Learning-Based MIMO Channel Estimation under Practical Pilot Sparsity and Feedback Compression

Mason del Rosario and Zhi Ding

***Abstract*—Wireless links using massive MIMO transceivers are vital for next generation wireless communications networks. Precoding in Massive MIMO transmission requires accurate downlink channel state information (CSI). Many recent works have effectively applied deep learning (DL) to jointly train UE-side compression networks for delay domain CSI and a BS-side decoding scheme. Vitally, these works assume that the full delay domain CSI is available at the UE, but in reality, the UE must estimate the delay domain based on a limited number of frequency domain pilots. In this work, we propose a linear pilot-to-delay estimator (P2DE) that acquires the truncated delay CSI via sparse frequency pilots. We show the accuracy of the P2DE under frequency downsampling, and we demonstrate the P2DE's efficacy when utilized with existing CSI estimation networks. Additionally, we propose to use trainable compressed sensing (CS) networks in a differential encoding network for time-varying CSI estimation, and we propose a new network, MarkovNet-ISTA-ENet (MN-IE), which combines a CS network for initial CSI estimation and multiple autoencoders to estimate the error terms. We demonstrate that MN-IE has better asymptotic performance than networks comprised of only one type of network.**

***Index Terms*—Massive MIMO, Deep learning CSI, efficient feedback, CSI estimation.**

I. Introduction

Large scale multiple-input multiple-output (MIMO) technologies are critical to achieving high link capacity in modern wireless networks [1]. To this end, MIMO base stations (BS) require accurate downlink channel state information (CSI) for transmit precoding and beamforming. While uplink-downlink reciprocity in TDD systems [2]–[4] often simplifies the task of downlink CSI acquisition at BS, the predominant approach relies on downlink CSI estimation and feedback from UEs in UE-specific precoding and/or beamforming at BS.

A number of recent works have studied deep learning (DL) for CSI compression by UE and subsequent estimation by the BS. Recent advances include the use of convolutional neural networks (CNNs) as autoencoder [5]–[8], the integration of magnitude-reciprocity between uplink/downlink CSIs for decoding [9], and the exploitation of temporal CSI coherence [10], [11]. These successes motivate further investigative efforts into learning-based CSI estimation to overcome several remaining practical challenges in high rate massive MIMO networking.

This work addresses two major practical issues in MIMO CSI feedback compression.

1) **Frequency domain pilots for delay domain feedback**: First, many existing DL frameworks rely on the condition that full downlink CSI or CSI estimates in time-frequency domain are available at the UE. However, practical wireless standards such as 4G/5G by 3GPP, only provide sparse position pilot reference configurations in time-frequency domain. See, e.g., [12]. With sparse pilots, only sparse downlink CSI is available at the UE instead of full time-frequency CSI. Therefore, practical DL algorithms for downlink CSI estimation and decoding must start with low-resolution, undersampled CSI in time-frequency domain under potential noisy conditions without assuming full ground truth CSI.

   Several recent works have addressed the problem of pilot-based CSI estimation. In [13], the authors propose a two-stage approach to pilot-based CSI estimation: 1) coarse estimation of pilots via spatial correlation between adjacent subcarriers and 2) pilot CSI refinement via a UE-side CNN. In [14], the authors propose a

M. del Rosario and Z. Ding are with the Department of Electrical and Computer Engineering, University of California at Davis, Davis, CA 95616 USA (e-mail: mdelrosa@ucdavis.edu, zding@ucdavis.edu).

This material is based upon work supported by the National Science Foundation under Grant No. 2029027 and No. 2002937

fully-connected network (FCN) to adaptively design pilots for UE-side channel estimation. In [15], the authors propose to train FCNs to design pilots and further propose to reduce pilot overhead by gradually pruning the FCNs, thereby reducing the pilot overhead and improving spectral efficiency. The proposed FCN outperforms a conventional least-squares approach to pilot-based CSI estimation.

Importantly, the above works focus on pilot-based CSI estimation and feedback stages in the traditional spatial-frequency domain. On the other hand, other recent works on deep learning based CSI feedback have demonstrated the benefits of compressing CSI in the delay domain (i.e., the IFFT of the frequency domain). Taking advantage of the sparsity in multipath channel delays, transforming CSI into delay domain makes it possible to compress CSI feedback through simple truncation before encoding and feedback. This step improves feedback efficiency substantially (see [5], [10], for example). Thus, explicitly linking the sparsely placed frequency domain pilots in practical wireless systems to the dominant delay domain CSI represents a critical step in deep-learning based CSI feedback framework.

2) **Improving temporal correlation-based networks**: A second practical consideration is the need to exploit CSI temporal coherence without significantly increasing DL complexity (e.g., via LSTM layers [10]). Our prior work has adopted a simple yet effective differential encoder, MarkovNet, based on an approximate first order Markov model of time-varying CSI [11]. MarkovNet relied on CNN autoencoder architecture for each timeslot, a design choice made in many works on trainable CSI feedback compression [5], [7], [9]. Yet recent work in CSI estimation has demonstrated that trainable compressed sensing (CS) networks can yield state-of-the-art performance [16]. For periodic CSI estimation and feedback, we propose a novel architecture that integrates the differential encoding concept of MarkovNet with a trainable CS network.

To summarize our works that target the above practical considerations in real-world wireless networks, we highlight our major contributions in this work as follows:

- **Pilots-to-Delay Estimator (P2DE)**: Based on a limited number of pilot-based estimates, we propose an accurate linear estimator of the truncated delay-domain CSI at the UE. We begin by quantifying the sparsity of the delay domain CSI, allowing us to specify the required amount of pilot downsampling in the frequency domain (see Section II-C, Figure 2). Next, to bridge the gap between prior works in feedback compression using delay domain CSI and 3GPP specifications which specify pilot locations in the frequency domain, we propose the Pilots-to-Delay Estimator (P2DE), which relates the pilot-based downsampled frequency domain CSI to the the truncated delay domain CSI (see Section III-A). To demonstrate the practicality of the P2DE, we outline a parameterized pilot allocation in the time-frequency resource grid based on CSI-RS/DMRS locations (see Section III-B). Using the P2DE as the input to a range of deep learning-based CSI compression networks, we show that this estimator provides a suitable surrogate for ground-truth delay domain CSI under noise-free and noisy conditions (see Section VI-A, Figures 8 and 9).
- **CS-based Differential Encoding with Pilot-based CSI**: Using the proposed P2D estimates at the UE, we propose to encode and feed back the estimation error. To compress the error terms, we compare unrolled optimization networks, which enable trainable compressive sensing algorithms via deep learning, with autoencoder networks, which have been commonly used in CSI feedback literature. We show that a differential network combining both unrolled compressed sensing networks and autoencoders can outperform prior autoencoder-based approaches to differential encoding.

II. Practical Channel Estimation Problem

This section provides background information regarding the modeling of OFDM CSI in MIMO

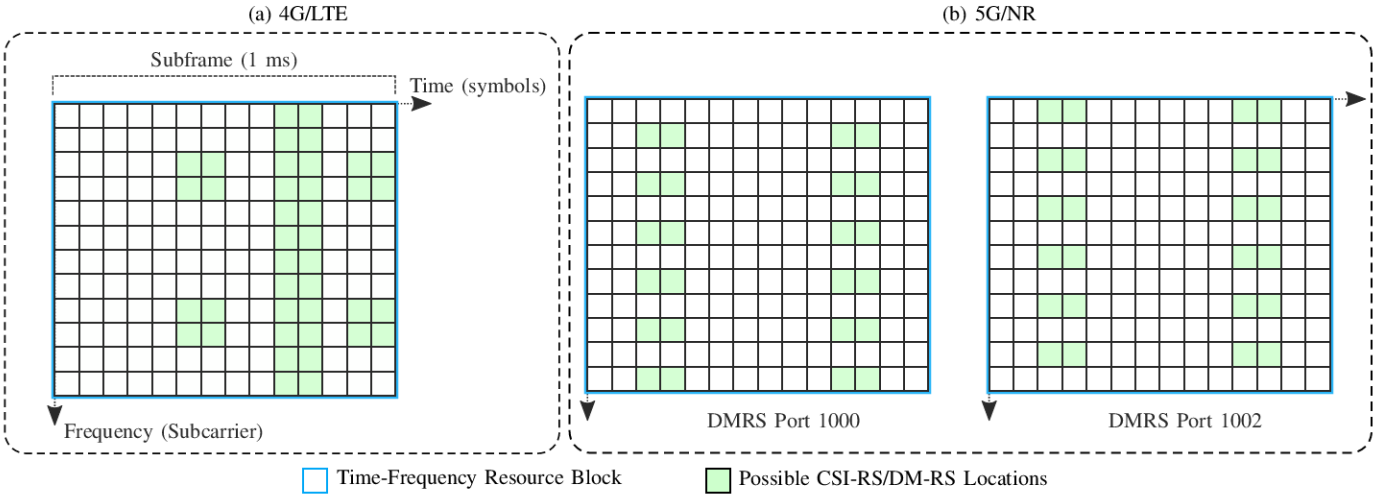

Fig. 1: (a) LTE resource blocks with CSI-RS locations. (b) 5G NR resource blocks with DM-RS locations.

networks. Section II-A describes the system model for an OFDM network. Section II-B details how pilots are used in the LTE/NR specifications to estimate downlink CSI. Section II-C provides an empirical analysis of the sparsity in MIMO CSI.

A. OFDM Downlink System Model

Without loss of generality, we consider a single-cell MIMO system with $N_b \gg 1$ antennas at the BS serving multiple UEs, each with a single receive antenna. The network operates under orthogonal frequency division multiplexing (OFDM) with bandwidth divided into N_f uniform subcarriers. Focusing on the downlink of a given OFDM symbol, the received signal on the m-th subcarrier/subband for the i-th UE out of N_{UE} UEs is given by

$$y_{m,i} = \mathbf{h}_{m,i}^H \mathbf{w}_{m,i} x_{m,i} + n_{m,i}, \tag{1}$$

for $m \in \{1, 2, \cdots, N_f\}$ and $i \in \{1, 2, \cdots, N_{\text{UE}}\}$, and $\mathbf{h}_{m,i} \in \mathbb{C}^{N_b \times 1}$ is the downlink CSI vector of the m-th subcarrier, $\mathbf{w}_{m,i} \in \mathbb{C}^{N_b \times 1}$ denotes the precoding vector, $x_{m,i} \in \mathbb{C}$ is the $m-$th data symbol of the OFDM symbol, and $n_m \in \mathbb{C}$ denotes corresponding subcarrier additive noise, and $(\cdot)^H$ denotes conjugate transpose. The downlink CSI spatial-frequency matrix for the i-th user is $\mathbf{H}_i = [\mathbf{h}_{1,i}\ \mathbf{h}_{2,i}\ \cdots,\ \mathbf{h}_{N_f,i}] \in \mathbb{C}^{N_b \times N_f}$.

Throughout this work, we present results from the perspective of a single UE feedback though the proposed framework is directly applicable for multiple UEs. For ease of notation, we omit the UE-specific subscript when referring to CSI matrices (e.g., $\mathbf{H}$ rather than $\mathbf{H}_1$). Note that this choice does not limit the applicability of the proposed methods to a multi-user system as described above, and in future works, there may be opportunities to improve network performance via CSI correlation between different users.

B. Sparse Pilots in Practical Networks

Prior works using deep learning algorithms for pilot-based CSI estimation generally discuss how pilots are allocated along the spatial-frequency dimensions (i.e., transmitter antennas and subcarriers). In order to make pilot estimation practical, we discuss below how pilots are allocated to time-frequency resources per 3GPP protocols, and later (i.e., Section III-B) we will explicitly show how we allocate the spatial-frequency domain pilots to logical antenna ports in the time-frequency domain.

In most deployed networks, transmitters provide pilot reference signals for the receivers to estimate the physical wireless CSI. Furthermore, in well known 3GPP standard cellular networks such as 4G (LTE) and 5G (NR) systems, BS facilitate downlink CSI estimation by allocating sparse pilots for N_b antennas and N_f subcarriers to a small number of time-frequency positions on $G_t \times G_f$ grids to preserve spectrum efficiency (where G_t is the number of symbols per subframe and G_f is the number of subcarriers to be allocated).

Consider the 3GPP placement of DMRS of PDSCH specified in §7.4.1.1.2 [12]. The downlink pilots (i.e., demodulation reference signal or

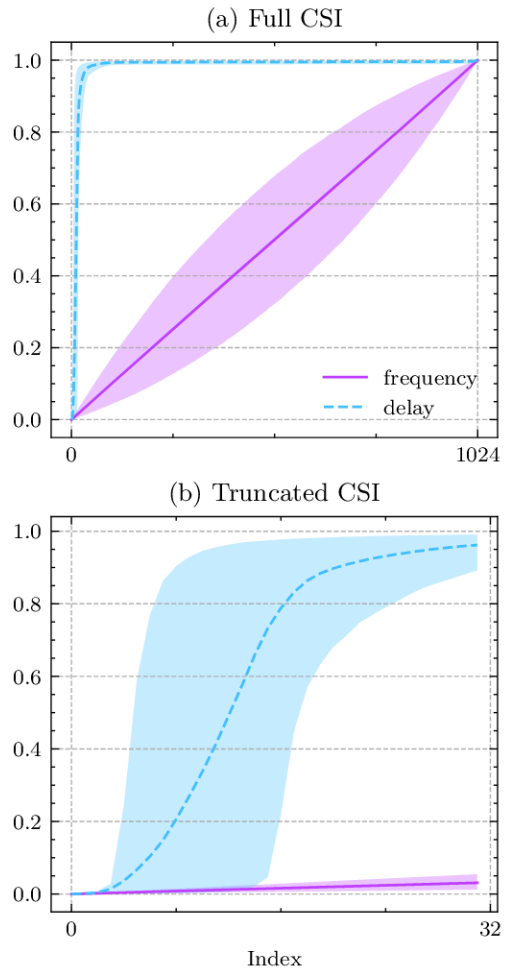

Fig. 2: Energy CDF for 5000 CSI samples of 32 antennas and 1024 subcarriers, generated from COST2100 outdoor models described in Section [17]. Mean percentage of energy in CSI matrix up to index is shown with 90% confidence intervals. The index denotes the amount of energy accounted for up to the corresponding frequency/delay element. The truncated angular-delay CSI contains a mean energy of 96.2% (c.i. 89.2%, 99.1%), while the truncated frequency-spatial CSI only contains a mean energy of 3.1% (c.i. 1.2%, 5.4%).

DMRS) decoding the Physical Downlink Shared Channel (PDSCH) must share resources with user data. The pilots for different physical antennas are assigned to "antenna ports," i.e., mutually orthogonal subframes on time-frequency resource grids. The number of antenna ports being used is determined by system configuration parameters, including a DMRS length parameter (1 or 2) and a DMRS configuration Type (1 or 2). When the DMRS legnth is 1, the number of antenna ports available is either 4 (Type 1 DMRS configuration) or 6 (Type 2 DMRS configuration). When DMRS length is 2, the number of antenna ports available is either 8 (Type 1 DMRS configuration) or 12 (Type 2 DMRS configuration). As seen from Figure 1(b), which shows an antenna port under Type 1 DMRS configuration and DMRS length of 2, the antenna ports include a relatively small number of pilots in each slot. Therefore, for any given slot, the UE can only directly estimate a few sparse elements of the downlink CSI matrix $\mathbf{H}$, instead of the full $\mathbf{H}$ considered in most existing works.

Without loss of generality, we consider a well defined case when the sparsity follows a regular downsampling pattern on the $N_b \times N_f$ grids such that the UE can obtain a downsampled CSI matrix $\mathbf{H}_d \in \mathbb{C}^{N_b \times M_f}$ where sampling of $M_f < N_f$ takes place along the frequency dimension. The rows of $\mathbf{H}$ are related to the rows of $\mathbf{H}_d$ as follows,

$$\mathbf{H} = \begin{bmatrix} \boldsymbol{\eta}_1 \\ \boldsymbol{\eta}_2 \\ \vdots \\ \boldsymbol{\eta}_{N_b} \end{bmatrix} \quad \mathbf{H}_d = \begin{bmatrix} \boldsymbol{\eta}_{d,1} \\ \boldsymbol{\eta}_{d,2} \\ \vdots \\ \boldsymbol{\eta}_{d,N_b} \end{bmatrix}, \tag{2}$$

where $\boldsymbol{\eta}_{d,i}$ is downsampled by matrices denoting CSI-RS/DMRS reference positions for N_b antenna ports $\mathbf{P}_i$ of size $N_f \times M_f$

$$\boldsymbol{\eta}_{d,i} = \boldsymbol{\eta}_i \mathbf{P}_i, \; i = 1, \; \cdots, \; N_b. \tag{3}$$

The product (3) represents a downsampled frequency domain CSI for antenna j. Note that each column of $\mathbf{P}_i$ is a one hot vector of all zeros except for a single element equal to 1, indicating the selected subcarrier. In most cases, the downsampling matrix $\mathbf{P}_i$ may be chosen from one of a few predefined matrices.

We now face the challenge of recovering the full downlink CSI $\mathbf{H}$ at BS based on low rate feedback from the UE, even though the UE itself only has a pilot-based estimate of the downsampled CSI matrix $\mathbf{H}_d$.

C. Downlink CSI Sparsity Analysis

Prior works have recognized the low delay spread of most downlink CSI within N_t columns after IFFT. This represents a structured sparsity that can be exploited in CSI feedback compression. Figure 2 demonstrates such sparsity of CSI generated with the COST2100 outdoor channel model [17] of 32

antennas and 1024 subcarriers (more detailed system parameters are given in Section VI). In this case, the first 32 columns of the delay domain CSI contain roughly 96% of CSI energy, meaning we can safely truncate the delay domain CSI (i.e., keeping the delay domain CSI in the first 32 delay elements and discarding the rest). We refer to the delay-domain CSI after truncation as the "truncated delay-domain CSI." Leveraging this sparsity, a 32/1024 reduction in feedback via downsampling should be possible.

In the next section, we present an algorithm to directly estimate angular-delay domain CSI using sparse spatial-delay domain pilots.

III. Linear Prediction of Delay-domain CSI via Frequency-domain Pilots

Using the limited number of frequency domain pilots available at the UE, we can estimate the truncated delay domain data. This delay domain estimate is directly compatible with the commonly used CSI basis in prior deep learning based CSI compression works [5], [11], which have demonstrated high estimation accuracy under substantial compression.

A. P2DE: Pilots-to-delay Estimator

Here, we describe the linear estimator for the truncated delay domain CSI using the pilot-based frequency domain CSI. Note that $\boldsymbol{\eta}_i$ is one of the rows of the spatial-frequency matrix $\mathbf{H}$. Consider the case where downsampling is performed along the frequency axis such that M_f subcarriers of the original N_f subcarriers remain. Downsampling is done by applying the pilot matrix $\mathbf{P} \in \mathbb{C}^{N_f \times M_f}$ to the frequency domain vector $\boldsymbol{\eta}_i$, resulting in the pilot vector $\boldsymbol{\eta}_{d,i} \in \mathbb{C}^{M_f}$.

To relate the frequency and delay domain, denote the Discrete Fourier Transform (FFT) matrix $\mathbf{F} \in \mathbb{C}^{N_f \times N_f}$ with which

$$\tilde{\boldsymbol{\eta}}_i \mathbf{F} = \boldsymbol{\eta}_i, \quad i \in \{1, \ldots, N_b\}$$

Note that $\tilde{\boldsymbol{\eta}}_i$ is the time/delay domain CSI row vector. Applying the pilot sampling matrix to both sides, we have

$$\begin{aligned} \tilde{\boldsymbol{\eta}}_i \mathbf{F} \mathbf{P}_i &= \boldsymbol{\eta}_i \mathbf{P}_i \\ \tilde{\boldsymbol{\eta}}_i \mathbf{Q}_i &= \boldsymbol{\eta}_{d,i}, \end{aligned}$$

for $\mathbf{Q}_i = \mathbf{F}\mathbf{P}_i \in \mathbb{C}^{N_f \times M_f}$.

Our previous experiments (see Figure 2) confirm the phenomenon reported in other prior works [5] that for most wireless CSI models, the delay domain CSI vectors $\tilde{\boldsymbol{\eta}}_i$ exhibit a clear sparsity as a result of short multipath delay spread. Without loss of generality, we can characterize the sparsity of $\tilde{\boldsymbol{\eta}}_i$ by noting that its trailing elements are approximately zero and can be replaced by zeros without introducing significant CSI estimation error.

Given the sparsity of CSI in the delay domain, we may truncate $\mathbf{Q}_i$ to the first N_t rows and restrict our attention to the truncated delay domain vector, $\tilde{\boldsymbol{\eta}}_c \in \mathbb{C}^{1 \times N_t}$. Thus, denoting the first N_t rows of $\mathbf{Q}_i$ by $\mathbf{Q}_{c,i}$ and defining

$$\tilde{\boldsymbol{\eta}}_i = [\tilde{\boldsymbol{\eta}}_{c,i} \;\; \mathbf{0}_{1 \times (N_f - N_t)}] \tag{4}$$

we have

$$\tilde{\boldsymbol{\eta}}_i \mathbf{Q}_i = \tilde{\boldsymbol{\eta}}_{c,i} \mathbf{Q}_{c,i}.$$

Now the task of downlink CSI estimation at the BS is transformed into the feedback and estimation of the lower dimensional vectors $\tilde{\boldsymbol{\eta}}_{c,i}, \quad i \in \{1, \cdots, N_b\}$.

From the downsampled pilot positions, the UE can estimate the CSI in frequency domain in the form of $\boldsymbol{\eta}_{d,i}, \quad i \in \{1, \cdots, N_b\}$, from which we can estimate the most significant part of time/delay domain CSI vector $\tilde{\boldsymbol{\eta}}_{c,i}$ based on the relationship of

$$\tilde{\boldsymbol{\eta}}_{c,i} \mathbf{Q}_{c,i} = \boldsymbol{\eta}_{d,i}, \quad \text{or} \quad \tilde{\boldsymbol{\eta}}_{c,i} = \boldsymbol{\eta}_{d,i} \mathbf{Q}_{c,i}^{\#} \tag{5}$$

where $\mathbf{Q}_{c,i}^{\#} = \mathbf{Q}_{c,i}(\mathbf{Q}_{c,i}\mathbf{Q}_{c,i}^H)^{-1}$ denotes the (pseudo)inverse of $\mathbf{Q}_{c,i}$ such that $\mathbf{Q}_{c,i}\mathbf{Q}_{c,i}^{\#} = \mathbf{I}$ under the condition that $N_t \leq M_f$. The estimator $\mathbf{Q}_{c,i}^{\#}$ relies solely on the downsampling matrix, $\mathbf{P}_i$, and the FFT matrix, $\mathbf{F}$, and we call this estimator the **P**ilots-**t**o-**D**elay **E**stimator (**P2DE**) since it uses sparse frequency domain pilots to estimate the delay domain CSI for feedback compression.

For convenience of notation, we form the truncated spatial delay domain CSI matrix $\tilde{\mathbf{H}}_\tau$ and its FFT, respectively, as

$$\tilde{\mathbf{H}}_\tau = \begin{bmatrix} \tilde{\boldsymbol{\eta}}_{c,1} \\ \tilde{\boldsymbol{\eta}}_{c,2} \\ \vdots \\ \tilde{\boldsymbol{\eta}}_{c,N_b} \end{bmatrix}, \qquad \mathbf{H}_\tau = \mathbf{F}_{N_b} \tilde{\mathbf{H}}_\tau \tag{6}$$

where $\mathbf{F}_{N_b} \in \mathbb{C}^{N_b \times N_b}$ is the DFT matrix. $\mathbf{H}_\tau$ is often known as the angular-delay domain CSI [5].

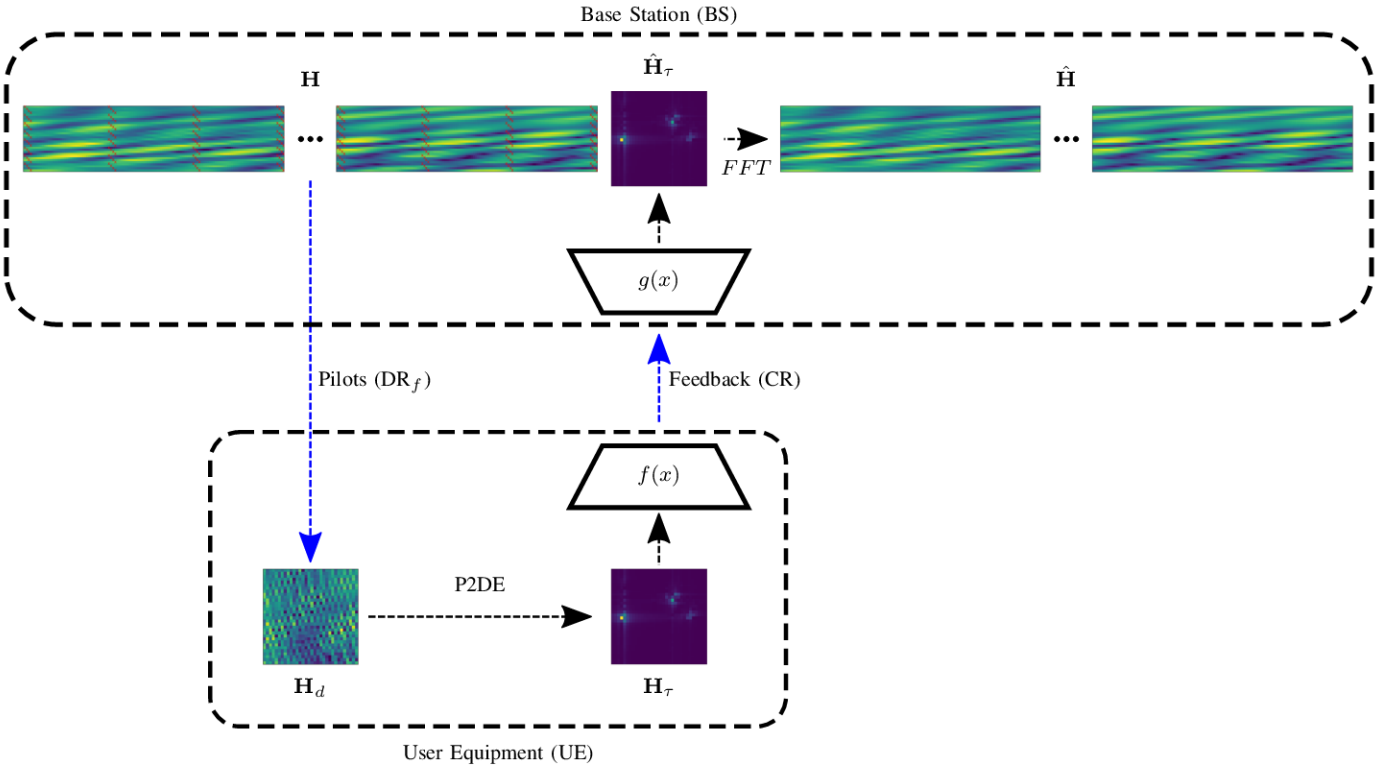

Fig. 3: Compressive CSI estimation based on linear P2D estimator. First, we use downlink pilots to generate a sparse, frequency domain CSI estimate of size $M_f << N_f$. We then apply the P2D estimator, $\mathbf{Q}_{N_t}^{\dagger}$ of (4), to establish the truncated delay domain CSI estimate. We train a learnable encoder, $f(x)$, and decoder, $g(x)$, to compress and decode the feedback, respectively. The BS recovers the frequency domain CSI from the decoded delay domain CSI estimate.

In practice, once $\tilde{\boldsymbol{\eta}}_{c,i}$ is recovered at the BS, we can directly obtain the full delay domain and frequency CSI vectors through zero padding and FFT

$$\boldsymbol{\eta}_i = [\tilde{\boldsymbol{\eta}}_{c,i} \;\; \mathbf{0}_{1\times(N_f-N_t)}]\mathbf{F} = \tilde{\boldsymbol{\eta}}_{c,i}\mathbf{F}_c \tag{7}$$

$$= \boldsymbol{\eta}_{d,i}\mathbf{Q}_{c,i}^{\#}\mathbf{F}_c \tag{8}$$

where $\mathbf{F}_c$ denotes the first N_t rows of the $N_f \times N_f$ DFT matrix $\mathbf{F}$. Using (8), the final frequency domain estimate $\hat{\mathbf{H}}$ can be recovered as

$$\hat{\mathbf{H}} = \begin{bmatrix} \hat{\boldsymbol{\eta}}_1 \\ \hat{\boldsymbol{\eta}}_2 \\ \vdots \\ \hat{\boldsymbol{\eta}}_{N_b} \end{bmatrix} = \begin{bmatrix} \boldsymbol{\eta}_{d,1}\mathbf{Q}_{c,1}^{\#} \\ \boldsymbol{\eta}_{d,2}\mathbf{Q}_{c,2}^{\#} \\ \vdots \\ \boldsymbol{\eta}_{d,N_b}\mathbf{Q}_{c,N_b}^{\#} \end{bmatrix} \mathbf{F}_c \in \mathbb{C}^{N_b\times N_t}. \tag{9}$$

In contrast with a "Compression Ratio (CR)" that is typically reported in the feedback stage, the P2DE is associated with a "Frequency Downsampling Ratio (DR$_f$)," which is given as

$$\text{DR}_f = \frac{M_f}{N_f}. \tag{10}$$

Figure 3 shows where the P2D estimator (P2DE) fits into the overall CSI feedback and estimation process.

B. Diagonal Pilot Patterns for LTE/5G Compatibility

In the 4G/LTE specification, downlink pilots for antenna ports are allocated to specific resource elements (CSI-RS) in the time-frequency resource grid [18], [19]. Similarly in 5G/NR, downlink pilots locations are reserved for resource elements called demodulation reference signal (DMRS) [**?**]. For a MIMO array, the different antenna ports are allocated to CSI-RS/DMRS locations in the resource grid, and multiple subframes might be necessary to

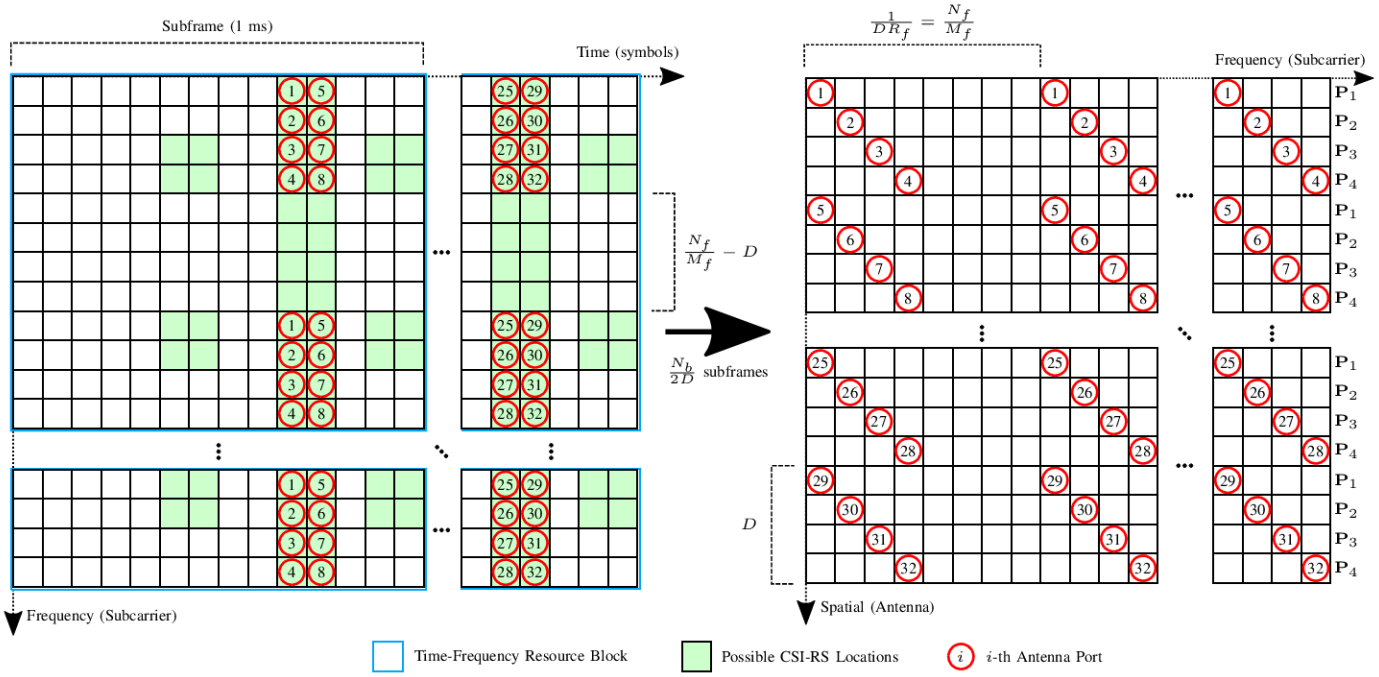

Fig. 4: (a) LTE Resource Blocks and CSI-RS locations where antenna port pilots are allocated. (b) Schematic for diagonal pilots with relevant parameters, size of diagonal D and frequency downsampling ratio DR_f. In this diagram, $N_b = 32, D = 4, \text{DR}_f = \frac{1}{8}$. The pilot matrix $\mathbf{P}_j$ indicates the downsampling pattern for the j-th element of the diagonal pattern. The number of subframes necessary to populate (b) is inversely proportional to D.

Algorithm 1 Pilots-to-delay Estimator (P2D) for Diagonal Pilot Pattern

Input: P2DE Matrices, $\mathbf{Q}_{c,j}^{\#},\ j \in \{1, \ldots, D\}$
Input: Pilot spatial-frequency CSI, $\mathbf{H}_d \in \mathbb{C}^{N_b \times M_f}$
Initialize: Spatial-delay CSI, $\tilde{\mathbf{H}}_\tau \in \mathbb{C}^{N_b \times N_t}$
Initialize: Angular-delay CSI estimate, $\mathbf{H}_\tau \in \mathbb{C}^{N_b \times N_t}$
for $i = 1, 2, \ldots, N_b$ **do**
    # Index for j-th pilot matrix
    $j = ((i-1) \bmod D) + 1$
    # Apply P2D to i-th antenna port
    $\boldsymbol{\eta}_{d,i} = \mathbf{H}_d(i, :)$
    $\tilde{\mathbf{H}}_\tau(i, :) = \boldsymbol{\eta}_{d,i} \mathbf{Q}_{c,j}^{\#}$
end for
 # Convert from spatial to angular
$\mathbf{H}_\tau = \mathbf{F}_{N_b} \tilde{\mathbf{H}}_\tau$
Return $\mathbf{H}_\tau$

acquire a downsampled CSI matrix. The number of subframes necessary depends on two parameters: 1) the size of the diagonal pattern, D, and 2) the number of antennas, N_b.

Figure 4(a) illustrates our proposed pilot allocation for a 4G/LTE time-frequency resource grid, and Figure 4(b) shows the resulting downsampling pattern in the spatial-frequency domain. Based on Figure 4, the benefit of diagonal pilot patterns becomes apparent, as the number of subframes needed to acquire the downsampled CSI matrix, $\mathbf{H}_d$ at the UE shrinks with increasing D. For example, the given diagonal size $D = 4$ requires 4 subframes (ms) to acquire $\mathbf{H}_d$, while $D = 1$ (i.e., no diagonal pattern or vertical columns of pilots) would require 16 subframes (ms) to acquire $\mathbf{H}_d$.

Similarly, Figure 5(a) illustrates the proposed pilot allocation for a 5G/NR scenario, and Figure 5(b) shows the resulting downsampling pattern in the spatial-frequency domain. The given diagonal size $D = 4$ requires 2 subframes (ms) to acquire $\mathbf{H}_d$, while $D = 1$ (i.e., no diagonal pattern or vertical columns of pilots) would require 8 subframes (ms) to acquire $\mathbf{H}_d$.

To utilize the P2DE with diagonal pilot patterns, it is necessary to account for different pilot matrices,

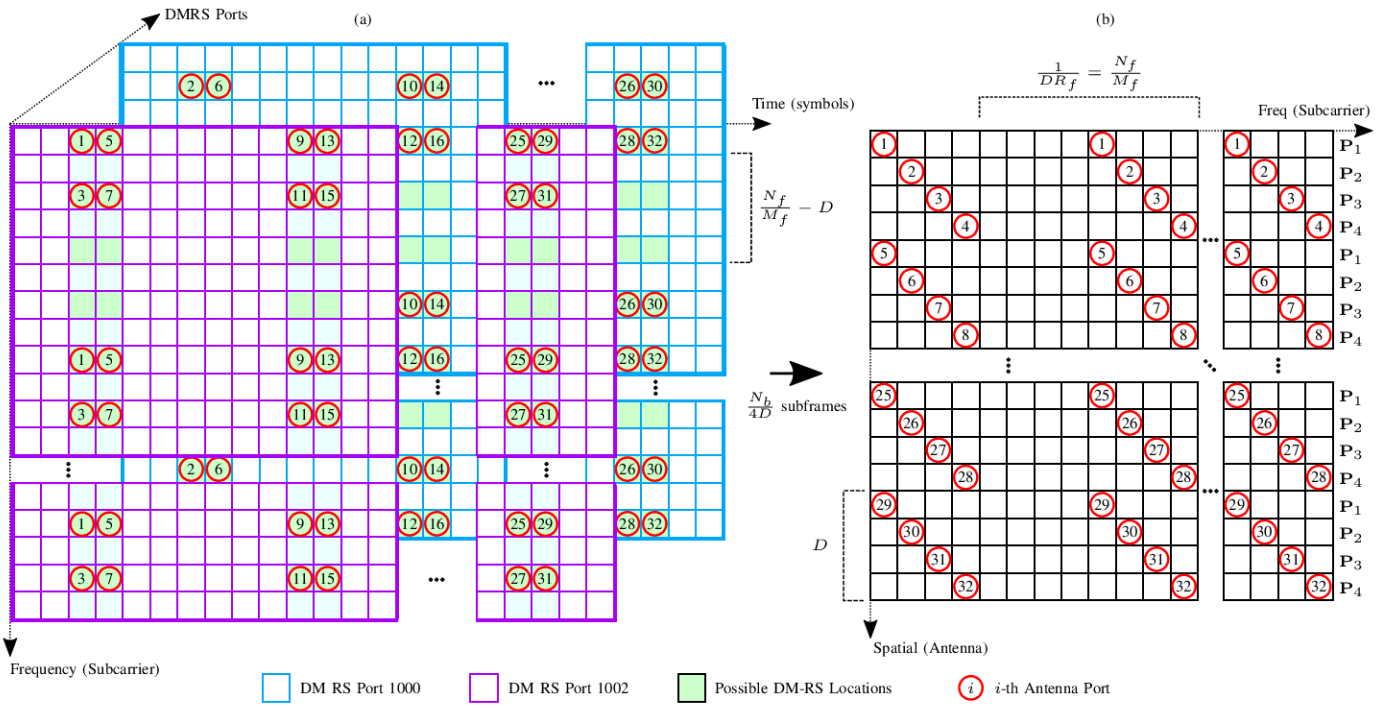

Fig. 5: (a) 5G NR Resource Blocks and DMRS locations where antenna port pilots are allocated. (b) Schematic for diagonal pilots with relevant parameters, size of diagonal D and frequency downsampling ratio DR_f. In this diagram, $N_b = 32, D = 4, \text{DR}_f = \frac{1}{8}$. The pilot matrix $\mathbf{P}_j$ indicates the downsampling pattern for the j-th element of the diagonal pattern. The number of subframes necessary to populate (b) is inversely proportional to D.

$\mathbf{P}_j$ for $j \in [1, \ldots, D]$, used with different antennas. These different pilot matrices result in D different P2DEs, $\mathbf{Q}^{\#}_{c,j}$. Algorithm 1 outlines the process for acquiring $\tilde{\mathbf{H}}_\tau$ by applying the P2DEs to $\mathbf{H}_d$.

C. Regularization of P2DE

Whenever pilot locations are spaced equidistantly and regularly, the pseudoinverse matrices $\mathbf{Q}^{\#}_{c,i} = \mathbf{Q}_{c,i}(\mathbf{Q}_{c,i}\mathbf{Q}^H_{c,i})^{-1}$ are typically well-conditioned and invertible. However, whenever the pilot placement is less regular (e.g., as in the proposed diagonal pattern of P2DE-Diag), then the pseudoinverse matrices $\mathbf{Q}^{\#}_{c,i}$ can be ill-conditioned, making the P2DE unstable. Consequently, the P2DE benefits from regularization of the matrix $\mathbf{Q}_{c,i}\mathbf{Q}^H_{c,i}$. This can be done via off-diagonal regularization (ODIR), where all off-diagonal elements are scaled down by a fixed constant. Denote $\mathbf{A} = \mathbf{Q}_{c,i}\mathbf{Q}^H_{c,i}$ as a matrix to be regularized where $A(j,k)$ is the element in the j-th row and k-th column. ODIR scales each off-diagonal element of $\mathbf{A}$ (i.e., $A(j,k)$ where $j \neq k$) uniformly by $(1+\delta)^{-1}$ where for $\delta \geq 0$. In practice, the choice of δ is determined empirically.

IV. Differential Encoding via Learned Compressed Sensing

To further improve the accuracy of CSI estimation under the P2DE, we can exploit the temporal coherence of the channel. Under typical circumstances, the channel does not change substantially for a given window of time, i.e. the coherence interval. Exploiting this coherence is beneficial from an information theoretic point of view [11]. Denote two subsequent timeslots within a coherence interval as t_1 and t_2, the entropy of the CSI at t_1 as $H(\tilde{\mathbf{H}}_{\tau,1})$ and the conditional entropy of the CSI at t_2 given t_1 as $H(\tilde{\mathbf{H}}_{\tau,2}|\tilde{\mathbf{H}}_{\tau,1})$. Prior work in time-varying CSI estimation has demonstrated that the conditional entropy is always lower than the entropy [11], i.e.,

$$H(\tilde{\mathbf{H}}_{\tau,2}|\tilde{\mathbf{H}}_{\tau,1}) \leq H(\tilde{\mathbf{H}}_{\tau,1}). \tag{11}$$

A reduction in entropy means a reduction in the rate of the compressed feedback, highlighting the

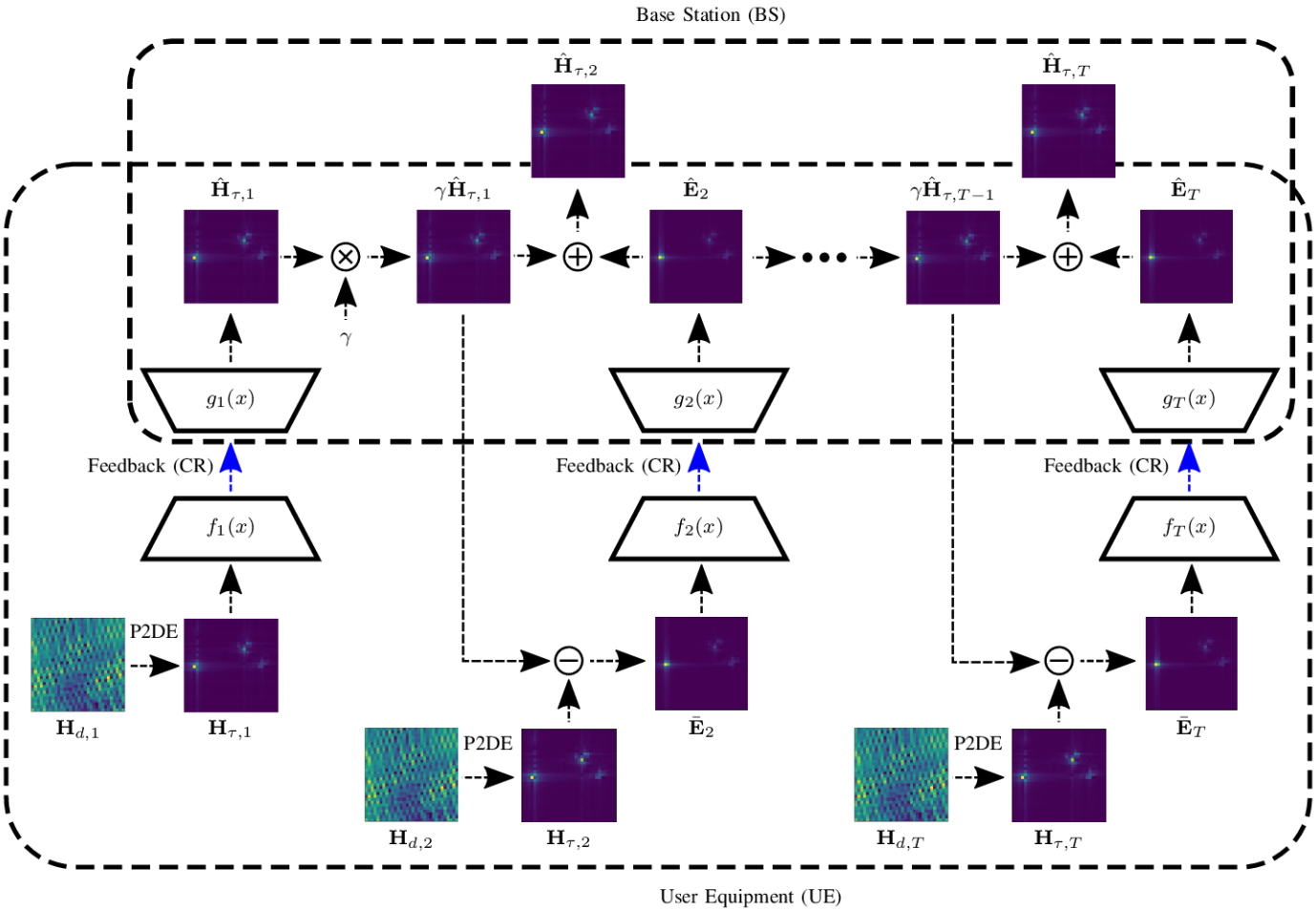

Fig. 6: Diagram of a CSI estimation network using compressed differential feedback based on the linear P2DE. First, downlink pilots are used to estimate a downsampled frequency domain CSI estimate, $\bar{\mathbf{H}}_t \in \mathbb{C}^{N_b \times M_f}$ where $M_f << N_f$ at the t-th timeslot. Then, the P2DE $\mathbf{Q}_{N_t}^{\#}$ of Algorithm 1 is applied to estimate $\tilde{\mathbf{H}}_t$. After P2DE, the learnable transforms $f_t(x)$ and $g_t(x)$ are used to compress and decode the feedback, respectively. For $t = 1$, the encoder/decoder are applied directly to $\tilde{\mathbf{H}}_1$. In all subsequent timeslots ($t > 1$), the differential term $\mathbf{E}_t$ is compressed and fed back.

utility of differential feedback. Instead of directly encoding/decoding the CSI (e.g., $g(f(\tilde{\mathbf{H}}_\tau))$), we propose to encode/decode the difference,

$$\begin{aligned}\bar{\mathbf{E}}_i &= \tilde{\mathbf{H}}_{\tau,i} - \hat{\tilde{\mathbf{H}}}_{\tau,i} \\ &= \tilde{\mathbf{H}}_{\tau,i} - \gamma\tilde{\mathbf{H}}_{\tau,i-1},\end{aligned} \tag{12}$$

where $\hat{\tilde{\mathbf{H}}}_{\tau,i} = \gamma\tilde{\mathbf{H}}_{\tau,i-1}$ is the least-squares estimate for $\tilde{\mathbf{H}}_{\tau,i}$ of the current $i-$th timeslot based on the estimate in the previous timeslot, $\hat{\tilde{\mathbf{H}}}_{\tau,i-1}$. Since the channel statistics are not known a-priori, we estimate the least-squares coefficient, $\hat{\gamma}$, as in [11],

$$\hat{\gamma} = \frac{\text{Trace}\left(E\{\mathbf{H}_t\mathbf{H}_{t-1}\}\right)}{E\|\mathbf{H}_{t-1}\|^2} \tag{13}$$

We apply the encoding/decoding process to the error term, $\hat{\mathbf{E}}_i = g_e(f_e(\bar{\mathbf{E}}_i))$, and the resulting CSI estimate can be written as

$$\hat{\tilde{\mathbf{H}}}_{\tau,i} = \hat{\mathbf{E}}_i + \hat{\gamma}\hat{\tilde{\mathbf{H}}}_{\tau,i-1}.$$

While the feedback is based on the error under the P2DE, the network at each timeslot is optimized using the mean-squared error loss function with respect to the error under the ground truth, $\mathbf{E}_i = \tilde{\mathbf{H}}_{\tau,i} - \hat{\gamma}\hat{\tilde{\mathbf{H}}}_{\tau,i-1}$,

$$L_{\text{MSE}} = \frac{1}{N_{\text{batch}}}\sum_{i=1}^{N_{\text{batch}}} \|\mathbf{E}_t^{(i)} - \hat{\mathbf{E}}_t^{(i)}\|_2^2 \tag{14}$$

where i indexes over the N_{batch} samples of a training batch.

Figure 6 demonstrates the principle of differential encoding used with P2D estimates. Notably, both the BS and the UE need access to a copy of the decoder, $g_t(x)$, in order to derive the error term $\mathbf{E}_t$ based on (12). Since both the encoder and the decoder are required on the UE side, we seek to design a differential encoding scheme with a small number of parameters.

A. CNN Autoencoders for CSI Feedback

Prior work utilized CNN autoencoders to implement a trainable differential encoding network for CSI estimation [11]. Using autoencoders in a differential encoding network, each timeslot t_i utilizes a CNN-based encoder ($f_i(x)$) and decoder ($g_i(x)$). Early work in deep learning-based CSI compression concluded that convolutional autoencoders consistently outperformed traditional compressed sensing (CS) approaches [5].

In this work, we investigate two autoencoder networks to realize our differential encoding network. First, we utilize CsiNet Pro [20], an improved version of CsiNet which utilizes a symmetric encoder/decoder structure without residual connections, and ENet [21], another symmetric architecture applied independently to the real and imaginary channels to produce a complex-valued matrix. These two networks can be viewed at the bottom of Figure 7.

B. Iterative Optimization Networks for Compressed Sensing-based CSI Feedback

While CNN autoencoders have been dominant in CSI estimation, recent work from image processing has shown promise in using trainable CS algorithms based on CNNs. These works treat iterative CS algorithms as sequential networks by "unrolling" them into discrete blocks [22], [23]. Investigating unrolled CS algorithms for CSI estimation warrants consideration, as CS algorithms can have guaranteed convergence under mild sparsity conditions (in contrast with CNNs autoencoder approaches, which do not have such guarantees). Since CSI data exhibits sparsity in the delay domain, specifying an appropriate compressed sensing approach could provide appreciable performance gains in our differential CSI encoding architecture.

To exploit the temporal coherence of the MIMO channel, we propose to construct a differential encoding network using an unrolled optimization network based on a trainable version of the iterative shrinkage-thresholding algorithm (ISTA), called ISTANet+ [23]. See the top of Figure 7 for a diagram of ISTANet+. Denote measurement matrix for the ISTANet+ as

$$\mathbf{\Phi} \in \mathbf{R}^{N_{\text{total}}\text{CR}\times N_{\text{total}}}. \tag{15}$$

For compressed sensing approaches, the measurement matrix is analogous to the 'encoder' for autoencoder approaches, i.e., $f(x) = \mathbf{\Phi}x$. The 'decoder' consists of K iterations of the following update steps,

$$\mathbf{r}^{(k)} = \mathbf{x}^{(k-1)} - \rho^{(k)}\mathbf{\Phi}^\top(\mathbf{\Phi}\mathbf{x}^{(k-1)} - \mathbf{y}) \tag{16}$$

$$\mathbf{x}^{(k)} = \mathbf{r}^{(k)} + \mathcal{G}^{(k)}\left(\tilde{\mathcal{H}}^{(k)}\left(\text{soft}\left(\mathcal{H}^{(k)}(\mathcal{D}^{(k)}(\mathbf{r}^{(k)}), \theta^{(k)})\right)\right)\right) \tag{17}$$

where $\mathbf{y} = \mathbf{\Phi}\mathbf{x}$, $\mathbf{x}^{(0)} = \mathbf{R}_{\text{init}}\mathbf{y}$, and $\mathbf{R}_{\text{init}} = \mathbf{X}\mathbf{Y}(\mathbf{Y}\mathbf{Y}^\top)^{-1}$ is the initialization matrix for the training data matrix $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_{N_{\text{train}}}]$ and the training measurement matrix $\mathbf{Y} = [\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_{N_{\text{train}}}]$. 'soft($\cdot$)' denotes the soft threshold function,

$$\text{soft}(x, \theta) = \text{sign}(x)\text{ReLU}(|x| - \theta). \tag{18}$$

$\mathcal{G}^{(k)}, \mathcal{D}^{(k)}, \mathcal{H}^{(k)}, \tilde{\mathcal{H}}^{(k)}$ indicate trainable nonlinear mappings (in this case, CNNs), and $\mathcal{H}^{(k)}, \tilde{\mathcal{H}}^{(k)}$ are subject to the symmetry constraint $\mathcal{H}^{(k)} \circ \tilde{\mathcal{H}}^{(k)} = \mathbf{I}$.

In the proposed differential encoding scheme, we use an instance of ISTANet+ in the first timeslot, t_1, with a large compression ratio such that $\text{CR}_{t_1} \geq \text{CR}_{t_i}$ for all $i > 1$. This choice in compression ratio allows us to initialize the network with a high-quality estimate at the first timeslot. Notably, the training data matrix, $\mathbf{X}$, differs between timeslots. For the first timeslot, the data vectors $\mathbf{x}_i$ are vectorized versions of the CSI matrices,

$$\mathbf{x}_j = \text{vec}\left(\mathbf{H}_{\tau,1}^{(j)}\right) \text{ for } j \in [N_{\text{train}}]. \tag{19}$$

However, the data vectors for all other timeslots are vectorized versions of the error matrices,

$$\mathbf{x}_j = \text{vec}\left(\bar{\mathbf{E}}_i^{(j)}\right) \text{ for } j \in [N_{\text{train}}]. \tag{20}$$

Denote the parameters for ISTANet+ in the t_i-th timeslot as $\mathbf{\Theta}_{t_i} = \{\mathcal{G}^{(k)}, \mathcal{D}^{(k)}, \mathcal{H}^{(k)}, \tilde{\mathcal{H}}^{(k)}\, \theta^{(k)}, \rho^{(k)}\}_{k=1}^{K}$. The loss

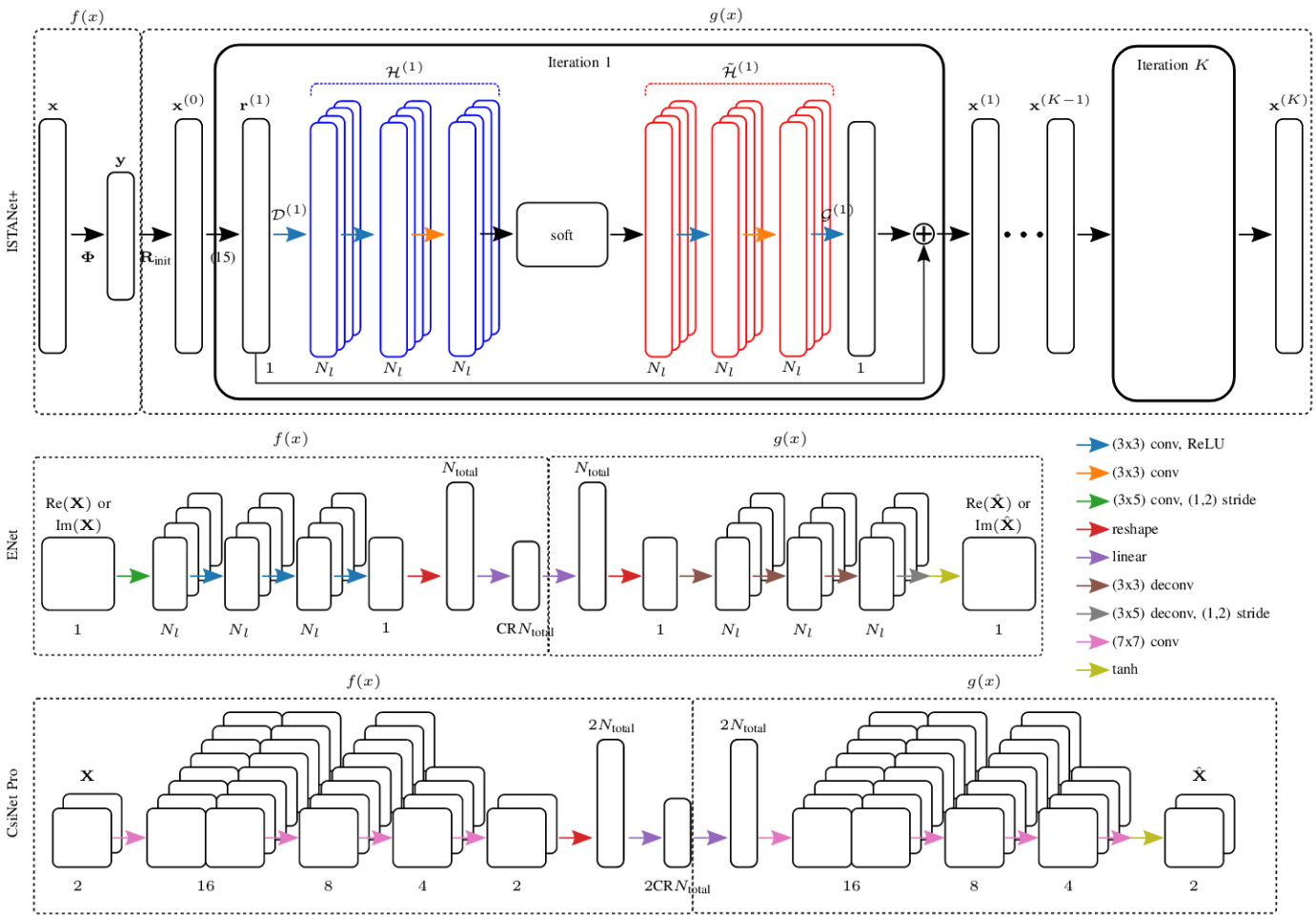

Fig. 7: Compressive CSI estimation architectures used in this work. $f(x)$ denotes the encoder, and $g(x)$ denotes the decoder. $N_{\text{total}} = N_b N_t$ is the size of the real or imaginary channel. N_l is the number of latent channels in a convolutional layer.

function is a weighted sum of the MSE and the symmetry constraint, i.e.,

$$L(\mathbf{\Theta}_{t_i}) = L_{\text{MSE}} + \alpha L_{\text{sym}} \tag{21}$$

$$L_{\text{MSE}} = \frac{1}{N_{\text{all}}} \sum_{i=1}^{N_{\text{batch}}} \|\mathbf{x}_i^{(K)} - \mathbf{x}_i\|_2^2 \tag{22}$$

$$L_{\text{sym}} = \frac{1}{N_{\text{all}}} \sum_{i=1}^{N_{\text{batch}}} \sum_{k=1}^{K} \|\tilde{\mathcal{H}}^{(k)}(\mathcal{H}^{(k)}(\mathbf{x}_i)) - \mathbf{x}_i\|_2^2 \tag{23}$$

where $N_{\text{all}} = N_{\text{batch}} N_b N_t$, N_{batch} is the batch size used during training, $N_b N_t$ is the size of the truncated CSI matrix, and K is the number of iterations in ISTANet+. As denoted in equations (19) and (20), the vectors $\mathbf{x}_i$ depend on the timeslot.

V. Random Phase Augmentation

Prior work leveraged the truncated delay domain, which allowed them to save large datasets of truncated CSI matrices. In order to acquire P2D estimates for different values of DR_f and D, we must store the full frequency domain CSI matrices. These full matrices can be prohibitively expensive to store under typical system parameters, meaning we need to use a smaller dataset. Since successful training of deep neural networks depends on a large number of training samples, we utilize a random phase augmentation on our smaller training data. Such random phase augmentation has been shown to be effective in training neural networks for CSI feedback compression [24]. For each sample in the training set, we sample a random phase from a uniform distribution, $\theta \sim \mathcal{U}(-\pi, \pi)$, and we rotate

TABLE I: Parameters for COST2100 model in this work.

Environment	Outdoor
Num. BS Antennas (N_b)	32
Truncation Value (N_t)	32
Num. Subcarriers (N_f)	1024
Downsampled Subcarriers (M_f)	$[512, 256, 128, 64]$
Carrier Frequency	300 MHz
UE Starting Position	400 m $\times$400 m
Num. Channel Samples (N)	2.5×10^4

all the elements in a given CSI matrix by this phase,

$$\mathbf{H}^{(i,j)}_{\text{augmented}} = \mathbf{H}^{(i,j)} e^{-j\theta} \; \forall \; i \in [N_t], j \in [M_f]. \quad (24)$$

We define a *phase augmentation factor*, N_{phase}, which is a multiplicative factor denoting the size of the training dataset after performing phase augmentation. For example, if we begin with a training set of size 5000, then $N_{\text{phase}} = 2$ would result in an augmented dataset of size $10,000$, meaning each sample in the training set is augmented once. More generally, each sample in the training set is augmented $N_{\text{phase}} - 1$ times.

VI. Numerical Results

We perform experiments using the COST2100 Model in an Outdoor scenario [17]. Table I summarizes the COST model parameters used to generate the Outdoor dataset. Importantly, the number of channel samples in the dataset is lower than the number used in similar works. A smaller dataset is necessary because we store full CSI matrices without truncating any subcarriers, which requires 32 times more space to store. For all networks, we utilize spherical normalization [20], and we test the networks using the following configurations:

- **ISTANet+**: We train the network described in Section IV-B for 100 epochs using the ADAM optimizer. The network utilizes $N_l = 32$ latent channels, $K = 9$ blocks, and a symmetry weight parameter of $\alpha = 10^{-2}$.
- **ENet**: The network hyperparameters are identical to those described by the authors of [21]. Since the training procedure was not described, we chose one which converged in a reasonable number of epochs (500 epochs, learning rate of $5 \cdot 10^{-4}$). As per the original paper, we train the network on the real channel data from the training set, then we report the validation loss by using the network on the real and imaginary channels from the validation set. We utilize $N_l = 32$ latent channels since this configuration achieved the best performance in the original paper.
- **CsiNet Pro**: The network was trained for 500 epochs with a learning rate of $5 \cdot 10^{-4}$.

We use a 80% (20%) training (validation) split, yielding 20,000 training samples (5000 validation samples). Unless stated otherwise, we augment the training set using $N_{\text{phase}} = 4$, yielding an augmented dataset of $80,000$ samples.

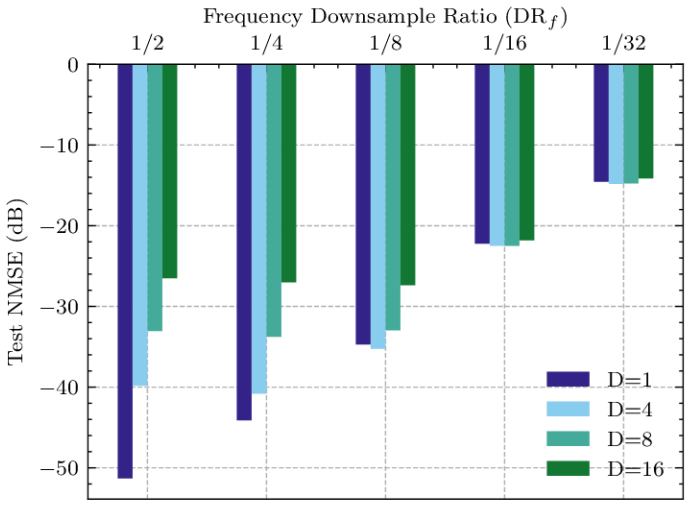

Fig. 8: P2D estimation performance under different frequency downsampling ratios (DR$_f$) and diagonal dimensions (D) for the Outdoor COST2100 dataset. Downsampling is done along the frequency axis.

A. Accuracy of P2D Estimator

To provide a bound on the estimation performance at the BS, Figure 8 shows the accuracy of the P2DE at the UE (i.e., before compression and feedback). This performance is based on perfect pilot estimates (i.e., no noise in $\mathbf{H}_d$). The performance of the P2DE under multiple diagonal sizes (D) is shown. For all P2DE tests, ODIR with $\delta = 0.5$ is used (see Section III-C for details). For all tested frequency downsampling ratios (DR$_f$), the accuracy of the P2DE is substantial, with the smallest DR$_f = \frac{1}{32}$ achieving about -14 dB. For increasing D, the error of the P2DE increases; however, the difference in performance for different values of D becomes negligible at more aggressive downsampling ratios, DR$_f \in \left[\frac{1}{16}, \frac{1}{32}\right]$. The accuracy

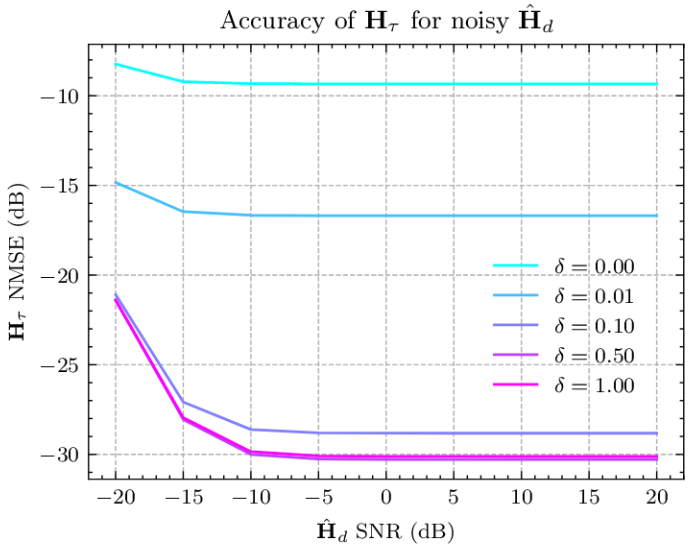

Fig. 9: Accuracy of P2DE output, $\mathbf{H}_\tau$, assuming noisy pilots, $\hat{\mathbf{H}}_d$. Additive Gaussian noise is used to model the error inherent in pilot estimation. Here, $D = 4, \text{DR}_f = \frac{1}{32}$.

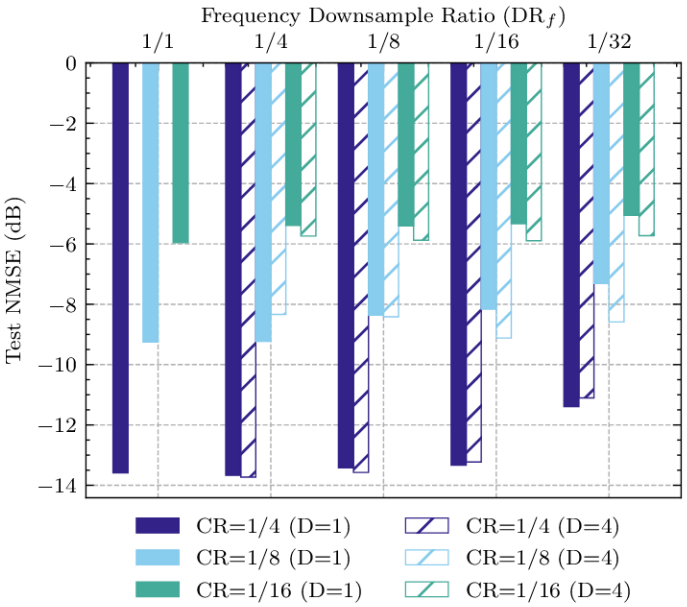

Fig. 10: Performance of ISTANet+ for multiple compression ratios using P2D estimates with different downsampling ratios ($\text{DR}_f = \frac{M_f}{N_f}$) for the Outdoor COST2100 dataset. Non-diagonal pattern ($D = 1$) is compared with a diagonal pattern of size $D = 4$. Performance for $\text{DR}_f = 1/1$, $D = 4$ is omitted since it is equivalent to the $\text{DR}_f = 1, D = 1$ case.

of the P2DE implies that it will perform well with compressive CSI feedback networks.

To understand the effect of pilot estimation error on the P2DE, Figure 9 shows the accuracy of the P2DE using a noise-corrupted $\hat{\mathbf{H}}_d$, defined below as:

$$\hat{\mathbf{H}}_d = \mathbf{H}_d + \mathbf{N}$$

where the elements of $\mathbf{N} \in \mathbb{C}^{N_b \times M_f}$ are i.i.d. zero-mean Gaussian (i.e., $N_{ij} \sim \mathcal{N}(0, \sigma^2)\ \forall\ i \in [1, \ldots, N_b], j \in [1, \ldots, M_f]$). By varying σ^2, we can control the realized SNR of $\hat{\mathbf{H}}_d$. We utilize $\hat{\mathbf{H}}_d$ as a surrogate for the estimated pilots, and we use P2DE to estimate $\mathbf{H}_\tau$ with $\hat{\mathbf{H}}_d$ as an input (see Algorithm 1, and substitute $\hat{\mathbf{H}}_d$ for $\mathbf{H}_d$).

To understand the effect of regularization on ill-conditioned matrices $\mathbf{Q}_{c,i}\mathbf{Q}_{c,i}^H$, we also vary the regularization parameter, δ. We observe that regularization of $\mathbf{Q}_{c,i}\mathbf{Q}_{c,i}^H$ is beneficial in both noisy and low noise conditions, as δ has an appreciable effect on the accuracy of the P2DE in either case. More specifically,

- **Noisy conditions (SNR $= -20$ dB)**: Compare $\delta = 0$ to $\delta = 0.5$, where the NMSE of $\mathbf{H}_\tau$ is -8 and -21 dB, respectively.
- **Low noise condition (SNR ≥ -10 dB)**: Compare $\delta = 0$ to $\delta = 0.5$, where the NMSE of $\mathbf{H}_\tau$ is -9 and -30 dB, respectively.

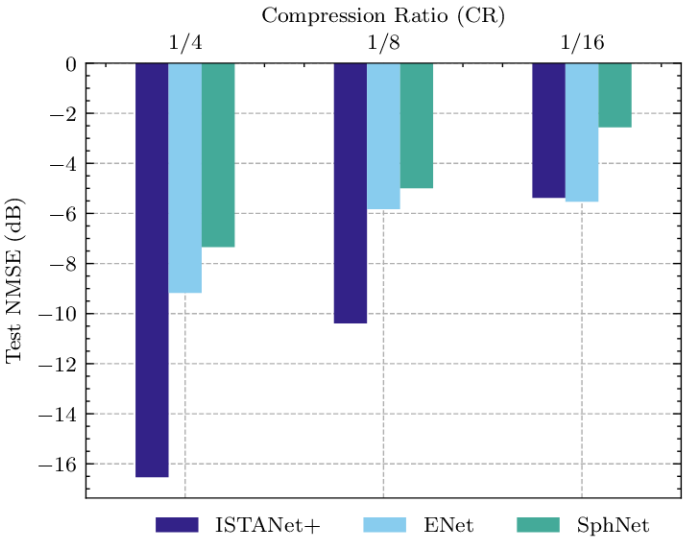

Fig. 11: Performance comparison for different feedback compression networks using P2D estimates ($\text{DF}_f = 1/16, D = 4$) for Outdoor COST2100 dataset. For all tested networks, we use $N_{\text{phase}} = 4$, resulting in an augmented training set with 80k samples.

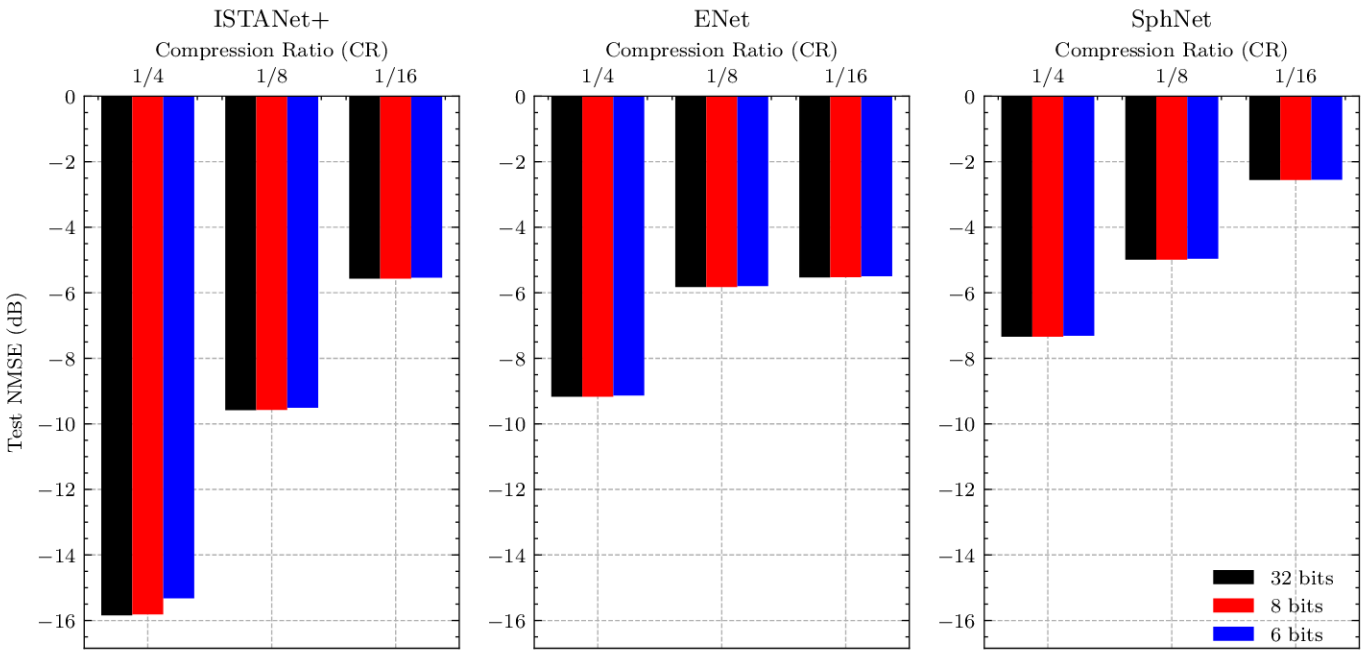

Fig. 12: Tested networks where feedback is subject to μ-law companding ($\mu = 255$) and uniform quantization for different numbers of quantization bits. P2DE parameters are $D = 4, \text{DR}_f = \frac{1}{16}$.

B. Accuracy of Compressive Networks with P2D Estimates

In these experiments, we use the P2D estimate as the input to different compressive CSI feedback networks. In this work, we propose to use the unrolled reconstruction network, ISTANet+ [23], as described in Section IV-B. In Figure 10, we assess the performance of ISTANet+ across multiple values of DR_f and CR. Comparing $\text{DR}_f = \frac{1}{1}$ to $\text{DR}_f = \frac{1}{16}$, the accuracy of ISTANet+ is remarkably stable, increasing negligibly for CR=$\frac{1}{4}$ and by only 1 dB for CR=$\frac{1}{16}$.

To provide a baseline for ISTANet+, we also compare the performance of ISTANet+ with two autoencoder-based CSI compression networks, CsiNet Pro [20] and ENet [21]. Figure 11 shows the performance comparison between all networks for the same DR_f and D. For large compression ratios (CR $\in [\frac{1}{4}, \frac{1}{8}]$), ISTANet+ achieves a better NMSE than the autoencoder approaches. For the smallest tested compression ratio (CR $= \frac{1}{16}$), ISTANet+ has similar performance to ENet.

To consider the effect of noise and decoding error during CSI feedback and estimation, we choose to quantize the digital feedback of each CSI estimation network using uniform quantization with μ-law companding (see [11] for full details). The resulting performance of each network for different levels of quantization is shown in Fig. 12. Note that 32 bit quantization corresponds to floating point representation which is nearly ideal. We observe that ISTANet+ exhibits superior performance over the autoencoder networks even under aggressive feedback quantization (i.e., 6 or 8 quantization bits).

C. Phase Augmentation Sensitivity Study

Using random phase augmentation as described in Section V, we assess the influence of different sized training sets on validation accuracy. We start with a base training set of size 1000, 5000, or 20000, and we utilize multiple values of N_{phase}, to yield augmented training sets of increasing size. We train ISTANet+ (CR=$\frac{1}{4}$) with P2D estimates ($\text{DR}_f = \frac{1}{8}$) on each of these training sets, and we report the validation loss on the same set of 5000 samples. The resulting validation accuracy can be seen in Figure 13. As expected, the accuracy improves appreciably as the size of the augmented training set is increased, and the difference between the networks trained with 5000 and 20000 training samples becomes negligible after augmentation.

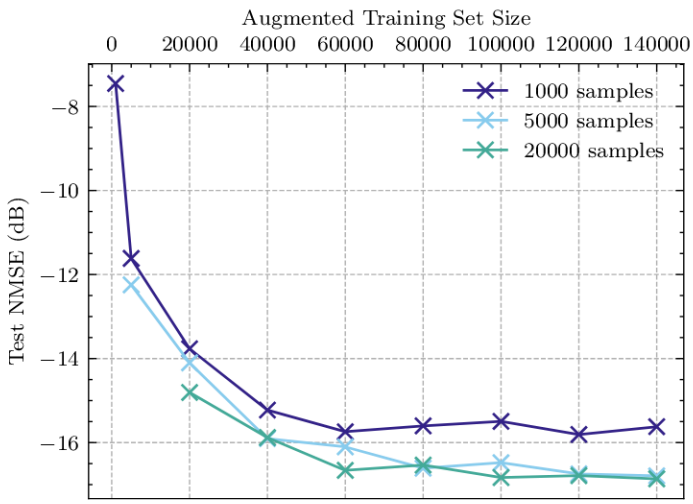

Fig. 13: Effect of phase randomization augmentation on performance of ISTANet+ (CR $= \frac{1}{4}$) with P2D estimates ($\text{DR}_f = \frac{1}{16}, D = 4$) under different training set sizes for the Outdoor COST2100 dataset. Downsampling is done along the frequency axis.

D. Differential Encoding with P2D Estimates

Figure 14 shows the performance of differential encoding when using either ISTANet+ and ENet at each timeslot, which are respectively named MarkovNet-ISTA (MN-I) and MarkovNet-ENet (MN-E). For all versions of MarkovNet, CR_{t_1} is the compression ratio in the first timeslot and CR is the compression ratio for all following timeslots. ISTANet+ has the benefit of providing accuracy in the first timeslot, while ENet is better at compressing the residual in each following timeslot. Based on this observation, we also test a version of MarkovNet which uses ISTANet+ in the first timeslot then ENet in the following timeslots, which we call MarkovNet-ISTA-ENet (MN-IE). For the networks where $\text{CR}_{t_1} = \text{CR}$, MN-IE can outperform MN-I, indicating that a combination of architectures can be better than a single architecture. Such an outcome appears reasonable given the difference in sparsity between the first timeslot (where CSI data is compressed) and the ensuing timeslots (where only error terms are compressed). Since angular-delay domain CSI is sparse, ISTANet+ (a network that emulates compressed sensing algorithms) is suitable for compressing this sparse CSI data. In contrast, error terms are not necessarily sparse, making the use of an autoencoder (in this case, ENet) architecture more suitable.

TABLE II: Computational complexity of networks used in this work. **Bold face** in a column indicates lowest value for given compression ratio. "CR" = compression ratio, "Enc" = encoder, "Dec" = decoder. FLOPs indicate computation during inference (i.e., not training/back-propagation).

		Parameters (M)				FLOPs (M)	
		Trainable		All			
	CR	Enc	Dec	Enc	Dec	Enc	Dec
ISTANet	1/2	**0.00**	**0.34**	2.10	4.54	**2.10**	393.78
	1/4	**0.00**	0.34	1.05	2.44	**1.05**	373.85
	1/8	**0.00**	0.34	0.52	1.39	**0.52**	363.89
	1/16	**0.00**	0.34	0.26	0.87	**0.26**	358.91
ENet	1/2	0.55	0.55	**0.55**	**0.55**	29.98	29.70
	1/4	0.29	**0.29**	**0.29**	**0.29**	29.46	29.18
	1/8	0.16	**0.16**	**0.16**	**0.16**	29.20	28.92
	1/16	0.09	**0.09**	**0.09**	**0.09**	29.07	28.79
CsiNet Pro	1/2	1.06	1.06	1.06	1.06	12.16	**12.16**
	1/4	0.53	0.53	0.53	0.53	11.11	**11.11**
	1/8	0.27	0.27	0.27	0.27	10.59	**10.59**
	1/16	0.14	0.14	0.14	0.14	10.33	**10.33**

E. Computational Complexity

Table II shows the computational complexity of the different trainable networks used in this work. Generally, ISTANet+ uses fewer trainable parameters than the autoencoder networks, but the autoencoder networks use fewer total parameters (i.e., trainable and nontrainable parameters) than ISTANet+. With respect to floating point operations (FLOPs), ISTANet+ uses an order of magnitude more FLOPs than the autoencoder approaches. This large computational cost is primarily due to the unrolled iterations of the compressed sensing algorithm (i.e., ISTA).

The large computational complexity of ISTANet+ motivates the used of autoencoders in conjunction with compressed sensing networks as done in MN-IE. In differential encoding networks, both the encoder and the decoder need to be stored and executed at the UE. For MN-I, T copies of the encoder/decoder must be kept at the UE, which would consume an unreasonable amount of memory and compute. For example, with $\text{CR}_{t_1} = \frac{1}{4}$, CR $= \frac{1}{4}$, MN-I would consume T(6.54) million parameters and T(395.88) million FLOPs. In contrast, MN-IE would consume $6.54 + (T-1)(1.1)$ million parameters and $395.88 + (T-1)(29.84)$ million FLOPs.

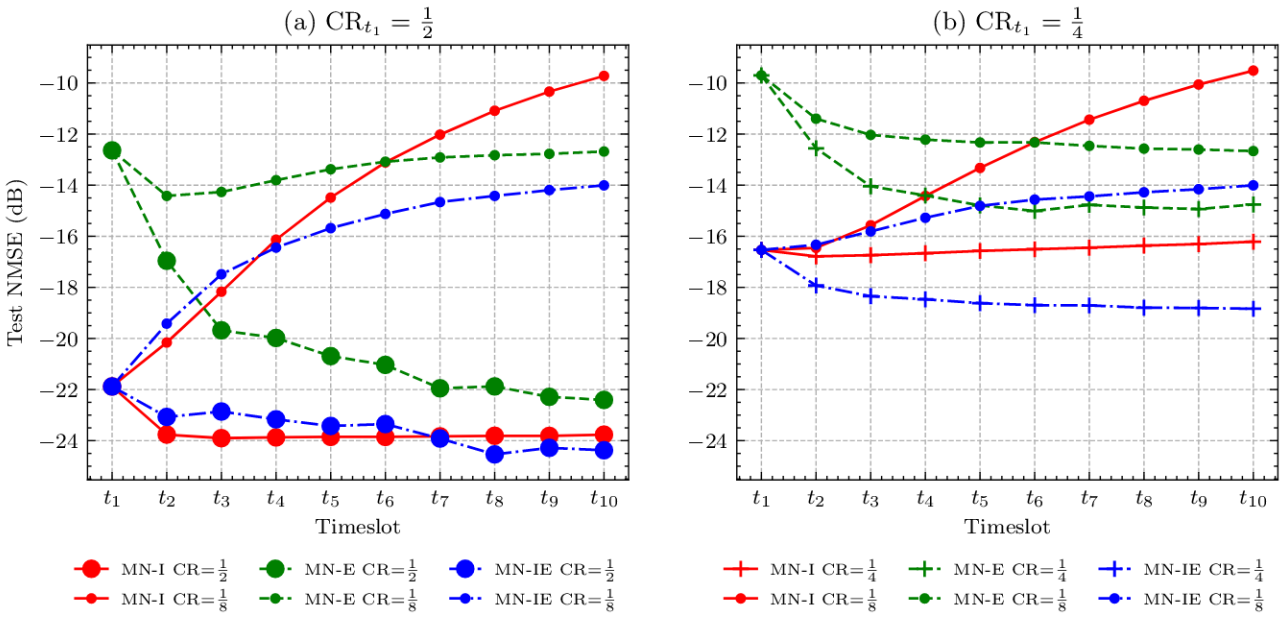

Fig. 14: Compressive CSI estimation using differential encoding and linear P2D estimator ($M_f = 128, \mathrm{DR}_f = \frac{1}{8}, D = 4$). MarkovNet-ISTA (MN-I), MarkovNet-ENet (MN-E), and MarkovNet-ISTA-ENet (MN-IE) are tested using two different compression ratios in the first timeslot, $\mathrm{CR}_{t_1} \in \left[\frac{1}{2}, \frac{1}{4}\right]$.

VII. Discussion

This work presents a novel P2D estimator for of downlink delay domain MIMO CSI based on sparsely populated frequency domain pilot signals. Taking advantage of the channel coherence exhibited in the final delay spread of CSI, the UE can leverage the P2D estimator to accurately estimate delay domain CSI based on practical CSI-RS (DMRS) pilot allocations that are consistent with the 4G/LTE (5G/NR) standards. Furthermore, we demonstrate that CSI estimates from the P2D estimator provide a suitable input to trainable compressive sensing networks and autoencoder networks. Lastly, to improve feedback efficiency for downlink CSI recovery by BS over multiple time slots, we integrate the concept of a differential encoding to develop a heterogeneous deep learning network, MarkovNet-ISTA-ENet. This new architecture combines a trainable CS network with a bank of autoencoders designed for successive time slots to take advantage of temporal coherence of wireless CSI by leveraging the high initial CSI recovery accuracy of the CS network on the first time slot to improve CSI recovery of subsequent time slots. In future works, it is also of interest to consider imperfect CSI estimates at pilots as well as adaptive methods for initial pilot estimation (see [14], [15], for example).

Acknowledgements

The authors would like to acknowledge Zhenyu Liu for his important contributions on data augmentation and Yu-Chien Lin for his useful discussions which helped the authors better understand the layout of CSI-RS/DMRS in 4G/5G specifications.

References

[1] A. Goldsmith, S. A. Jafar, N. Jindal, and S. Vishwanath, "Capacity Limits of MIMO Channels," *IEEE Journal on Selected Areas in Communications*, vol. 21, no. 5, pp. 684–702, June 2003.

[2] F. Kaltenberger, H. Jiang, M. Guillaud, and R. Knopp, "Relative Channel Reciprocity Calibration in MIMO/TDD Systems," in *2010 Future Network Mobile Summit*, June 2010, pp. 1–10.

[3] D. Mi, M. Dianati, L. Zhang, S. Muhaidat, and R. Tafazolli, "Massive MIMO Performance with Imperfect Channel Reciprocity and Channel Estimation Error," *IEEE Trans. Communications*, vol. 65, no. 9, pp. 3734–3749, 2017.

[4] Q. Gao, F. Qin, and S. Sun, "Utilization of Channel Reciprocity in Advanced MIMO System," in *2010 5th International ICST Conference on Communications and Networking in China*, Aug 2010, pp. 1–5.

[5] C. Wen, W. Shih, and S. Jin, "Deep Learning for Massive MIMO CSI Feedback," *IEEE Wireless Communications Letters*, vol. 7, no. 5, pp. 748–751, Oct 2018.

[6] Z. Lu, J. Wang, and J. Song, "Multi-resolution CSI Feedback with Deep Learning in Massive MIMO System," in *ICC 2020 - 2020 IEEE International Conference on Communications (ICC)*, 2020, pp. 1–6.

[7] M. Hussien, K. K. Nguyen, and M. Cheriet, "PRVNet: Variational Autoencoders for Massive MIMO CSI Feedback," *arXiv*, 2020.

[8] Y. Sun, W. Xu, L. Fan, G. Y. Li, and G. K. Karagiannidis, "AnciNet: An Efficient Deep Learning Approach for Feedback Compression of Estimated CSI in Massive MIMO Systems," *IEEE Wireless Communications Letters*, vol. 9, no. 12, pp. 2192–2196, 2020.

[9] Z. Liu, L. Zhang, and Z. Ding, "Exploiting Bi-Directional Channel Reciprocity in Deep Learning for Low Rate Massive MIMO CSI Feedback," *IEEE Wireless Comm. Letters*, vol. 8(3), pp. 889–892, 2019.

[10] T. Wang, C. Wen, S. Jin, and G. Y. Li, "Deep Learning-Based CSI Feedback Approach for Time-Varying Massive MIMO Channels," *IEEE Wireless Communications Letters*, vol. 8, no. 2, pp. 416–419, April 2019.

[11] Z. Liu†, M. del Rosario†, and Z. Ding, "A Markovian Model-Driven Deep Learning Framework for Massive MIMO CSI Feedback," *IEEE Transactions on Wireless Communications*, vol. 21, no. 2, pp. 1214–1228, 2022, †equal contribution.

[12] 3GPP, "NR; Physical Channels and Modulation," 3rd Generation Partnership Project (3GPP), TS 38.211, Jan. 2022. [Online]. Available: http://www.3gpp.org/dynareport/38211.htm

[13] P. Dong, H. Zhang, G. Y. Li, I. S. Gaspar, and N. Naderi-Alizadeh, "Deep cnn-based channel estimation for mmwave massive mimo systems," *IEEE Journal of Selected Topics in Signal Processing*, vol. 13, no. 5, pp. 989–1000, 2019.

[14] J. Guo, C.-K. Wen, and S. Jin, "Canet: Uplink-aided downlink channel acquisition in fdd massive mimo using deep learning," *IEEE Transactions on Communications*, vol. 70, no. 1, pp. 199–214, 2021.

[15] M. B. Mashhadi and D. Gündüz, "Pruning the pilots: Deep learning-based pilot design and channel estimation for mimo-ofdm systems," *IEEE Transactions on Wireless Communications*, vol. 20, no. 10, pp. 6315–6328, 2021.

[16] J. Guo, L. Wang, F. Li, and J. Xue, "CSI Feedback with Model-Driven Deep Learning of Massive MIMO Systems," *IEEE Communications Letters*, 2021.

[17] L. Liu, C. Oestges, J. Poutanen, K. Haneda, P. Vainikainen, F. Quitin, F. Tufvesson, and P. D. Doncker, "The COST 2100 MIMO Channel Model," *IEEE Wireless Comm.*, vol. 19, no. 6, pp. 92–99, Dec. 2012.

[18] 3GPP, "Evolved Universal Terrestrial Radio Access (E-UTRA); Physical channels and modulation," 3rd Generation Partnership Project (3GPP), TS 36.211, Jan. 2022. [Online]. Available: http://www.3gpp.org/dynareport/36211.htm

[19] H. Asplund, D. Astely, P. von Butovitsch, T. Chapman, M. Frenne, F. Ghasemzadeh, M. Hagström, B. Hogan, G. Jöngren, J. Karlsson, F. Kronestedt, and E. Larsson, "Chapter 8 - 3GPP Physical Layer Solutions for LTE and the Evolution Toward NR," in *Advanced Antenna Systems for 5G Network Deployments*. Academic Press, 2020, pp. 301–350. [Online]. Available: https://www.sciencedirect.com/science/article/pii/B9780128200469000083

[20] Z. Liu, M. del Rosario, X. Liang, L. Zhang, and Z. Ding, "Spherical Normalization for Learned Compressive Feedback in Massive MIMO CSI Acquisition," in *IEEE ICC Workshops*, 2020, pp. 1–6.

[21] Y. Sun, W. Xu, L. Liang, N. Wang, G. Y. Li, and X. You, "A Lightweight Deep Network for Efficient CSI Feedback in Massive MIMO Systems," *IEEE Wireless Communications Letters*, 2021.

[22] Y. Yang, J. Sun, H. Li, and Z. Xu, "Deep ADMM-Net for Compressive Sensing MRI," in *Proceedings of the 30th international conference on neural information processing systems*, 2016, pp. 10–18.

[23] J. Zhang and B. Ghanem, "ISTA-Net: Interpretable Optimization-inspired Deep Network for Image Compressive Sensing," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 1828–1837.

[24] Z. Liu and Z. Ding, "Enhancing the Performance of Deep Learning-Based Massive MIMO CSI Feedback For Small Dataset," 2022, Under preparation.

Mason del Rosario (S'19) is a PhD candidate in the Electrical and Computer Engineering Graduate Program at the University of California, Davis. His research involves efficient deep learning for channel state estimation and feedback in massive MIMO networks. Currently, he is interested in methods for reducing the computational complexity of deep learning networks for CSI estimation.

Before pursuing his PhD, Mason was a Powertrain Test Engineer at Tesla Motors where he conducted environmental durability testing on power electronics including the high voltage battery and drive inverters. Currently, Mason was a Graduate Writing Fellow with UC Davis' University Writing Program and a Professors for the Future Fellow with UC Davis' GradPathways Institute for Professional Development, and he is a licensed Professional Engineer in the state of California.

Zhi Ding (S'88-M'90-SM'95-F'03) is with the Department of Electrical and Computer Engineering at the University of California, Davis, where he holds the position of distinguished professor. He received his Ph.D. degree in Electrical Engineering from Cornell University in 1990. From 1990 to 2000, he was a faculty member of Auburn University and later, University of Iowa. Prof. Ding has held visiting academic positions in Australian National University, Hong Kong University of Science and Technology, NASA Lewis Research, and Southeast University. Prof. Ding has active collaboration with researchers from various universities in Australia, Canada, China, Finland, Hong Kong, Japan, Korea, Singapore, Taiwan, and USA.

Prof. Ding is a Fellow of IEEE and currently serves as the Chief Information Officer and Chief Marketing Officer of the IEEE Communications Society. He was associate editor for IEEE Transactions on Signal Processing from 1994-1997, 2001-2004, and associate editor of IEEE Signal Processing Letters 2002-2005. He was a member of technical committee on Statistical Signal and Array Processing and member of technical committee on Signal Processing for Communications (1994-2003). Dr. Ding was the General Chair of the 2016 IEEE International Conference on Acoustics, Speech, and Signal Processing and the Technical Program Chair of the 2006 IEEE Globecom. He was also an IEEE Distinguished Lecturer (Circuits and Systems Society, 2004-06, Communications Society, 2008-09). He served on as IEEE Transactions on Wireless Communications Steering Committee Member (2007-2009) and its Chair (2009-2010). Dr. Ding is a coauthor of the textbook: Modern Digital and Analog Communication Systems, 5th edition, Oxford University Press, 2019. Prof. Ding received the IEEE Communication Society's WTC Award in 2012 and the IEEE Communication Society's Education Award in 2020.