corresponding training scheme are proposed, and it is shown that LORA can operate successfully at different bit rates, providing flexibility in terms of the CSI feedback overhead. Various realistic scenarios are considered to demonstrate the effectiveness and robustness of LORA through numerical simulations.

Index Terms—Massive MIMO, CSI feedback, model-driven, deep learning, regularization learning.

I. INTRODUCTION

A core technology for the sixth generation (6G) of wireless networks, massive multiple input multiple output (MIMO) systems can provide higher data rates and link reliability [1]. To realize the benefits provided by massive MIMO systems, such as beamforming [2] and more reliable signal detection [3], accurate channel state information (CSI) at the base station (BS) is necessary in both the time division duplex (TDD) and frequency division duplex (FDD) modes. In the TDD mode, downlink CSI can be obtained directly from uplink CSI under the assumption of perfect channel reciprocity. However, the TDD mode may not work well in time sensitive scenarios, such as live streaming and vehicular communications [4]. In the FDD mode, the uplink and downlink use different frequency resources at the same time. However, due to the lack of perfect channel reciprocity in the FDD mode, user equipments (UEs) need to estimate downlink CSI and feed it back to the BS [5]. Nevertheless, the huge feedback overhead due to the large number of antennas at the BS and the large number of users being served can become a significant performance bottleneck. Therefore, a CSI feedback method with low overhead and high accuracy is essential to deliver the promised gains of massive MIMO systems in next generation communication networks.

Due to the strong spatial correlations and the shared local scatterers in the propagation environment in massive MIMO systems, CSI exhibits approximate sparsity in the angular-delay domain, which means that CSI matrix can be compressed significantly to reduce the feedback overhead [6]. Compressive sensing (CS)-based methods can be used to project sparse signals to a low-dimensional space and recover them efficiently with theoretical guarantees. The first CS-based CSI feedback method for massive MIMO systems was proposed in [7], which considered both two-dimensional-discrete Fourier transformation (2D-DFT) and Karhunen-Loeve Transform (KLT) as sparsifying bases. In [8], the authors used the statistical information about the angle-of-departure (AOD) to develop a basis for sparsity mapping and a weighted $l_1$-norm was proposed for recovery, which was shown to achieve a better performance than the DFT basis. Considering orthogonal frequency-division multiplexing (OFDM) systems, a multidimensional CS-based analog CSI feedback method was proposed in [9], which treated the CSI feedback design as a multidimensional matrix compression and recovery problem, and exploited the tensor decomposition method. However, these methods are limited in general as they cannot identify the best basis, and the projected CSI matrices are often not perfectly sparse, resulting in performance loss. Although some particular priors are shown to reduce the strict requirement of sparsity [8], [10], [11], these ‘manual’ designs...
are done in a case-by-case basis and are still not efficient enough due to the diverse use cases and high performance requirements of future systems.

In recent years, data-driven methods, in particular, deep learning (DL), has achieved notable success in a variety of wireless communication applications [12], such as channel estimation [13], signal detection [3], joint source-channel coding [14], channel prediction [15], and beamforming [16]. Besides, DL-based methods have also made tremendous strides in increasing the quality of CSI feedback [17].

The authors in [18] were the first to employ DL for CSI feedback reduction and proposed a simple convolutional neural network (CNN) auto-encoder architecture for dimensionality reduction, which has been considered as a baseline for the follow-up studies on DL-based CSI feedback methods. The encoder and decoder in [18] carry out the compression and recovery operations, respectively. Since increasing the receptive field in CNN can extract more information from the input, CsiNet in [19] considers different convolution kernel dimensions. Inspired by the inception model, the authors in [20] designed CRNet, which used multi-paths and multi-receptive fields in both encoder and decoder to improve the performance. MRFNet proposed in [21] shows that the larger number of convolution channels can recover more details of CSI. The authors in [22] exploited the dilate convolution operator to increase the receptive field without increasing the number of parameters, while retaining a high performance. In [23], the popular self-attention architecture, named transformer, was exploited for CSI feedback. To design the lightweight neural network (NN) for CSI feedback, the authors in [24] exploited the complex-value convolution operator and proposed a X-shape NN architecture to reduce the complexity of the NN. In [25], the NNs for real and imaginary parts share parameters, which reduces the complexity of the NN. In [26], projected CSI coefficients are further quantized, and entropy coded to reduce the required rate, and a significant improvement was reported with respect to CsiNet [18]. This approach was extended to CSI feedback from multiple nearby users in [27], where the correlation among CSI matrices was taken into account to achieve better compression efficiency.

Researchers have also adopted other DL techniques for CSI feedback. The authors in [28] proposed a CSI feedback method based on generative adversarial networks (GANs). In [29], CSI feedback is modeled as an image super resolution problem, and SRNet is proposed. In [30], inspired by the elastic network, ACRNet is proposed to adjust the complexity and performance of the NN. To extract time correlation of CSI, CNN-LSTM-A [31] and CsiNet-LSTM [32] are proposed. Feedback schemes that are designed for a particular task, such as beamforming, were investigated in [33] and [34], which could reduce the feedback overhead by bypassing channel estimation. Although the aforementioned CNN-based methods have achieved significant performance improvements compared to their CS-based counterparts, these methods simply treat the channel matrix as a two-dimensional ‘image’ with local correlations, which may limit their performance.

Model-driven DL methods exploit our prior knowledge about the particular learning problem. Bringing model-driven and data-driven approaches together, model-driven DL methods not only make the learned model more explainable and predictable [35], but also avoid the requirements for accurate and explicit modeling. In [36], the authors proposed a model-driven DL method to improve the recovery accuracy in CSI feedback, by unfolding a conventional CS algorithm into a NN and learning the measurement matrix. Inspired by the transformation matrix design and unfolding, TiLISTA-Joint was proposed in [37], which not only learned the down-sampling matrix, but also used a sparse auto-encoder architecture to learn the sparse transformation. To further improve the recovery accuracy, the authors exploited the attention mechanism for learning sparse transformation and proposed FISTA-Net in [38]. In [39], the authors designed a model-driven module to pre-compress CSI based on self-information, and the entire NN was trained in a data-driven manner. For the better presentation and easier reading, we present TABLE I to summarize and show the detailed technical information of related works. Key ideas and main characteristics show the similarities and differences of these works, respectively.

Although model-driven DL methods have exhibited remarkable success in CSI feedback, current methods are all designed with an $l_1$-norm regularization term, which cannot extract the prior knowledge of CSI in some cases. Actually, how to design a suitable regularization (i.e., data prior) is an active research problem in machine learning. It is well-known that $l_0$-norm is the optimal regularization term to describe sparsity, but the optimization with $l_0$-norm is intractable. When the measurement matrix satisfies restricted isometry property (RIP) condition, $l_1$-norm is equivalent to $l_0$-norm in terms of sparse signal recovery [43]. The authors in [44] utilized a mixture of Gaussian distributions to learn the noise distribution. Although the proposed method in [44] does not explicitly formulate a regularization term, the data prior is learned through the loss function. Due to the strength of DL, the authors proposed a proximal dehaze-net, which learned a haze-related prior to achieve the obvious performance gain in single photo dehazing [45]. In [46], the authors proposed RCDNet to automatically extract the prior from rain images for better deraining.

Inspired by the model-driven methods for CSI feedback and regularization term learning in [44], [45], and [46], in this paper, we propose a joint regularization and optimization method, called learnable optimization and regularization algorithm (LORA). LORA exploits a NN to learn the regularization term for better fitting the characteristics of CSI, and develops an iterative algorithm with learnable parameters to achieve performance gains.

The main contributions of this work are summarized as follows:

- Existing model-driven DL architectures for CSI feedback all unfold the algorithm derived from an optimization problem with the $l_1$-norm regularization term, which cannot describe the prior of CSI well due to its imperfect sparsity. Instead, LORA treats the regularization term as a learnable function that can be adjusted according to the characteristics of CSI itself. The proposed method results in a novel algorithm of model-driven DL for CSI feedback with significant improvements in the
performance in terms of both the normalized mean square error (NMSE) and the achievable average rate.

- To further mitigate the effect of quantization in LORA, we exploit quantization-aware training (QAT) with learnable quantization parameters, such as quantization scale and zero point value. The proposed quantization method eases the performance decay caused by quantization in different bit levels.

- The numerical results show that LORA has a superior performance than CsiNet+ [19], CRNet [20], DCRNet [22], TransNet [23], ACRRNet [30], TiLISTA-Joint [37] and ISTA-NET in different scenarios based on 3GPP TR 38.901 [47]. Moreover, the performance with channel estimation error and complexity comparisons are provided. We also carry out ablation studies to explore the effects of different modules of LORA on the final performance.

The rest of this work is organized as follows. Section II describes the massive MIMO system, CSI feedback procedure and channel model. We re-formulate the CSI feedback problem and present the basic algorithm in Section III. In Section IV, we present LORA with insights in detail. Numerical results and analyses are provided in Section V to demonstrate the superiority of LORA compared to the existing CSI feedback schemes. Finally, the paper is concluded in Section VI.

Notations: Throughout the paper, bold uppercase letters, bold lowercase letters and non-bold letters are used to denote matrices, vectors and scalars, respectively. \( \parallel \cdot \parallel_2 \) is the Euclidean norm. \( \cdot \parallel \) stands for element-wise absolute value. \( (\cdot)^T \) and \( (\cdot)^H \) are transpose and conjugate transpose, respectively. The real and complex number fields are \( \mathbb{R} \) and \( \mathbb{C} \), respectively. The expectation operation is represented by \( \mathbb{E}\{\cdot\} \). \( \hat{\cdot} \) is used to indicate the definition of the value and new variable, while \( \approx \) represents approximate equality. \( \mathcal{CN}(\mu_g, \sigma_g^2) \) denotes complex Gaussian distribution with mean \( \mu_g \) and variance \( \sigma_g^2 \).

II. SYSTEM MODEL

A. Massive MIMO System and DL-Based CSI Feedback

We consider the downlink of a single-cell massive MIMO OFDM system in the FDD mode, where a BS equipped with \( N_t \gg 1 \) antennas serves a single-antenna user equipment

| Key ideas | Related works | Main characteristics |
|----------|---------------|---------------------|
| Convolution operator design | CNet [19], CRNet [20], MDRNet [21], DCRNet [22] | Increasing the depth and width of the CNet, Strip convolution kernel pattern, multiple paths and feature fusion, Large number of convolution channels, Increasing encoding fields with dilated convolutions |
| Novel DL techniques | TransNet [23], SRNs [20], ACRNet [30] | Self-attention and transformer, GAN, Image super resolution technique, Elastic network and network aggregation technique |
| Multi-domain correlation utilization | DDNet [27], CSMN-LSTM-A [31], CSINET-LSTM [32] | Correlation among nearby users’ CSI, Correlation among the time domain with the attention mechanism, Correlation among the time domain based on CNet |
| Joint other modules | [33], [34] | Joint the channel estimation, CSI feedback and beamforming, Joint the channel estimation, CSI feedback and beamforming of multiple users, and exploiting users correlations |
| Model-driven DL methods | TiLISTA-Joint [37], FSTNet [38], IdiaNet [39] | Learning the measurement matrix, Learning the measurement matrix, step size, sparse auto-encoder and sparse transformation, Unfolding FISTA and considering both sparsity and low-rank properties of CSI, Using a model-driven module to pre-compress CSI based on the self-information |
| Changeable compression rate | [40], [41] | Designing an overhead control unit to discard part of the output of the encoder, Using principal component analysis to compress CSI and proposing a quantization method based on k-means clustering |
| Viewing CSI as sequence | [42] | Regarding CSI as a sequence and proposing a novel network based on 2D-LSTM |

Fig. 1. The real part of the complete channel matrix in (a) spatial-frequency domain, the real part of the complete (b) and the cropped (c) channel matrices in angular-delay domain.

(UE) [18] over \( \tilde{N}_c \) subcarriers. The received signal at the \( n \)-th subcarrier \( (n = 1, \ldots, \tilde{N}_c) \) in the frequency domain can be expressed as

\[
y_n = \tilde{h}_n^H v_n x_n + z_n,
\]

where \( \tilde{h}_n \in \mathbb{C}^{N_t \times 1} \), \( v_n \in \mathbb{C}^{N_t \times 1} \) and \( x_n \in \mathbb{C} \) are the downlink channel vector, corresponding precoding vector and the modulated transmitted signal, respectively, while \( z_n \sim \mathcal{CN}(0,1) \) denotes the complex random additive Gaussian noise. The detailed channel model will be introduced in Section II-B. In the FDD mode, the downlink channel vector \( \tilde{h}_n \) has to be estimated at the UE and sent back to the BS. The overall downlink CSI matrix can be expressed as

\[
\tilde{H}_f = F_d \tilde{H} F_a,
\]

where \( F_d \in \mathbb{C}^{N_t \times N_t} \) and \( F_a \in \mathbb{C}^{N_c \times N_c} \) are the DFT matrices. As mentioned, \( \tilde{H}_f \) has approximate sparsity in the angular-delay domain. Moreover, only the first few columns of \( \tilde{H}_f \) have significant values because the delay between multipath components typically lies within a limited period in

\[\text{1The exact subset of significant values depends on the distribution of the channel, and does not always correspond to the first part.}\]
the delay domain. So, we preserve only the first \( N_c < \tilde{N_c} \) columns and remove the rest. An example illustrating the real part of the complete and cropped channel matrices in the angular-delay domain, and the complete channel matrix in the spatial-frequency domain are shown in Fig. 1. We denote the truncated CSI matrix by \( \tilde{H} \), which consists of \( 2N_tN_c \) real numbers after decomposing the real and imaginary parts. Although the required number of feedback parameters has been reduced from \( 2N_t \times \tilde{N_c} \) to \( 2N_t \times N_c \), the feedback overhead is still too large and will consume significant channel resources.

Following the prior works, we consider a pair of encoder-decoder networks to compress and recover CSI, and a pair of quantizer-dequantizer to reduce the required CSI feedback bits. As shown in Fig. 2, \( \tilde{H} \in \mathbb{R}^{2 \times N_t \times N_c} \) is fed into the encoder network, whose output is \( M \) floating point parameters. Then, these parameters are quantized to a bit stream by the quantizer, whose output is denoted by \( \bar{c} \). The dequantizer and the decoder are applied at the BS. The dequantizer transforms \( \bar{c} \) back to floating point numbers, and the decoder transforms the dequantizer output to \( \tilde{H} \in \mathbb{R}^{2 \times N_t \times N_c} \). The compression ratio (CR) is defined as \( \text{CR} \triangleq M/2N_tN_c \). The whole feedback procedure can be presented as

\[
\begin{align*}
\bar{c} &= Q_e(\bar{f}_c(\tilde{H}, \Theta_c), \Theta_q) , \\
\tilde{H} &= f_d(\bar{Q}_d(\bar{c}, \Theta_{dq}), \Theta_d)
\end{align*}
\]

where \( \tilde{H} \) is the reconstructed and cropped CSI, \( \bar{f}_c \) and \( \bar{f}_d \) denote the encoder and decoder functions, \( Q_e \) and \( \bar{Q}_d \) denote the quantizer and dequantizer functions, while \( \Theta_c \), \( \Theta_{dq} \), \( \Theta_d \) are the parameters of \( f_c \), \( f_d \), \( Q_e \) and \( \bar{Q}_d \), respectively. The complete CSI can be obtained by zero-padding followed by inverse DFT operation on \( \tilde{H} \).

### B. Channel Model

Due to the large size of the antenna array in massive MIMO systems, spherical wave channel model should be considered instead of a plane wave channel model [48]. This is also verified through measurements in [49] and [50]. Spherical wave channel model is more realistic, and has been widely adopted in wireless communication applications [51], [52], [53]. We also adopt a 3-D geometric stochastic channel model [54] in this work, which incorporates the spherical wave channel model.

We model the paths and sub-paths of channels distinctly by considering the scatterers in the environment. The adopted model does not require an exact geometric representation of the environment, but instead, relies on the statistical distribution of the scattering clusters. The last-bounce scatterers in the single-bounce\(^2\) model is considered, which focus on the signal propagation from the last scatterers to the receiver, and is not an exact geometric representation of the propagation environment. The locations of the last-bounce scatterers are used to generate the channel coefficients. The length of the \( k \)-th path, \( d_k \), can be expressed as

\[
d_k = \|r\|^2 + \tau_k c,
\]

where \( \|r\|^2 \) is the distance between the BS and the UE, \( \tau_k \) is the delay of the \( k \)-th path, and \( c \) is the speed of light. We define \( \mathbf{q}_{k,m} \) as the arrival vector of the \( m \)-th sub-path in the \( k \)-th path pointing from the UE location to the scatterers. Then, its length can be expressed as

\[
\|\mathbf{q}_{k,m}\|^2 = \frac{d_k^2 - \|r\|^2}{2(d_k + r^T \mathbf{q}_{k,m})},
\]

where

\[
\mathbf{q}_{k,m} = \begin{bmatrix}
\phi^a_{k,m} \\
\theta^a_{k,m} \\
\sin \theta^a_{k,m} \\
\sin \phi^a_{k,m}
\end{bmatrix},
\]

\( \phi^a_{k,m} \) and \( \theta^a_{k,m} \) are the azimuth and elevation angle of arrival (AOA) of the \( m \)-th sub-path in the \( k \)-th path, respectively. The location of the scatterers can be obtained by the geometric relationship shown in Fig. 3.

To model the spherical wave, each antenna element \( s \) at the BS is considered separately. Given a reference antenna (which can be chosen arbitrarily in the antenna array) element location and its departure vector of the \( m \)-th sub-path in the \( k \)-th path \( \mathbf{p}_{k,m} \), the departure vector of the \( s \)-th antenna of the \( m \)-th sub-path in the \( k \)-th path \( \mathbf{p}_{s,k,m} \), the corresponding elevation AOD \( \phi^d_{s,k,m} \) and azimuth AOD \( \theta^d_{s,k,m} \), can be derived as

\[
\phi^d_{s,k,m} = \arcsin \frac{\mathbf{p}_{s,k,m,z}}{\|\mathbf{p}_{s,k,m}\|},
\]

and

\[
\theta^d_{s,k,m} = \arctan \frac{\mathbf{p}_{s,k,m,y}}{\mathbf{p}_{s,k,m,x}},
\]

where

\[
\mathbf{p}_{s,k,m} = \mathbf{p}_{k,m} - \mathbf{e}_s,
\]

and \( \mathbf{e}_s \) is the vector from the reference antenna element to the \( s \)-th antenna element, while \( \mathbf{p}_{s,k,m,x} \), \( \mathbf{p}_{s,k,m,y} \) and \( \mathbf{p}_{s,k,m,z} \) are

\(^2\) The following illustrations are based on the single-bounce model for simplifying notations, which can be easily extended to the multi-bounce model. In Section V, the proposed method is evaluated on the multi-bounce model. More details about the multi-bounce model can be found in [54].
the Cartesian coordinate components of \( p_{s,k,m} \). Therefore, the deterministic phase \( \psi_{s,k,m} \) and delay \( \tau_{s,k} \) can be derived by

\[
\psi_{s,k,m} = \frac{2\pi}{\lambda_c} (d_{s,k,m} \mod \lambda_c),
\]

and

\[
\tau_{s,k} = \sum_{m=1}^{M_k} d_{s,k,m} / M_k,
\]

where

\[
d_{s,k,m} = ||p_{s,k,m}||_2 + ||q_{k,m}||_2,
\]

\( M_k \) is the number of sub-paths in the \( k \)-th path, \( \lambda_c \) is the wavelength, and \( \mod \) stands for the modulo operation.

Therefore, the channel between the \( s \)-th BS antenna and the UE via the \( k \)-th path can be described as (14), shown at the bottom of the page, where \( P_{s,k,m} \), \( F_{tx,\theta} \), \( F_{tx,\phi} \), \( F_{rx,\theta} \), \( F_{rx,\phi} \), \( j \) and \( \psi_{s,k,m} \) are the polarization coupling matrix of the \( s \)-th antenna of the \( k \)-th sub-path in the \( k \)-th path, elevation polarimetric antenna response at the receiver, azimuth polarimetric antenna response at the receiver, elevation polarimetric antenna response at the transmitter, azimuth polarimetric antenna response at the transmitter, imaginary unit, and the random phase of the \( m \)-th sub-path in the \( k \)-th path, respectively.

Therefore, the \( (s,l) \)-th element of \( \tilde{H} \) in spatial-frequency domain can be expressed as

\[
\tilde{H}_{s,l} = \sum_{k=1}^{K'} g_{s,k} e^{-j 2\pi \frac{l-1}{N_c} B' \tau_{s,k}},
\]

where \( B' \) is the bandwidth, \( K' \) is the number of paths, \( s = 1, \ldots, N_t \) and \( l = 1, \ldots, N_c \).

III. PROBLEM FORMULATION

In this section, we first formulate CSI feedback as a linear inverse problem. Then, the iterative shrinkage-thresholding algorithm (ISTA) will be introduced, which inspired the proposed method.

A. Problem Formulation

We consider a learnable matrix as the encoder, which can be conveniently designed as a light linear layer, and is appropriate for the UE due to its limited computation and storage ability. Therefore, the projected vector \( v \) can be expressed as

\[
v = Ax,
\]

where \( A \) is the learnable matrix and \( x \in \mathbb{R}^{2N_t N_c} \) is the CSI matrix \( H \) in the vector form. Then, the decoder at the BS can be regarded as solving an inverse problem, which is presented as

\[
\min_x \frac{1}{2} ||v - Ax||_2^2.
\]

Due to the huge dimension reduction, the problem (17) is highly ill-posed; and hence, hard to solve directly. Typically, a regularization term is introduced into the optimization function to exploit any known prior information about the optimal solution. Therefore, the problem (17) can be modified as

\[
\min_x \frac{1}{2} ||v - Ax||_2^2 + R(x),
\]

where \( R(x) \) is the regularization term.

B. ISTA

Considering the sparsity of CSI, conventional CS-based and model-driven DL methods utilize \( l_1 \)-norm as the regularization term. Then, the problem (18) can be written as

\[
\min_x \frac{1}{2} ||v - Ax||_2^2 + \lambda \|x\|_1.
\]

ISTA [55] is a classical iterative method to solve (19), and the related model-driven DL methods for CSI feedback [36], [37] are inspired by it. Its iterative formulation at the \( t \)-th step can be expressed as follows:

\[
\begin{align*}
\mathbf{u}^{(t)} & = x^{(t-1)} - \alpha A^T (Ax^{(t-1)} - v), \\
\mathbf{x}^{(t)} & = \text{sign}(u^{(t)}) \max(0, |u^{(t)}| - \theta),
\end{align*}
\]

where \( u^{(t)} \), \( 0, \theta \) and \( \alpha \) are the intermediate variable, zero vector, thresholding term and step size, respectively. The sign and \( \max \) are element-wise operations, which can be expressed as

\[
\text{sign}(u) = \begin{cases} 1 & \text{if } u > 0, \\ 0 & \text{if } u = 0, \\ -1 & \text{otherwise}, \end{cases}
\]

and

\[
\max(u, \theta) = \begin{cases} u & \text{if } u \geq \theta, \\ \theta & \text{if } u < \theta. \end{cases}
\]

IV. DESIGN OF LORA AND TRAINING SCHEME

In this section, we first propose a novel model-driven DL method, called LORA, which unfolds the derived iterative formulations to a NN and incorporates a regularization learning module. Moreover, considering the quantization in CSI feedback procedure, QAT and learnable quantization methods will be employed.

\[
g_{s,k} = \sum_{m=1}^{M_k} \begin{bmatrix} F_{tx,\theta} \left( \theta^d_{s,k,m}, \varphi^d_{s,k,m} \right) \\ F_{tx,\phi} \left( \theta^d_{s,k,m}, \varphi^d_{s,k,m} \right) \\ F_{rx,\theta} \left( \theta^d_{s,k,m}, \varphi^d_{s,k,m} \right) \\ F_{rx,\phi} \left( \theta^d_{s,k,m}, \varphi^d_{s,k,m} \right) \end{bmatrix}^T P_{s,k,m} e^{-j \psi_{s,k,m}}
\]

(14)
learn the regularization term implicitly. To learn to fit a fixed regularization term, we employ a learnable from the previous works designing the sparsity transformation matrix $A$ a learnable parameter that is different in each layer. The connected sequentially. The step size $\alpha$ by treating each iteration as a separate layer, which are the idea of unfolding, the decoder of LORA is developed $\nabla$ where $R$ of problem (18) can be derived as $\nabla R(x^{(t-1)})$, (24)

where $\nabla R(\cdot)$ stands for the gradient of $R(\cdot)$. Built upon the idea of unfolding, the decoder of LORA is developed by treating each iteration as a separate layer, which are connected sequentially. The step size $\alpha$ in (24) is set as a learnable parameter that is different in each layer. The matrix $A$ is the same one used by the encoder. Different from the previous works designing the sparsity transformation to fit a fixed regularization term, we employ a learnable regularization term to better fit the characteristics of CSI, and learn the regularization term implicitly. To learn $\nabla R(\cdot)$, each layer employs a regularization learning module, which will be introduced in detail in the next sub-section. Together with the encoder, a complete NN architecture is established, and the overall architecture of LORA is presented in Fig. 4 (a).

Model-driven DL exploits NNs to replace explicit expressions or manually set parameters in model-based iterative algorithms. This can mitigate the performance loss due to inaccurate modeling, while exploiting the valuable knowledge of the model simultaneously. Besides, model-driven DL methods can also prevent the over-fitting problem, and are usually easier to train compared to purely data-driven NN approaches.

Notably, the initialization of $x$ is important due to the use of gradient descent. Since CSI has sparsity, and its values are near zero, $x^{(0)} = 0$ is considered as the initialization for LORA.

### A. Architecture of LORA

As in Equation (16) presented above, we consider the CSI in the vector form. However, instead of fixing the regularization term to $l_1$-norm, we consider $R(x)$ as a learnable transform, which is assumed to be differentiable. Then, the iterative formulation at the $t$-th step ($t = 1, \ldots, T$) in the solution of problem (18) can be derived as

$$x^{(t)} = x^{(t-1)} - \alpha^{(t-1)} \left( A^T (A x^{(t-1)} - v) + \nabla R(x^{(t-1)}) \right),$$

where $\nabla R(\cdot)$ stands for the gradient of $R(\cdot)$. Built upon the idea of unfolding, the decoder of LORA is developed by treating each iteration as a separate layer, which are connected sequentially. The step size $\alpha$ in (24) is set as a learnable parameter that is different in each layer. The matrix $A$ is the same one used by the encoder. Different from the previous works designing the sparsity transformation to fit a fixed regularization term, we employ a learnable regularization term to better fit the characteristics of CSI, and learn the regularization term implicitly. To learn $\nabla R(\cdot)$, each layer employs a regularization learning module, which will be introduced in detail in the next sub-section. Together with the encoder, a complete NN architecture is established, and the overall architecture of LORA is presented in Fig. 4 (a).

Model-driven DL exploits NNs to replace explicit expressions or manually set parameters in model-based iterative algorithms. This can mitigate the performance loss due to inaccurate modeling, while exploiting the valuable knowledge of the model simultaneously. Besides, model-driven DL methods can also prevent the over-fitting problem, and are usually easier to train compared to purely data-driven NN approaches.

Notably, the initialization of $x$ is important due to the use of gradient descent. Since CSI has sparsity, and its values are near zero, $x^{(0)} = 0$ is considered as the initialization for LORA.

### B. Regularization Learning Module

The parameters in ISTA, such as the measurement matrix, step size, etc, are all treated as learnable parameters in existing works. However, the regularization term is set as conventional $l_1$-norm. Meanwhile, $l_1$-norm is not a fully accurate prior because of the weak sparsity of CSI. Even with strict sparsity, the measurement matrix needs to satisfy RIP condition to ensure the exact signal recovery by using $l_1$-norm. Therefore, making the regularization learnable to directly fit the characteristics of CSI is a promising approach.

The architecture of the regularization learning module is shown in Fig. 4 (b), which is a light multi-layer perceptron (MLP). The number of neurons of the input layer, hidden layer, and output layer of the regularization learning module are set as $2N_tN_c$, $N_m$, and $2N_tN_c$, respectively. Therefore, the mathematical expression for the regularization learning module can be expressed as

$$MLP(x) = W_2 \sigma(W_1 x),$$

where $W_1 \in \mathbb{R}^{N_m \times 2N_tN_c}$ and $W_2 \in \mathbb{R}^{2N_tN_c \times N_m}$ are the parameters of the first and second linear layers, $\sigma(\cdot)$ is the rectified linear unit (ReLU) activation function and $x \in \mathbb{R}^{2N_tN_c}$ is the input.

The reasons for choosing MLP as the architecture of the regularization learning module are as follows: (1) MLP has the universal approximation property [56], and hence, it is capable to characterize the complex properties of CSI; (2) The linear layer in MLP has a dense connection architecture, which can keep the original information as much as possible compared to a locally connected architecture, such as a convolution operator. Since the regularization term is part of the optimization problem, which is fixed once the training is finished, the parameters of the MLP should be shared by all the layers. This also reduces the number of trainable parameters.

### C. Training Scheme With Quantization

LORA is trained in an end-to-end manner. Mean square error (MSE) is used as the loss function, which can be written...
as
\[ L(\Theta) = \frac{1}{T} \sum_{i=1}^{T} \| \hat{H}_i - H_i \|_2^2, \]  \tag{26} 
where \( T \) is the total number of samples in the training set and \( \Theta \) denotes the parameters of the NN, including the MLP. To avoid hyper-parameter tuning for dynamic learning rate adjustment operator, ADAM [57] with fixed learning rate is applied as the optimization operator.

The quantization module is also considered in CSI feedback for practical implementation. The conventional quantization procedure can be written as
\[ q = \text{round} \left( \text{clip} \left( \frac{r - z}{s}, n, p \right) \right), \tag{27} \]
\[ b = \text{num2bit}(q), \tag{28} \]
where \( r \) refers to real number to be quantized, \( s \) and \( z \) are the scale and zero point values, respectively (i.e., the parameters of the quantization function), \( q \) is an integer corresponding to the quantized value, \( \text{num2bit} \) is the function that converts an integer to its binary representation, \( b \) is the resultant bit stream, \( n = -2^{B-1} \) and \( p = 2^{B-1} - 1 \) are the lower and upper bounds on the clip function, \( B \) is the number of quantization bits per real dimension, \( \text{round} \) is the rounding function that maps a floating number to the closest integer, \( \text{clip} \) function aims to limit the range of the input, which can be expressed as
\[ \text{clip}(x, n, p) = \begin{cases} 
  n & \text{if } x < n, \\
  x & \text{if } n \leq x \leq p, \\
  p & \text{if } x > p. 
\end{cases} \tag{29} \]

The corresponding dequantization procedure can be expressed as
\[ \hat{q} = \text{bit2num}(b), \tag{30} \]
\[ \hat{r} = \hat{q} \times s + z, \tag{31} \]
where \( \text{bit2num} \) is the function which converts a binary number \( b \) to the equivalent integer value, \( \hat{q} \), and \( \hat{r} \) is the dequantized float values corresponding to \( r \). Quantization and dequantization operations can be regarded as two blocks, which are inserted to the end of the encoder and the beginning of the decoder, respectively. Therefore, the forward procedure of NN can be regarded as CSI feedback with quantization and dequantization. Meanwhile, since end-to-end training is applied, the backward procedure can be regarded as learning the parameters with the effect of quantization. However, the rounding function is not differentiable, which would prevent back-propagation during training. Instead, we can employ straight-through differentiation [58], where we set
\[ \frac{\partial \text{round}(x)}{\partial x} \triangleq 1. \tag{32} \]

The aforementioned training scheme is a modified QAT method, which is inspired by the QAT in NN quantization [59]. Although QAT achieves a reasonable performance, it still has the weakness that the scale and zero point values are manually set. As the quantization procedure is embedded into the end-to-end training, the scale and zero point values can also be learned and trained jointly. Considering back-propagation during training, the derivatives of scale and zero point values can be derived from (27), (31) and (32) as follows:
\[ \frac{\partial \hat{r}}{\partial s} = \frac{\partial q}{\partial s} s + q \sim \partial \left( \frac{r - z}{s} \right) s + q \]
\[ \sim \begin{cases} 
 r - z & \text{if } n < \frac{r - z}{s} < p \\
 n & \text{or } p & \text{otherwise}, \end{cases} \]
\[ \frac{\partial \hat{r}}{\partial z} = \frac{\partial q}{\partial z} s + 1 \sim \partial \left( \frac{r - z}{s} \right) s + 1 \]
\[ \sim \begin{cases} 
 0 & \text{if } n < \frac{r - z}{s} < p \\
 1 & \text{otherwise}, \end{cases} \tag{34} \]
where the derivatives of \( \text{clip}(x, n, p) \) at \( x = n \) and \( x = p \) are set as 0 [60].

We name the scale learnable quantization method as LSQ, and both scale and zero point value learnable quantization method as LSZQ. The LSQ and LSZQ are modified versions of the NN quantization methods in [60] and [61], respectively.

V. NUMERICAL EXPERIMENT

In this section, we study the effects of different design options of LORA. We present numerical results evaluating the performance of LORA in terms of the reconstruction accuracy, and the achievable rate.

A. Experiment Settings

1) Data Generation: QuaDRiGa [54] is a general channel simulator that meets the 3GPP standards. The spherical waves introduced in Section II as well as other realistic scenarios can be modeled by QuaDRiGa. Therefore, in this work, we use QuaDRiGa to generate CSI matrices in rural macro non-line-of-sight (RMANLOS), urban macro non-line-of-sight (UMANLOS), and urban micro non-line-of-sight (UMINLOS) scenarios. The carrier frequency, number of subcarriers, subcarrier interval, and \( N_c \) are set as 3.5GHz, 1024, 30kHz, and 1024, respectively, for the above three scenarios. The BS is equipped with a cross-polarized uniform planar array (UPA) with half wavelength antenna spacing, where the number of horizontal antennas is 4, and the number of vertical antennas is 4 and 8 for \( N_t = 32 \) and \( N_t = 64 \), respectively. The UE is assumed to move along a linear trajectory with a velocity of \( v = 6 \text{km/h} \). The heights of the BS are 10m, 10m, and 25m for RMANLOS, UMINLOS and UMANLOS scenarios, respectively. Training and test datasets are generated with 40000 and 10000 samples, respectively.

2) Training Settings and Evaluation Metric: LORA is implemented in PyTorch. The parameters of the NN are updated and optimized by the ADAM optimizer with default settings. The learning rate, number of epochs and batch size are set to 0.001, 1000 and 200, respectively. The number of layers in LORA, i.e., \( T \), is set as 4, while \( N_{in} \) is set as 1024 in
the simulations. We use the NMSE as the evaluation metric, which is defined as

$$\text{NMSE} = \mathbb{E} \left\{ \frac{\| H - \hat{H} \|^2}{\| H \|^2} \right\}. \quad (35)$$

Unless stated otherwise, all experiments are implemented with the above settings.

### B. Ablation Studies for the LORA Architecture

In this sub-section, the effects of different design options of LORA will be studied. The design motivations presented in Section IV are supported by the following results.

1) **Architecture of the Regularization Learning Module:** To further motivate using MLP as the regularization learning module, we present the output of the last layer using MLP and CNN in Fig. 5. The part of the output of the last layer, which corresponds to the real part of the CSI, is shown in the figure. By comparing the visualization results, it can be seen that using CNN loses some information in the region of red box, while the MLP recovers them well. To further investigate the effect of two architectures to the iterative optimization procedure, we also calculate the NMSE of the output of each layer in LORA, which is given in TABLE II. It can be seen that MLP has better performances than CNN at all iterations. The above visualization and numerical results show that using MLP for the regularization learning module in LORA achieves a better performance than using CNN.

2) **Effect of the Learnable Regularization Module:** To verify the effectiveness of the regularization learning module in LORA, and to motivate the particular architecture argued for in Section IV-A, the performance of conventional ISTA, ISTA-NET, TiLISTA-Joint and LORA are compared next. The ISTA-NET stands for an ISTA unfolding method, which has learnable measurement matrix, step size and threshold. In TiLISTA-Joint, in addition to the parameters in ISTA-NET, a sparse transformation is also learned. In this experiment, the conventional ISTA is the baseline method, which has no learnable part. The results are shown in Fig. 6. As we can see, the performance increases as we learn more parameters of the underlying model. This phenomenon suggests that the performance can be improved by making more of the model parameters learnable. Specifically, the results verify that the learnable regularization outperforms fixing $l_1$-norm as the regularization term. In addition, the importance of

| Order of Layers | MLP  | CNN  |
|----------------|------|------|
| 1              | 0.2936 | 0.317 |
| 2              | 0.0888 | 0.5911 |
| 3              | 0.0174 | 0.0848 |
| 4              | 0.0012 | 0.0753 |

### TABLE II

**The NMSE Performance of the Output of Each Layer of a Trained LORA Using MLP and CNN for CR = 1/16 in the RMANLOS Scenario**

regularization in terms of the performance can be shown among the compared DL methods.

We also compare the $l_1$-norm of the output of the ISTA-NET and LORA, to further study the learned regularization term. The results are shown in TABLE III. Although the learned regularization term has no explicit formulation, we can conclude that the learned regularization term is different from the conventional $l_1$-norm, and the improved performance of LORA with the learned regularization term confirms that $l_1$-norm is not the right regularization for this problem.

3) **Effect of Different Layers of LORA and the Width of MLP:** The number of layers of LORA and the width of the hidden layer in MLP are investigated next. The impact of these two hyper-parameters on the performance of LORA can be seen in TABLE IV. If the performance is more important and the increased complexity can be accommodated, such as the high accuracy communication, LORA has the potential to improve the performance by using more layers according to the results in TABLE IV. We also observe that increasing the width of MLP from $N_w = 512$ to $N_w = 1024$ can significantly boost the performance. Indeed, simply using a wider MLP can be sufficient as the gains from increasing the number of layers from $T = 4$ to $T = 7$ is relatively marginal in this case. Moreover, it is worth mentioning that increasing the width or depth of a NN can make training process more difficult, e.g., due to gradient vanishing and explosion. In TABLE IV, we can observe that increasing $N_w$ from 1024 to 4096 results in a worse NMSE performance. To maximize the performance in each setting, the training parameters, such as the learning rate and batch size, need to be carefully adjusted.

### C. Further Performance Results

In this sub-section, the performance of LORA is evaluated in the aforementioned three scenarios, and compared with seven benchmarks to investigate its effectiveness and robustness. CsiNet++ [19] and CRNet [20] are considered as two CNN-based benchmarks. Meanwhile, TiLISTA-Joint [37] and the ISTA-NET are considered as the unfolding-based benchmarks. To further highlight the superiority of LORA, DCRNet [22], TransNet [23] and ACRNet [30] are also applied...
as benchmarks. The results are shown in Fig. 7 - Fig. 9 for the three scenarios RMANLOS, UMANLOS and UMINLOS, respectively. LORA outperforms all benchmarks clearly for all CR values in all the scenarios, especially for small CR values. The presented results demonstrate the superiority of LORA compared to recent works in terms of the recovery accuracy even for CR = 1/64. Moreover, the robustness of LORA in terms of achieving a superior performance in a variety of communication scenarios and CR values are verified. We conclude that LORA is a promising method for practical cases with high performance and small overhead. Next, we consider a more complex scenario with a 64-antenna BS serving UEs with 2 antennas. The NMSE performance of the proposed method in this scenario is presented in TABLE V. The results show that the proposed method can be easily extended to more antennas at the UEs or the BS, and serve more complex massive MIMO systems.

Table V: The NMSE performance of the proposed method in RMANLOS scenario with a 64-antenna BS serving UEs with 2 antennas

| CR   | 1/16 | 1/32 | 1/64 |
|------|------|------|------|
| NMSE | 0.00331 | 0.00347 | 0.01062 |

Fig. 10. The NMSE performance with different SNRs.
Joint and LORA in the RMANLOS scenario are compared. The signal-to-noise ratio (SNR) of CSI is used to adjust the noise level introduced to the CSI. According to the results in Fig. 10, LORA significantly outperforms TiLISTA-Joint in all imperfect CSI noise levels, which suggests that LORA has better capability to adapt to different degrees of channel estimation errors. Notably, the NMSE here is defined as

$$\text{NMSE} \triangleq \mathbb{E} \left\{ \frac{\|H_{gt} - \hat{H}\|^2_2}{\|H_{gt}\|^2_2} \right\}, \quad (36)$$

where $H_{gt}$ denotes the perfect CSI. Since the input of LORA contains channel estimation errors, the results in Fig. 10 also show the denoising capability of LORA in addition to the CSI compression and recovery.

### D. Complexity

Here, we analyze the storage and computational complexity of LORA. At the encoder of LORA, there is only a measurement matrix, which has $(2N_tN_c)^2 \times CR$ parameters. At the decoder of LORA, each layer has a learning rate parameter and a regularization learning module, which has $(4N_tN_c \times N_m)$ parameters. Thus, the total number of parameters of LORA is $((2M + 4TN_m) \times N_tN_c + T)$. Thanks to parameter sharing among regularization learning modules of different layers, the number of trainable parameters of LORA can be reduced to $((2M + 4N_m)N_tN_c + T)$. The number of parameters of CsiNet+, TiLISTA-Joint and LORA at the encoder and decoder are also shown in TABLE VI for both CR = 1/4 and CR = 1/64. The results show that unfolding-based methods have slightly fewer parameters at the encoder, but significantly more parameters are needed at the decoder. It is because the convolution operator in CsiNet+ shares parameters and has less connections with the output of the former layer than MLP. However, since the BS usually has large storage, it is desirable to employ a larger model on the decoder side.

| Method        | CR=1/4 | CR=1/64 |
|---------------|--------|---------|
| CsiNet+       |        |         |
| TiLISTA-Joint |        |         |
| LORA(ours)    |        |         |
| Encoder       | 1,048,772 | 1,099,936 |
| Decoder       | 65,732  | 86,856  |

### TABLE VI

The Number of Parameters of Different Methods in the Encoder and Decoder, Respectively

In this sub-section, the proposed quantization method is evaluated. We first compare jointly training LORA and the quantizer-dequantizer parameters with training the quantizer-dequantizer on a well-trained LORA. The NMSE of these two approaches are 0.0053 and 0.0069, respectively, for CR = 1/4 in the RMANLOS scenario. Hence, as one would expect, joint training is preferable as the NN parameters adapt to the quantization effect. We next compare three methods for four different CR values and two different bit levels in the RMANLOS scenario. The NMSE performance of the quantization methods are presented in Fig. 11. QAT is the baseline method for comparison, while the dashed line without marks refers to LORA without quantization. It can be seen that LSQ and LSZQ both outperform QAT. Moreover, the performance increases with the number of learnable parameters in the quantization module; that is LSZQ outperforms LSQ. Comparison with LORA without quantization benchmark shows that quantization decreases the performance of LORA, especially when the number of quantization bits is small. The results not only verify that the scale and zero point value are essential parameters of the quantization, but also demonstrate the effectiveness of making these parameters learnable. As expected, the performance of all three methods improve with the number

### TABLE VII

The Computation Costs of Different Methods for CR = 1/64 in the RMANLOS Scenario

| Method        | Training time (sec.) | Test time (sec.) |
|---------------|----------------------|------------------|
| CsiNet+       | 64,307               | 6,712            |
| TiLISTA-Joint | 109,688              | 10,432           |
| LORA(ours)    | 29,337               | 3.54             |
Fig. 12. The achievable rate versus SNR for (a) CR = 1/64 and (b) CR = 1/4 in the UMANLOS scenario.

Fig. 13. The achievable rate versus SNR with 4-bit quantization with different methods in the RMANLOS scenario.

of quantization bits. In addition, the performance of LSQ and LSZQ increase with the increase in CR more significantly in 8-bit quantization than 4-bit quantization. It is also interesting that the improvement of LSQ and LSZQ compared to QAT also increases with the number of quantization bits. This phenomenon may be attributed to the fact that the impacts of scale and zero point parameters are amplified due to the increase in the number of quantization bits.

F. Communication Performance Analysis

So far, we have mainly analyzed the NMSE performance of LORA. However, in practical systems, the goal of providing accurate CSI feedback is to improve the communication performance. Hence, in this section, we consider the average achievable user rate to evaluate the performance, where zero-forcing (ZF) is used as the precoding algorithm. ZF is applied independently for all sub-carriers to calculate the corresponding rates, and the results are averaged over the number of sub-carriers. The details of how these rates are evaluated can be found in [63] and [64]. We generate 8000 samples for training and 2000 samples for the test. The number of sub-carriers is set to 512, while the other variables remain the same as in Section V-A.

The performance of TiLISTA-Joint, CsiNet+, and the proposed LORA method versus SNR for CR = 1/64 and CR = 1/4 in the UMANLOS scenario are shown in Fig. 12 (a) and Fig. 12 (b), respectively. The dotted line represents the achievable rate with perfect CSI knowledge, which can be regarded as an upper bound. From the figure, it can be seen that the proposed method outperforms the compared methods for both CR values, while the improvement with respect to TiLISTA-Joint is more significant in the high compression case. Moreover, the gaps between LORA and the upper bound are uniformly narrow across all SNRs for both CR = 1/64 and CR = 1/4. Since the overhead is large for CR = 1/4, the gap between LORA and TiLISTA-Joint is small, and both of which perform close to the upper bound. The achievable average rate results corroborate the comparisons based on the NMSE, and verify the superiority of LORA.

To further verify the effectiveness of the proposed quantization method, we evaluate the rate performance of the three quantization methods considered in Fig. 11. According to the results presented in Fig. 13, the quantization methods with learnable parameters outperform the method without learnable parameters, also in terms of the rate performance. We also observe that the gap between the rates achieved by the LSQ and LSZQ is rather small, which may attribute to the fact that the rate is a logarithmic function of the SNR, which shrinks the relatively larger gap observed in terms of the NMSE.

VI. CONCLUSION

In this paper, a model-driven DL method, called LORA, has been proposed for efficient CSI feedback in FDD massive MIMO systems. LORA is constructed by unfolding an iterative optimization algorithm with learnable parameters. Specifically, the derivative of the regularization term of the optimization problem is parameterized as a MLP to automatically and directly extract the characteristics of CSI instead of using the fixed conventional $l_1$-norm. Besides, a scale and zero point value learnable quantization method with the end-to-end training was proposed to ease the performance decay caused by quantization. Numerical results not only show effects of various components of LORA, supporting the presented architecture, but also demonstrate the superiority and robustness of this architecture with respect to existing techniques in the literature. It has been also shown that LORA with the proposed quantization method can be effective at different bit levels, providing flexibility in terms of the available feedback channel capacity.

REFERENCES

[1] F. Rusek, D. Persson, B. K. Lau, E. G. Larsson, T. L. Marzetta, and F. Tufvesson, “Scaling up MIMO: Opportunities and challenges with very large arrays,” IEEE Signal Process. Mag., vol. 30, no. 1, pp. 40–60, Jan. 2013.
[2] S. Lakshminarayana, M. Assaad, and M. Debbah, “Coordinated multicell beamforming for massive MIMO: A random matrix approach,” *IEEE Trans. Inf. Theory*, vol. 61, no. 6, pp. 3387–3412, Jun. 2015.

[3] J. Sun, Y. Zhang, J. Xue, and Z. Xu, “Learning to search for MIMO detection,” *IEEE Trans. Wireless Commun.*, vol. 19, no. 11, pp. 7571–7584, Nov. 2020.

[4] S. Ji and M. Li, “CLNet: Complex input lightweight neural network designed for massive MIMO CSI feedback,” *IEEE Wireless Commun. Lett.*, vol. 10, no. 6, pp. 2318–2322, Oct. 2021.

[5] M. S. Sim, J. Park, C. B. Chae, and R. W. Heath Jr., “Compressed channel feedback for correlated massive MIMO systems,” *J. Commun. Netw.*, vol. 18, no. 1, pp. 95–104, Feb. 2016.

[6] Z. Gao, L. Dai, S. Han, Z. Wang, and L. Hanzo, “Compressive sensing techniques for next-generation wireless communications,” *IEEE Wireless Commun.*, vol. 25, no. 3, pp. 144–153, Jun. 2018.

[7] P.-H. Kuo, H. T. Kung, and P.-A. Ting, “Compressive sensing based channel feedback protocols for spatially-correlated massive antenna arrays,” in *Proc. IEEE Wireless Commun. Netw. Conf. (WCNC)*, Apr. 2012, pp. 492–497.

[8] L. Lu, G. Y. Li, D. Qiao, and W. Han, “Sparsity-enhancing basis for compressive sensing based channel feedback in massive MIMO systems,” in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, Dec. 2014, pp. 1–6.

[9] P. Cheng and Z. Chen, “Multidimensional compressive sensing based analog CSI feedback for massive MIMO-OFDM systems,” in *Proc. IEEE 80th Veh. Technol. Conf. (VTC-Fall)*, Sep. 2014, pp. 1–6.

[10] M. Jankowski, D. Gunduz, and K. Mikolajczyk, “AirNet: Neural network transmission over the air,” in *Proc. IEEE 80th Veh. Technol. Conf. (VTC-Fall)*, Sep. 2014, pp. 1–6.

[11] G. Liu, Z. Hu, L. Wang, J. Xue, H. Yin, and D. Gesbert, “Spatio-temporal neural network for channel prediction in massive MIMO-OFDM systems,” *IEEE Trans. Commun.*, vol. 70, no. 1, pp. 2184–2199, Oct. 2019.

[12] M. Soltani, V. Pourrahimi, A. Mirzaei, and H. Sheikhzadeh, “Deep learning-based channel estimation and feedback for FDD multi-user massive MIMO systems,” *IEEE Trans. Signal Process.*, vol. 62, no. 2, pp. 3261–3271, Jun. 2014.

[13] D. Gündüz, P. de Kerret, N. D. Sidiropoulos, D. Gesbert, C. R. Murthy, and M. van der Schaar, “Machine learning in the air,” *IEEE J. Sel. Areas Commun.*, vol. 37, no. 10, pp. 2184–2199, Oct. 2019.

[14] J. Xia and D. Gunduz, “Meta-learning based beamforming design for spatially-correlated massive antenna arrays,” in *Proc. IEEE Wireless Commun. Netw. Conf. (WCNC)*, Apr. 2012, pp. 492–497.

[15] L. Lu, G. Y. Li, D. Qiao, and W. Han, “Sparsity-enhancing basis for compressive sensing based channel feedback in massive MIMO systems,” in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, Dec. 2014, pp. 1–6.

[16] P. Cheng and Z. Chen, “Multidimensional compressive sensing based analog CSI feedback for massive MIMO-OFDM systems,” in *Proc. IEEE 80th Veh. Technol. Conf. (VTC-Fall)*, Sep. 2014, pp. 1–6.

[17] A. C. Metzler, A. Maleki, and R. G. Baraniuk, “From denoising to compressed sensing,” *IEEE Trans. Inf. Theory*, vol. 62, no. 9, pp. 5117–5144, Sep. 2016.

[18] X. Rao and V. K. N. Lau, “Distributed compressive CSIT estimation and feedback for FDD multi-user massive MIMO systems,” *IEEE Trans. Signal Process.*, vol. 62, no. 12, pp. 3261–3271, Jun. 2014.

[19] D. Gündüz, P. de Kerret, N. D. Sidiropoulos, D. Gesbert, C. R. Murthy, and M. van der Schaar, “Machine learning in the air,” *IEEE J. Sel. Areas Commun.*, vol. 37, no. 10, pp. 2184–2199, Oct. 2019.

[20] J. Xia and D. Gunduz, “Meta-learning based beamforming design for MISO downlink,” in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, Jul. 2022.

[21] Z. Hu, J. Guo, G. Liu, H. Zheng, and J. Xue, “MRFNet: A deep learning-based channel estimation and feedback approach for time-varying massive MIMO channels,” *IEEE Wireless Commun. Lett.*, vol. 8, no. 2, pp. 416–419, Apr. 2019.

[22] J. Li, J. Guo, C. Wen, S. Jin, and G. Y. Li, “Overview of deep learning-based CSI feedback in massive MIMO systems,” *IEEE Trans. Commun.*, vol. 70, no. 12, pp. 8017–8045, Dec. 2022.

[23] C. Wen, W. Shih, and S. Jin, “Deep learning for massive MIMO CSI feedback,” *IEEE Wireless Commun. Lett.*, vol. 7, no. 5, pp. 748–751, Oct. 2018.

[24] J. Guo, C. Wen, S. Jin, and G. Y. Li, “Convolutional neural network-based multiple-rate compressive sensing for massive MIMO CSI feedback: Design, simulation, and analysis,” *IEEE Trans. Wireless Commun.*, vol. 19, no. 4, pp. 2827–2840, Apr. 2020.

[25] Z. Hu, J. Guo, G. Liu, H. Zheng, and J. Xue, “MRFNet: A deep learning-based CSI feedback approach of massive MIMO systems,” *IEEE Trans. Commun.*, vol. 75, no. 10, pp. 3430–3431, Oct. 2021.

[26] Z. Hu, J. Guo, G. Liu, H. Zheng, and J. Xue, “MRFNet: A deep learning-based CSI feedback approach of massive MIMO systems,” *IEEE Trans. Veh. Technol.*, vol. 71, no. 10, pp. 11216–11221, Oct. 2022.

[27] Y. Sun, W. Xu, L. Li, N. Wang, G. Y. Li, and X. You, “A lightweight deep network for efficient CSI feedback in massive MIMO systems,” *IEEE Wireless Commun. Lett.*, vol. 10, no. 8, pp. 1840–1844, Aug. 2021.

[28] Y. Sun, W. Xu, L. Li, N. Wang, G. Y. Li, and X. You, “A lightweight deep network for efficient CSI feedback in massive MIMO systems,” *IEEE Wireless Commun. Lett.*, vol. 10, no. 8, pp. 1840–1844, Aug. 2021.

[29] Y. Sun, W. Xu, L. Li, N. Wang, G. Y. Li, and X. You, “A lightweight deep network for efficient CSI feedback in massive MIMO systems,” *IEEE Wireless Commun. Lett.*, vol. 10, no. 8, pp. 1840–1844, Aug. 2021.

[30] Y. Sun, W. Xu, L. Li, N. Wang, G. Y. Li, and X. You, “A lightweight deep network for efficient CSI feedback in massive MIMO systems,” *IEEE Wireless Commun. Lett.*, vol. 10, no. 8, pp. 1840–1844, Aug. 2021.
