# A Markovian Model-Driven Deep Learning Framework for Massive MIMO CSI Feedback

Zhenyu Liu†, Mason del Rosario†, *Graduate Student Member, IEEE*, and Zhi Ding, *Fellow, IEEE*

***Abstract*—Channel state information (CSI) plays a vital role in scheduling and capacity-approaching transmission optimization of massive MIMO communication systems. In frequency division duplex (FDD) MIMO systems, forward link CSI reconstruction at transmitter relies on CSI feedback from receiving nodes and must carefully weigh the tradeoff between reconstruction accuracy and feedback bandwidth. Recent application of recurrent neural networks (RNN) has demonstrated promising results of massive MIMO CSI feedback compression. However, the cost of computation and memory associated with RNN deep learning remains high. In this work, we exploit channel temporal coherence to improve learning accuracy and feedback efficiency. Leveraging a Markovian model, we develop a deep convolutional neural network (CNN)-based framework called MarkovNet to efficiently encode CSI feedback to improve accuracy and efficiency. We explore important physical insights including spherical normalization of input data and deep learning network optimizations in feedback compression. We demonstrate that MarkovNet provides a substantial performance improvement and computational complexity reduction over the RNN-based work. We demonstrate MarkovNet's performance under different MIMO configurations and for a range of feedback intervals and rates. CSI recovery with MarkovNet outperforms RNN-based CSI estimation with only a fraction of computational cost.**

## I. Introduction

Massive MIMO wireless interface has been identified as a critical radio technology at the physical layer capable of substantially improving the bandwidth efficiency and delivering Gigabits/s services to many heterogeneous subscribers simultaneously. The efficacy of such massive MIMO downlink depends on the availability of accurate forward (down) link CSI estimates at base station (BS) for transmission precoding. Given the large number of antennas in massive MIMO and potentially broad bandwidth, such downlink CSI estimation and acquisition require a substantial amount of feedback from each subscriber user equipment (UE). To support high mobility UEs in modern mobile wireless, timely feedback for time varying (i.e., fading) CSI estimates [1], [2] pose critical challenges. Unnecessarily frequent reporting of CSI for massive MIMO coverage would consume too much network bandwidth and UE power. The need for efficient CSI feedback in massive MIMO networks strongly motivates many research efforts aimed at downlink CSI compression, feedback, and reconstruction.

The problem of CSI feedback and reconstruction in massive MIMO has been an active research area recently. Traditional vector quantization and codebook-based methods reduce feedback overhead by quantizing the CSI at the UE side [3]–[6]. However, the feedback overhead grows with the number of antennas, often requiring large amount of uplink bandwidth or low accuracy for practical massive MIMO wireless transmission. Compressive sensing (CS)-based approaches exploit the sparsity channel property in some domain to lower the CSI feedback overhead [7]–[9]. However, CS-based approaches often hinge on strong channel sparsity conditions not strictly satisfied in some domains. Moreover, iterative CS reconstruction methods may need a large amount of computation time to accurately recover downlink CSI estimates.

There has been a surging wave of interest in applying artificial neural networks for forward CSI estimation [10]–[13]. The popularity and versatility of deep learning (DL) have motivated a number of recent works [14]–[23] that explored deep neural networks (DNN) for downlink channel feedback compression and recovery, particularly for massive MIMO wireless interface when traditional feedback consume s substantial bandwidth. For example, CsiNet [14] is a convolutional autoencoder which is trained to compress and reconstruct downlink CSI under limited bandwidth. Subsequently, variational autoencoders [18], multi-resolution convolutional neural networks (CNNs) [16] and denoising modules [17] have shown enhancement of CSI feedback performance. Besides modifying the structure of CSI feedback network, other works have demonstrated the benefit of leveraging additional side information in DL such as bi-directional CSI correlation [15] and temporal CSI correlation [19] to achieve more accurate CSI feedback. Specifically, Recurrent Neural Networks (RNNs) have exploited temporal CSI coherent for feedback compression in massive MIMO systems [19]–[22]. RNNs use hidden states in architectures such as long short-term memory (LSTM) cells to exploit information of past inputs.

Existing works have demonstrated that RNNs can provide efficient CSI feedback and reconstruction for time-varying MIMO channels [19]–[22]. However, some open questions remain:

This material is based upon work supported by the National Science Foundation under Grants ECCS-1711823, ECCS-2029027, and CNS-2002937.

Z. Liu is with the School of Computer Science (National Pilot Software Engineering School), Beijing University of Posts and Telecommunications, China (e-mail: lzyu@bupt.edu.cn).

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

†Z. Liu and M. del Rosario contributed equally to this manuscript.

1) **Complexity and Storage**: The number of parameters in RNN layers for CSI compression and reconstruction of massive MIMO systems can be staggeringly large. For example, RNN modules may add $10^8$ additional parameters [19], which strains storage and computation resources. Although a fully connected layer-based autoencoder which can reduce computational complexity and memory needs has been proposed for the CSI feedback in time varying channels [23], the achieved CSI recovery accuracy is less impressive in comparison to [19]. Among other works attempting to reduce RNN size [22], [23], the most competitive models still require about $10^7$ parameters per snapshot. Considering the large RNN parameter count, it is hard to justify the huge memory overhead for the limited performance improvement thus far. Furthermore, the works of [22], [23] reported a significant drop of CSI recovery performance for large compression ratios; this performance drop was likely due to the same compression ratio being used in successive time slots, making it difficult to obtain accurate CSI estimates at the initial time slot.
2) **Physical Insight**: The success of RNNs in areas such as video processing [24] and natural language processing (NLP) [25], [26] has stimulated their applications in forward CSI feedback and reconstruction. However, despite the apparent similarities among time series, the physical nature of underlying CSI in massive MIMO is considerably different from those in video and image data. Leveraging domain knowledge and physical characteristics on mobile wireless channels can be beneficial. For example, each LTE subframe spans 1ms of airtime and permits CSI feedback intervals that are integer multiples of subframe duration. DNN-based CSI feedback and recovery should consider the practical constraint of how often such feedback can be transmitted and how CSI of fading channels would vary due to the Doppler effect.

In order to improve CSI recovery accuracy and reduce feedback payload, we develop a novel deep learning framework based on Markovian learning model, MarkovNet. MarkovNet systematically exploits temporal channel correlation characteristics to achieve a much smaller model size, thereby substantially reducing computational complexity and memory requirements.

Our contributions in this paper are summarized as follows:

- Instead of training an RNN as a black box to acquire the underlying forward link CSI's fading characteristics, we develop a simple but effective Markovian model that strongly motivates the differential CSI feedback framework. We provide an information theoretic rationale for MarkovNet based on the correlation between MIMO CSIs at successive timeslots. We demonstrate the efficacy of a simple low order auto-regressive model as one of simplest form of Markovian models.
- To achieve high-accuracy initial CSI feedback as the prior information for subsequent CSI estimates, we develop a spherical CSI feedback framework [27] which can regulate the input distribution and make the network's objective function (i.e., MSE) more suitable to the commonly adopted accuracy metric for CSI estimation (i.e., NMSE).
- To further mitigate the deployment cost of DL-based CSI feedback solutions, we tackle the high parameter count of fully connected layers widely used in existing CSI dimension compression and decompression modules, and we propose a lightweight CNN-based DL module with substantial complexity reduction without noticeable performance loss.
- Applying a benchmark from an established RNN for CSI estimation, we demonstrate that MarkovNet achieves better recovery accuracy, lower computational complexity, and less sensitivity to feedback quantization. We also demonstrate the efficacy of MarkovNet for a variety of channel configurations (i.e., indoor vs. outdoor, antenna counts, feedback intervals).
- We uncovered a problem in previous works which only measured CSI recovery error with respect to the truncated CSI matrix in the delay domain. Such CSI recovery error neglected the CSI recovery error due to the truncation and artificially under-reported the estimation error at the decoder. We assess the recovery error with respect to the entire MIMO CSI matrix to accurately report the full CSI estimation accuracy.

This paper is organized as follows. Section II describes the massive MIMO system model commonly adopted in this and similar works. Section III presents two approaches to exploit CSI temporal coherence (RNNs and differential encoding) and introduces our conditional entropy-based motivation for differential encoding. Section IV describes our proposed differential encoding-based CSI feedback framework, MarkovNet, as well as data pre-processing techniques to further improve CSI recovery accuracy for individual channel snapshots such as power-based spherical normalization. Section V introduces the proposed CNN-based dimension compression and decompression module for model size and complexity reduction. Section VI presents our experimental results, including analysis of computational complexity and performance under feedback quantization, for MarkovNet in comparison with a benchmark RNN-based network. Section VII concludes this manuscript and outlines directions for future work.

## II. System Model

### A. Forward Link Channnel Estimation and Reconstruction

In this paper, we consider a massive MIMO BS known in 5G as gNB equipped with $N_b \gg 1$ antennas to serve a number of single-antenna UEs within its cell. We apply orthogonal frequency division multiplexing (OFDM) in downlink transmission over $N_f$ subcarriers.

To model the received signal of a UE, consider the $m-$th subcarrier at time $t$. Let $\mathbf{h}_{t,m} \in \mathbb{C}^{N_b \times 1}$ denote the channel vector, $\mathbf{w}_{t,m} \in \mathbb{C}^{N_b \times 1}$ denote transmit precoding vector, $x_{t,m} \in \mathbb{C}$ be the transmitted data symbol, and $n_{t,m} \in \mathbb{C}$ be the additive noise. Then the received signal of the UE on the $m-$th subcarrier at time $t$ is given by

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

where $(\cdot)^H$ represents the conjugate transpose. The downlink CSI matrix in the spatial frequency domain at time $t$ is denoted as $\tilde{\mathbf{H}}_t = \left[\mathbf{h}_{t,1}, ..., \mathbf{h}_{t,N_f}\right]^H \in \mathbb{C}^{N_f \times N_b}$. Based on the downlink CSI matrix $\tilde{\mathbf{H}}_t$, the gNB can apply transmit precoding for each subcarrier. However, since the CSI matrix size is $N_f \times N_b$, the CSI feedback payload by UE is large and consumes a staggering amount of uplink bandwidth for massive MIMO systems.

To reduce feedback overhead, we first exploit CSI sparsity in the time domain delay space. Multipath effects cause short delay spreads, resulting in sparse CSI matrices in the delay domain [28]. With the help of 2D discrete Fourier transform (DFT), CSI matrix $\mathbf{H}_f$ in spatial-frequency domain can be transformed to be $\mathbf{H}_d$ in angular-delay domain using

$$\mathbf{F}_d^H \mathbf{H}_f \mathbf{F}_a = \mathbf{H}_d, \tag{2}$$

where $\mathbf{F}_d$ and $\mathbf{F}_a$ denote the $N_f \times N_f$ and $N_b \times N_b$ unitary DFT matrices, respectively. After 2D DFT of $\mathbf{H}_f$, most elements in the $N_f \times N_b$ matrix $\mathbf{H}_d$ are negligible except for the first $R_d$ rows that dominate the channel response [14]. Therefore, we can approximate the channel by truncating CSI matrix to the first $R_d$ rows. $\mathbf{H}_t$ is utilized to denote the first $R_d$ rows of matrices after 2D DFT of $\tilde{\mathbf{H}}_t$. Using $\mathbf{H}_t$ as a supervised learning objective, a DL based encoder and decoder, which is often referred to as an autoencoder, can be jointly trained and optimized to achieve efficient CSI compression and reconstruction as shown in Fig. 1. Several recent works that adopted this autoencoder structure [14] [15] have reported notable successes.

To allow gNB to track the time-varying characteristics of wireless fading channels, UEs need to periodically estimate and feed back instantaneous CSI with high power and bandwidth efficiency. Considering a time duration with $T$ successive time slots, the sequence of time-varying channel matrix is defined as $\{\mathbf{H}_t\}_{t=1}^T = \{\mathbf{H}_1, \mathbf{H}_2, \cdots, \mathbf{H}_T\}$.

### *B. High Efficiency CSI Feedback Encoding*

To reduce feedback overhead, temporal coherence of the radio fading channels can be exploited. Since RF channels of mobile UEs are governed by physical scatterers, multipaths, bandwidth, and Doppler effect, the fading CSI exhibits physically coherent characteristics including coherence time, coherence bandwidth, and coherence space. For mobile users, coherence time measures temporal channel variations and describes the Doppler effect caused by UE mobility. For most application scenarios, the massive MIMO channels do not vary abruptly. By exploiting the channel coherence time, the UE and the gNB can rely on their previously stored CSI estimates to encode only the innovation components within the CSI. Specifically, the UE can encode and feed back CSI variations instead of the full CSI to substantially reduce feedback cost. Accordingly, gNB can combine the new feedback with its previously recovered CSI within coherence time to reconstruct subsequent CSI estimates.

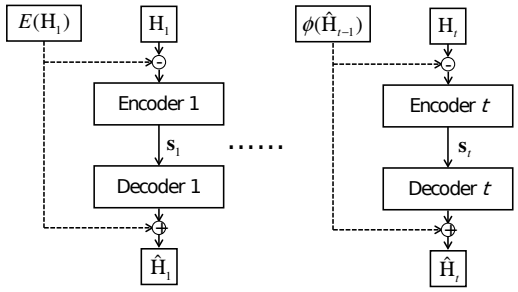



Fig. 1: Illustration of the temporal correlation based CSI feedback. $(t > 1)$

We can adopt a general first order Markovian channel model

$$p(\mathbf{H}_t | \mathbf{H}_{t-1}, \cdots, \mathbf{H}_1) = p(\mathbf{H}_t | \mathbf{H}_{t-1}). \tag{3}$$

Given knowledge of the CSI at the previous time slot, the minimum mean square error (MMSE) estimator of $\mathbf{H}_t$ can be defined as

$$\phi(\mathbf{H}_{t-1}) = E\{\mathbf{H}_t | \mathbf{H}_{t-1}\}. \tag{4}$$

We define the MMSE estimation error as

$$\mathbf{V}_t = \mathbf{H}_t - E\{\mathbf{H}_t | \mathbf{H}_{t-1}\} = \mathbf{H}_t - \phi(\mathbf{H}_{t-1}). \tag{5}$$

Consider the scenario that, at time $t-1$, the UE and the gNB have successfully exchanged the CSI $\mathbf{H}_{t-1}$. Then it would be more efficient for the UE to compress and feed back the CSI estimation error $\mathbf{V}_t$ to the gNB instead of the raw $\mathbf{H}_t$.

Based on this CSI model, we can develop a novel DL encoder and decoder architecture by exploiting a trainable neural network to learn the unknown MMSE estimation function $\phi(\mathbf{H}_{t-1}) = E\{\mathbf{H}_t | \mathbf{H}_{t-1}\}$. This new DL encoder and decoder architecture is shown in Fig. 1.

As shown in Fig. 1, the feedback for the CSI matrix sequence can be divided into two phases: a) The feedback of CSI at the first (initial) time slot ($t = 1$) without prior information; b) The feedback of CSI in subsequent time slots ($t = 2, 3, ..., T$) given the prior CSI information. Denote $\hat{\mathbf{H}}_t$ as the reconstruction of CSI matrix $\mathbf{H}_t$ at time slot $t$. Define the encoding and decoding function as $f_e(\cdot)$ and $f_d(\cdot)$, respectively. For downlink CSI feedback architecture in the first time slot, the encoder network and decoder network can be denoted, respectively, by

$$\mathbf{s}_1 = f_{e,1}(\mathbf{H}_1 - E\{\mathbf{H}_1\}), \tag{6}$$

$$\hat{\mathbf{H}}_1 = f_{d,1}(\mathbf{s}_1) + E\{\mathbf{H}_1\} \tag{7}$$

This initial step assumes that the CSI mean is known from training or past information. For downlink CSI feedback architecture of subsequent time slot $t$ ($t \geq 2$), the encoder network and decoder network can be executed, respectively, by

$$\mathbf{s}_t = f_{e,t}(\mathbf{H}_t - \phi(\hat{\mathbf{H}}_{t-1})), \quad (8)$$

$$\hat{\mathbf{H}}_t = f_{d,t}(\mathbf{s}_t) + \phi(\hat{\mathbf{H}}_{t-1}) \quad (9)$$

Since the optimum function $\phi(\hat{\mathbf{H}}_{t-1})$ is unknown, one direct solution is to approximate the function with deep neural network architecture trained by using a set of CSI samples.

## III. Exploiting Channel Temporal Coherence

We now discuss two avenues for exploiting the temporal coherence of CSIs at successive time-slots: a DNN architecture that utilizes long-short term memory (LSTM) layers and an information theoretic basis for differential encoding.

### A. Recurrent Neural Networks

Recurrent neural networks (RNNs) include layers which encode memory of previous states. Through backpropagation training, recurrent layers learn whether to incorporate information stored in memory in the layer's output and whether that information should be kept in memory [29]. The memory incorporation enables RNNs to store, remember, and process information that resides in past signals for long time periods. RNNs can utilize past input sequence samples to predict future states [30].

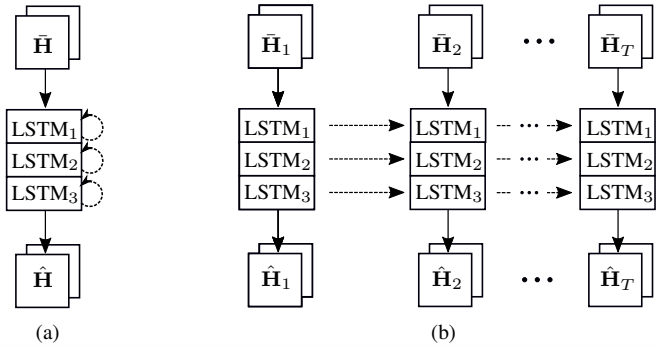



Fig. 2: (a) A "stacked" LSTM network of depth 3 shown with recurrent connections. (b) "Unrolled LTSM network" into $T$ timeslots. Training can use either full or quantized CSI.

RNNs have found wide applications in areas such as natural language processing (NLP), including machine translation [25] and sentiment extraction [26]. For NLP tasks, empirical results have demonstrated the effectiveness of "deep" or "stacked" RNNs, networks which use the outputs of hidden recurrent layers as inputs to subsequent recurrent layers [31].

Prior works have investigated stacked RNNs for CSI estimation. Several proposals have favored the use of Long Short Term Memory (LSTM) cell [19]–[21], a recurrent unit that can tackle the vanishing gradient problem inherent in recurrent backpropagation [32]. Existing LSTM-based works in CSI estimation have assumed that stacked LSTMs are better than shallow LSTMs, presenting models which used LSTM cells of depth 3 [19]. Fig. 2 demonstrates the principle of this LSTM network for CSI feedback and estimation. This bias towards deep RNNs is likely due to the aforementioned successes in NLP, where deep recurrent layers are theorized to learn hierarchical levels of semantic abstraction [26], [33].

This RNN approach has been recently proposed in [19]. In this work, we shall consider the proposed architecture of [19] as the benchmark method. However, deep LSTMs can be problematic, as the number of parameters per LSTM cell can be quite large. If a parsimonious model is desired due to memory constraints, then memory intensive RNNs can be very costly.

### B. CSI Entropy and Feedback Encoding

Despite the success of deep RNNs in CSI estimation and recovery, several important research questions remain.

- First, what simplifications can be made to reduce computational complexity while maintaining efficient CSI feedback and accurate CSI recovery?
- Second, how much CSI feedback in terms of bitwidth per CSI coefficient is sufficient?
- Third, how frequently should a UE provide CSI feedback for gNB to update its CSI estimate?

It is therefore important to tackle these open questions that hamper the practical application and efficacy of DL based CSI estimation and recovery in massive MIMO networks.

Consider random channel matrix $\mathbf{H}_t$ that consists of complex fading coefficients for the $t$-th timeslot. For joint probability density function $p(\mathbf{H}_t)$, the CSI entropy is

$$H(\mathbf{H}_t) = -\sum_{\mathbf{H}_t} p(\mathbf{H}_t) \log p\left(\mathbf{H}_t\right) \quad (10)$$

where (10) is the sum over all realizations of r.v. $\mathbf{H}_t$. The CSI entropy of (10) describes the required number of bits for the UE to feed back its CSI estimate to the gNB for reconstruction. Denote the $(i,j)$-th CSI element within $\mathbf{H}_t$ as $\mathbf{H}_{t,(i,j)}$ at time $t$. If all elements are independent, then we have a simple upper bound on the entropy of the full CSI matrix as

$$H(\mathbf{H}_t) \leq H_{\mathrm{UB}} = \sum_{i,j} H(\mathbf{H}_{t,(i,j)}) \quad (11)$$

This entropy bound $H_{\mathrm{UB}}$ describes the approximate number of total bits necessary for direct encoding of forward link CSI for UE feedback.

Fortunately, in mobile wireless networks, CSI within a coherence time exhibits strong correlation [34]. Therefore, instead of constructing CSI independently by relying on CSI feedbacks for individual time slots, the gNB can utilize this CSI dependency by leveraging both previously reconstructed CSIs and the current CSI feedback. In other words, the UE

feedback should focus on providing information that is not available at the gNB from CSIs of previous time slots.

Taking advantage of the Markovian CSI model, we can investigate how much the gNB can benefit from the previous CSI. Given the Markovian channel model of (3), the conditional CSI entropy quantifies the amount of information needed to characterize the CSI matrix based on the available CSIs from earlier reconstruction:

$$H(\mathbf{H}_t|\mathbf{H}_{t-1},\ldots,\mathbf{H}_1) = H(\mathbf{H}_t|\mathbf{H}_{t-1}) \quad (12)$$
$$= -\sum_{\mathbf{H}_{t-1}}\sum_{\mathbf{H}_t} p(\mathbf{H}_t)\log p(\mathbf{H}_t|\mathbf{H}_{t-1})$$

From the well known relationship of $H(\mathbf{H}_t|\mathbf{H}_{t-1}) \leq H(\mathbf{H}_t)$, it is clear that by utilizing the most recently reconstructed CSI, the gNB would require less feedback bandwidth and improve the UE feedback efficiency.

A stationary first order Markovian channel model is characterized by the conditional probability density function of $p(\mathbf{H}_t|\mathbf{H}_{t-1})$. In practice, such distribution information on CSI is difficult to acquire analytically. To gain valuable insights into the time-coherence between CSI at different feedback intervals, we shall provide a numerical evaluation of typical wireless channel models by comparing the entropy and the conditional entropy of the forward link CSI parameters. Note the following CSI entropy relationship at $t$ and $t-\delta$ where $\delta$ is the feedback interval:

$$H(\mathbf{H}_{t,(i,j)}|\mathbf{H}_{t-\delta}) \leq H(\mathbf{H}_{t,(i,j)}|\mathbf{H}_{t-\delta,(i,j)}) \leq H(\mathbf{H}_{t,(i,j)}). \quad (13)$$

For practical purposes, we consider the empirical mean conditional entropy of $H(\mathbf{H}_{t,(i,j)}|\mathbf{H}_{t-\delta,(i,j)})$ averaged over all coefficients in $\mathbf{H}_t$,

$$\hat{H}(\mathbf{H}_{t,(i,j)}|\mathbf{H}_{t-\delta}) = \frac{1}{R_d N_b}\sum_{i=1}^{R_d}\sum_{j=1}^{N_b} H(\mathbf{H}_{t,(i,j)}|\mathbf{H}_{t-\delta,(i,j)}). \quad (14)$$

Conditional entropy provides useful information on the degree of CSI dependency in time.

In this experiment, we consider the link with $N_b = 32$ transmit antennas and 1 receive antenna over $N_f = 1024$ subcarriers. After applying the 2D DFT, $R_d = 32$ rows of significant CSI elements in delay domain are retained in $\mathbf{H}_t$. Since the real and imaginary components of complex CSI matrices are typically treated as separate real-valued inputs to the neural network [14], [19]–[23], we consider the conditional entropy of the CSI's real and imaginary parts individually. Fig. 3 demonstrates the estimated conditional entropy averaged over the complex CSI matrix.

We generate 5,000 random indoor and outdoor channel responses using the channel models given in [35] and [19]. Following the examples in [19], the indoor channel is in the 5.3 GHz band, with little or no mobility at velocity of $10^{-3}$ m/s. The outdoor channel is in the 300 MHz band, at velocity of 0.9 m/s. The bandwidth for indoor and outdoor channels

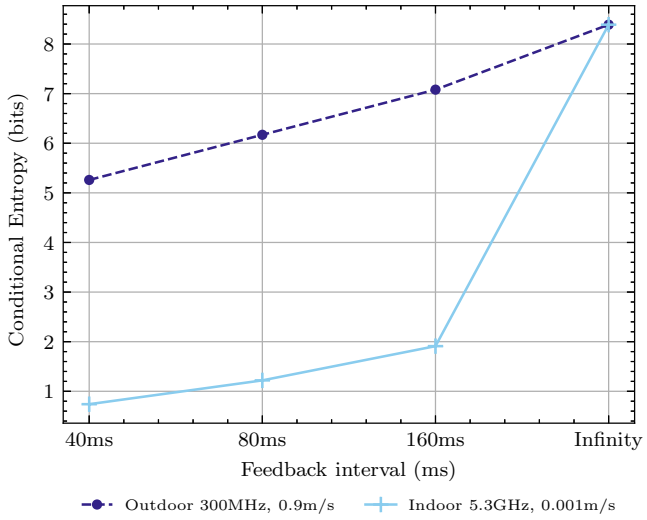



Fig. 3: Mean CSI conditional entropy per element for different intervals ($\delta$) in indoor and outdoor.

is 20 MHz. The conditional entropy is evaluated for different lengths of feedback interval $\delta$ = 40ms, 80ms, 160ms, and $\infty$ (i.e., no feedback) using the $k$-nearest-neighbor entropy estimator [36].

From Fig. 3, it is evident that the conditional entropy increases with $\delta$ because of the limited CSI coherence time. In addition, it is intuitive that the outdoor channel exhibits higher conditional entropy since higher velocity corresponds to shorter coherence time [37]. For both channel models, the average entropy of the CSI elements without prior CSI ($\delta = \infty$) is approximately 8 bits. However, the conditional entropy can be reduced based on prior CSI conditions, which clearly motivates the development of CSI feedback models based on the Markovian concept. For example, the outdoor channel CSI exhibits an average entropy reduction of nearly 1/4 per CSI coefficient if CSI matrix from 80ms ago is made available. Even more striking is the CSI dependency of low mobility indoor channels. For $\delta = 80$ ms, the conditional entropy is approximately reduced by 7/8 from the original CSI entropy. These examples from well known CSI channel models support the development of a Markovian learning model for efficient CSI feedback.

The entropy reduction under conditions of known prior CSI knowledge motivates the idea of condition-based encoding such as differential encoding by the UE. Encoding the difference between successive feedback instants, $\mathbf{H}_t$ and $\hat{\mathbf{H}}_{t-\delta}$, can reduce the number of UE feedback bits, facilitating high degree of compression without loss of CSI estimation performance [38].

## IV. Differential CSI Encoding

### A. A Simplified Markovian Model

In contrast to RNNs which, as a black box, would require a large number of DL parameters to autonomously learn

the underlying temporal correlation among channels of the multiple timeslots, we leverage the practical insight into CSI in terms of time dependency and conditional entropy to develop a structured differential autoencoder which is based on a simplified Markovian model. The first order Markovian CSI model [39] is selected to leverage the temporal correlation while reducing the complexity:

$$\mathbf{H}_t = \gamma \mathbf{H}_{t-1} + \mathbf{V}_t, \tag{15}$$

where $\gamma$ is a constant and $\mathbf{V}_t$ is a zero-mean i.i.d. random matrix. Given ensemble samples of $\mathbf{H}_{t-1}$ and $\mathbf{H}_t$, the unknown $\gamma$ can be estimated via

$$\hat{\gamma} = \frac{\text{Trace}(E\{\mathbf{H}_t \mathbf{H}_{t-1}^H\})}{E\|\mathbf{H}_{t-1}\|^2}. \tag{16}$$

Based on this first order autoregressive (AR) model, we formulate a low complexity encoder-decoder model for time slot $t$ ($t \geq 2$) as

$$\mathbf{s}_t = f_{e,t}(\mathbf{H}_t - \gamma \hat{\mathbf{H}}_{t-1}), \tag{17}$$

$$\hat{\mathbf{H}}_t = f_{d,t}(\mathbf{s}_t) + \gamma \hat{\mathbf{H}}_{t-1} \tag{18}$$

Based on this simplified model, we propose a differential encoding architecture, "MarkovNet," for efficient CSI feedback and reconstruction in the massive MIMO systems. To fully exploit the temporal CSI coherence, MarkovNet requires accurate CSI at the initial time slot $t_1$ to establish sufficient baseline information for the CSI feedback in subsequent time slots. To this end, the proposed MarkovNet framework shall apply CSI pre-processing and improve the neural network structure. Specifically,

- To pre-process CSI data, we use spherical normalization which makes the objective function more applicable to commonly adopted accuracy metric of NMSE [27].
- We propose a enhanced autoencoder, CsiNet-Pro, which includes a deeper encoder with more convolutional layers to better extract features of CSI and a symmetric decoder for CSI decoding. The details are provided in Section IV.C.

### *B. Transforming CSI Feedback in Spherical Coordinate*

How to effectively apply DL techniques to exploit channel data properties and optimization objects remains an open research issue, as many existing DL based works mainly utilize the deep learning architectures and optimization functions successfully developed for other application areas. Direct adoption of DL architectures without customization for CSI data characteristics risks unsatisfactory performance. In particular, data processing methods and loss functions developed for computer vision may not be well suited for CSI encoding and reconstruction.

To begin, many existing DL-based CSI encoding-decoding schemes conveniently view the 2D MIMO channel matrix $\mathbf{H}_t$ as akin to an image such that the normalized elements of the CSI matrix are utilized as image-like input to DL networks in both training and testing. However, the distribution of multipath fading MIMO channels differs substantially from the distribution of 2D image data.

Among other differences, images are represented as matrices of intensity pixel values. For color images, each color channel corresponds to a 2D matrix of pixel values that are unsigned integers, e.g., in the range between 0 and 255. By normalizing these pixels, there can be strong benefit in preparing the images as inputs of the DL model. However, unlike different images whose pixel values are mostly in the same order of magnitude, the dynamic range of CSI data can be much greater. For example, RF pathloss grows polynomially with distance between gNB and UE [40]. As a result, CSI of one UE can be different from CSI of another UE by several orders of magnitude, depending on their respective distances to gNB. A naive normalization can render CSIs of some UEs too small for the DL networks to respond to. Another different feature is that the baseband CSI parameters are complex values, consisting of both magnitude and phase, whereas image pixels are nonnegative real (with normalization).

In terms of learning objectives, several existing DL-based CSI feedback works adopt the loss function similar to image recovery for training the DL networks. Specifically, the objective is to minimize the mean square error (MSE):

$$\text{MSE} = \frac{1}{N} \sum_{k=1}^{N} \|\mathbf{H}_k - \hat{\mathbf{H}}_k\|^2, \tag{19}$$

where $k$ and $N$ are, respectively, the index and total number of samples in the data set, and $\|\cdot\|$ denotes the Frobenius norm. On the other hand, it is typically more meaningful in CSI estimation to apply the normalized MSE (NMSE)

$$\text{NMSE} = \frac{1}{N} \sum_{k=1}^{N} \frac{\|\mathbf{H}_k - \hat{\mathbf{H}}_k\|^2}{\|\mathbf{H}_k\|^2}, \tag{20}$$

to assess the accuracy of CSI recovery at the gNB [41] and feedback efficiency [14], [15], [19]. By directly using MSE as the loss function, the DL networks would be biased toward the CSI accuracy of stronger MIMO channels.

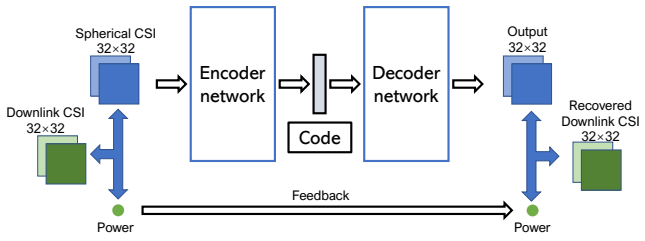



Fig. 4: Architecture of spherical CSI feedback in SphNet.

In response to the domain-specific characteristics of data and objective in CSI estimation, we propose to use a spherical CSI data structure for feedback as shown in Fig. 4. The spherical CSI feedback architecture splits the downlink CSI matrix $\mathbf{H}_k$ into a power value $p_k$ and a spherical downlink

CSI matrix $\check{\mathbf{H}}_k$, where $p_k = \|\mathbf{H}_k\|$ is the power of the CSI matrix and $\check{\mathbf{H}}_k = \mathbf{H}_k/\|\mathbf{H}_k\|$ is the the unit norm spherical CSI. Note that the elements of $\check{\mathbf{H}}_k$ remain strictly inside or on the surface of the unit hyper-sphere. The UE encodes and sends back the power $p_k$ and the spherical CSI matrix $\check{\mathbf{H}}_k$.

Spherical CSI feedback architecture presents numerical advantages. First, we can construct an encoder DL network that focuses on compressing and encoding the spherical CSI matrix $\check{\mathbf{H}}_k$. The power of the CSI would be directly sent back to the gNB separately since it contains little redundancy. Thus, even for CSI matrices of different magnitudes, they are equally important in training the encoder and decoder networks. During training the gradients for UEs that are far away from the gNB would no longer be negligible [42]. Moreover, the decoder will have a more limited domain for more accurate CSI recovery under spherical normalization [27].

As shown in Fig. 4, our joint CSI compression and reconstruction architecture still utilizes the effective autoencoder structure in which the encoder at the UE attempts to learn a low-dimensional CSI representation for a relatively high-dimensional dataset represented in the form of spherical CSI matrices. The decoder at the gNB reconstructs the CSI matrix based on feedback information extracted from the UE encoder and the direct feedback of CSI magnitude $p_k$.

### C. CsiNet Pro: An Enhanced CSI Encoder-Decoder Network

We propose an efficient neural network structure, named CsiNet Pro, for UE encoding and gNB decoding of CSI in massive MIMO networks. The structure of CsiNet Pro is illustrated in Fig. 5. In comparison with existing neural networks such as those from [14] [19], CsiNet Pro provides a deeper encoder that uses more convolutional layers to better extract features of CSI. There is a corresponding decoder at the gNB that also contains 4 convolution layers.

The design of encoder for dimension compression is crucial. However, the encoders in [14], [15], [19] all utilized one convolutional layer and one fully connected layer. As a major departure, the encoder of CsiNet Pro utilizes 4 convolutional layers for feature extraction and 1 fully connected layer for dimension compression. Specifically, the 4 convolutional layers apply $7 \times 7$ kernels to generate 16, 8, 4 and 2 feature maps, respectively (see Fig. 5).

Another change in CsiNet Pro is the use of a different normalization range and output activation function. Recall that the decoder network utilizes 4 convolutional layers as shown in Fig. 5. Unlike the nonnegative pixel values in image reconstruction, CSI values contain both real and imaginary parts that can be either positive or negative. Thus, unlike previous works that normalize the CSI values to fall within $[0, 1]$ in order to use "sigmoid" or "ReLU" as the activation function of the last layer, our proposed CsiNet Pro normalizes the real and imaginary CSI values to the range [-1, 1] while using "tanh" as its activation function in the last layer.

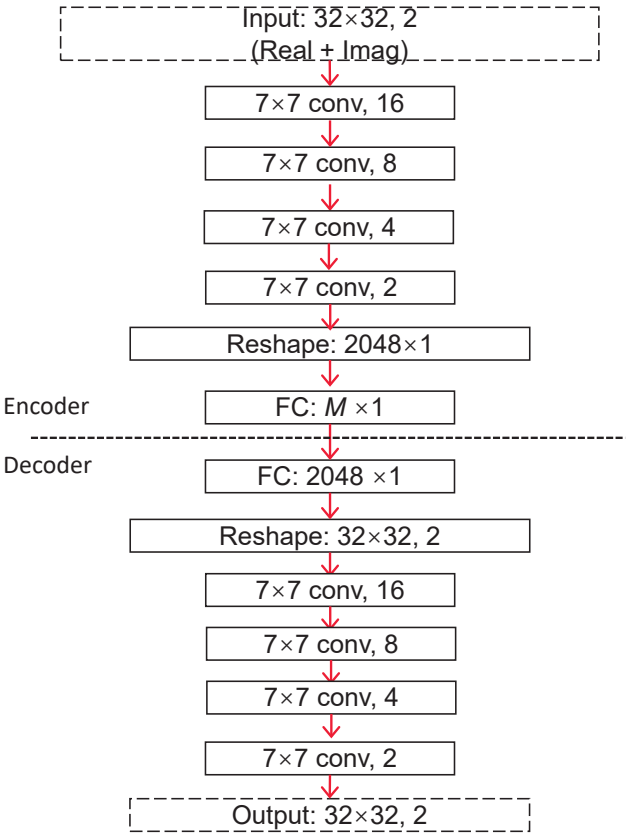



Fig. 5: Architecture of CsiNet Pro.

We integrate CsiNet Pro (Fig. 5) with the spherical CSI feedback framework (Fig. 4) to provide enhanced CSI recovery accuracy.

### D. Differential CSI Encoding

Motivated by the simplified first order AR model for CSI, we propose a differential CSI feedback framework MarkovNet to improve bandwidth efficiency. Different from the RNN based networks such as LSTM which relies on neural networks to learn the required information sharing and corresponding CSI compression simultaneously, MarkovNet proactively leverages the simplified AR model (15) for CSI and encodes the CSI prediction error as shown in (17) between two successive time slots.

Recall that the difference based on first order estimation of the CSI in two adjacent time slots $\mathbf{H}_t - \hat{\gamma}\mathbf{H}_{t-1}$ is an approximation of the innovation $\mathbf{V}_t$. As shown in Fig. 6, for time-slots beyond the initial time-slot, the linear prediction difference $\mathbf{H}_t - \hat{\gamma}\mathbf{H}_{t-1}$ is sent to the encoder network to execute the encoding process of $\mathbf{s}_t = f_{e,t}(\mathbf{H}_t - \gamma\hat{\mathbf{H}}_{t-1})$ given in (17). At the gNB receiver, the decoder network can utilize the previously recovered CSI $\hat{\mathbf{H}}_{t-1}$ to reconstruct $\hat{\mathbf{H}}_t$ according to $\hat{\mathbf{H}}_t = f_{d,t}(\mathbf{s}_t) + \gamma\hat{\mathbf{H}}_{t-1}$ as described in (18).

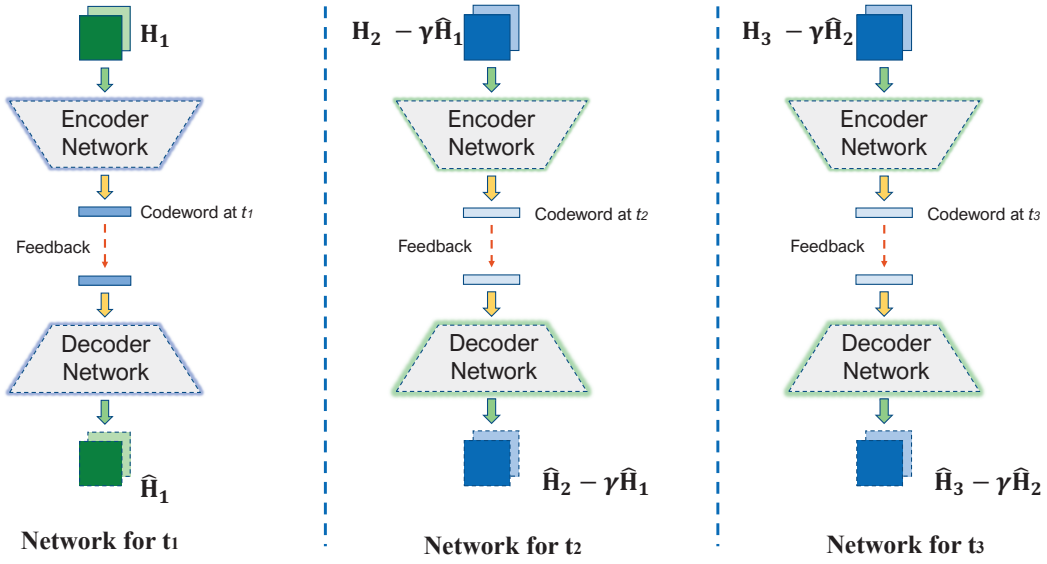



Fig. 6: Illustration of the multi-stage, differential CSI feedback framework MarkovNet.

MarkovNet from $t_2$ onward would employ the same network architecture CsiNet Pro as shown in Fig. 5. Compared with network for $t_1$ which uses a larger compression ratio to ensure the high recovery accuracy in the first timeslot, MarkovNet from $t_2$ can afford a smaller compression ratio to achieve a higher bandwidth efficiency with the help of prior information.

MarkovNet exhibits several additional advantages in practical implementation. First, compared to RNN-based CSI feedback, MarkovNet can exploit pretrained model as initial neural network parameters for models used in later timeslots to improve training efficiency since the CSI at adjacent time slots share similar data features. Second, differential CSI matrices tend to be more sparse, hereby enabling MarkovNet to achieve a higher degree of compression during feedback. Third, for most wireless network applications, both gNB and UEs have limited power, computation, and storage resources. MarkovNet simplifies the learning tasks of neural networks and is more applicable in a wider variety of wireless deployment scenarios.

## V. Model Reduction

Practical implementation of deep neural networks for CSI feedback and recovery can be challenging to some mobile devices. Because DL network architectures often use large numbers of parameters, they require substantial computation and memory resources. Unrolled RNN models, such as the LSTM layers in Fig. 2 are particularly computationally expensive. For example, CsiNet-LSTM [19] at a compression ratio (CR) of $1/16$ contains $1.19 \times 10^8$ parameters per timeslot. One of the main advantages of MarkovNet (see Fig. 6) is its relatively low parameter count, as a comparable version of MarkovNet at a CR of $1/16$ has $5.43 \times 10^5$ parameters per timeslot, a reduction of three orders of magnitude relative to CsiNet-LSTM.

Our proposed MarkovNet can clearly reduce the model size by eliminating the repeated structure used to learn the from the sequence data in RNN-style architecture. It is important to note, however, the fully connected (FC) layers for dimension compression and decompression in the current MarkovNet still contains a large number of parameters. For example, there are more than $10^6$ parameters for the FC layers at CR = $1/8$.

FC layers for dimension compression and decompression, as shown in Fig. 7(a), have often been adopted in deep learning based CSI feedback [14], [15], [19]–[21]. However, elements of the CSI matrix only exhibit strong correlation with its neighbors in angular-delay domain. Thus, we recognize that the FC layers, though effective and popular, still contain a large fraction of non-essential connections with very weak weight parameters. This realization presents another opportunity for model reduction. To further reduce model size, we propose a CNN-based latent structure to replace the FC layers for dimension compression. As shown in Fig. 7(b), we slice the two square feature maps into 64 feature maps of dimension $1 \times 32$. We then design $M$ CNN kernels of length $1 \times 7$ to compress the codewords dimension. The integer $M$ is adaptive in accordance with the encoder compression ratio denoted by $\frac{M}{64}$. Through this feature processing, connections between CSI elements that are far apart in the angular-delay domain are removed. Strongly correlated features of CSI matrix across the angular-delay domain can effectively be captured by the small CNN kernels.

To illustrate the effect of the proposed model size reduction, we summarize the number of parameters and the floating point operations (FLOPs) in Table I. This information provides a

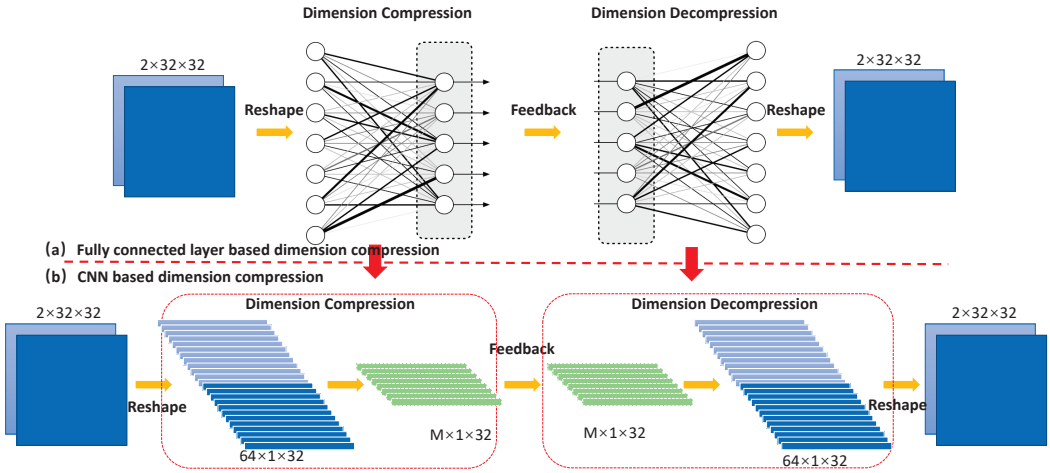



Fig. 7: Proposed CNN-based dimension compression and decompression module.

TABLE I: Number of parameters and FLOPs comparison for FC-based and proposed CNN-based dimension compression and decompression module. M: million, K: thousand.

| | Number of parameters | | FLOPs | |
|---|---|---|---|---|
| | **FC-based** | **Proposed** | **FC-based** | **Proposed** |
| **CR=**1/4 | 2.1 M | 14.4 K | 4.2 M | 0.9 M |
| **CR=**1/8 | 1.1 M | 7.2 K | 2.1 M | 0.5 M |
| **CR=**1/16 | 0.5 M | 3.7 K | 1.0 M | 0.2 M |

comparison of the storage size and computational complexity between the use of FC-layer and proposed CNN-layer in CSI compression module and the corresponding decompression module. As shown in Table I, the proposed CNN-based dimension compression and decompression module reduces the number of parameters by over 100 times and the number of FLOPs by at least 4 times. The comparison results demonstrate that our new CNN design for CSI compression and decompression represents an important step in broadening the range of practical applications for effectively deploying deep learning based CSI encoding, feedback, and reconstruction in massive MIMO wireless systems. MarkovNet using CNN-based dimension compression and decompression module is named as MarkovNet-CNN.

## VI. Performance Evaluation

We assess the performance of both RNN-based CsiNet-LSTM [19] and MarkovNet for two different massive MIMO scenarios generated from the well known COST 2100 model [35] through a range of experiments. Section VI-A defines the parameters for the simulations and hyperparameters used in evaluations. Section VI-B discusses the single-timeslot performance of MarkovNet (Section VI-B1), the overall performance of MarkovNet (Section VI-B2) and MarkovNet-CNN (Section VI-B3), and the performance of MarkovNet for larger antenna arrays (Section VI-B4). Section VI-C compares the computational complexity and parameter count of MarkovNet with CsiNet-LSTM. Section VI-D shows the performance of MarkovNet and CsiNet-LSTM under feedback quantization. Section VI-E demonstrates the performance of MarkovNet under different feedback intervals.

### A. Evaluation Parameters

For the COST2100 model, we generate data for two different channel environments which are typically used for assessing the performance of CSI estimation techniques [14], [18], [19]:

1) **Indoor** channels using a 5.3GHz downlink at 0.001 m/s UE velocity, served by a gNB at center of a 20m×20m coverage area.
2) **Outdoor** channels using a 300MHz downlink at 0.9 m/s UE velocity served by a gNB at center of a 400m×400m coverage area.

We give $N_b = 32$ antennas to the gNB to serve single antenna UEs randomly distributed within the coverage area. We use $N_f = 1024$ subcarriers and truncate the delay-domain CSI matrix to include the first $R_d = 32$ rows.

The gNB uses antennas arranged in a uniform linear array (ULA) with half-wavelength spacing. UEs are randomly positioned within the coverage area such that their CSIs are random. For each indoor/outdoor environment, we generate a dataset of $10^5$ sample channels and divide them into $7.5 \cdot 10^4$ and $2.5 \cdot 10^4$ for training and testing sets, respectively. The batch size for the training of MarkovNet is 200. MarkovNet at $t_1$ was trained for 1000 epochs using MSE as the loss function. For the MarkovNet after $t_2$, only 150 epochs are used with the help of initialization using the pretrained model of the previous time slot to reduce training expenses. We utilize the Adam optimizer with default learning rate $10^{-3}$. For CsiNet-LSTM, we utilize the hyperparameters outlined in the original

paper [19] with the exception of the batch size, which was reduced from 200 to 100 due to memory constraints.

To compare the recovery accuracy of different networks, the NMSE metric is adopted. Unless noted otherwise, all evaluations in this section report the NMSE of the *entire* CSI matrix ($\text{NMSE}_{\text{all}}$) rather than the NMSE of the truncated CSI matrix ($\text{NMSE}_{\text{truncate}}$). Denote the dropped CSI elements of the $k$-th channel sample as $\mathbf{H}_{k,\text{drop}}$. $\text{NMSE}_{\text{all}}$ is given as

$$\text{NMSE}_{\text{all}} = \frac{1}{N}\sum_{k=1}^{N}\frac{\|\mathbf{H}_k - \hat{\mathbf{H}}_k\|^2 + \|\mathbf{H}_{k,\text{drop}}\|^2}{\|\mathbf{H}_k\|^2 + \|\mathbf{H}_{k,\text{drop}}\|^2}. \tag{21}$$

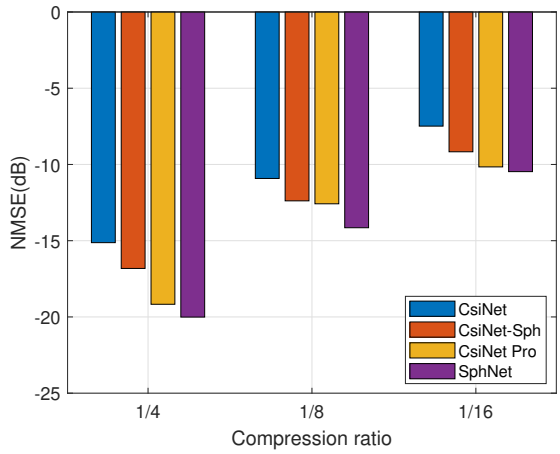
Observe that (21) is similar to (20) with the only difference being the $\|\mathbf{H}_{k,\text{drop}}\|^2$ terms. Since prior works in CSI estimation [14], [19] are used to estimate the truncated CSI matrix of size $(R_d \times N_b)$, these works report only $\text{NMSE}_{\text{truncated}}$ (i.e., (20)) in their numerical results. Since $\text{NMSE}_{\text{all}}$ accounts for the error due to setting low-magnitude values of full CSI matrices to zero, $\text{NMSE}_{\text{all}}$ is a better indicator of the CSI estimator's performance than $\text{NMSE}_{\text{truncated}}$. For the Outdoor (Indoor) network, $\text{NMSE}_{\text{all}}$ represents an average increase of $2.5 \times 10^{-2}$ ($7.0 \times 10^{-3}$) relative to $\text{NMSE}_{\text{truncated}}$ on a linear scale.

### B. MarkovNet

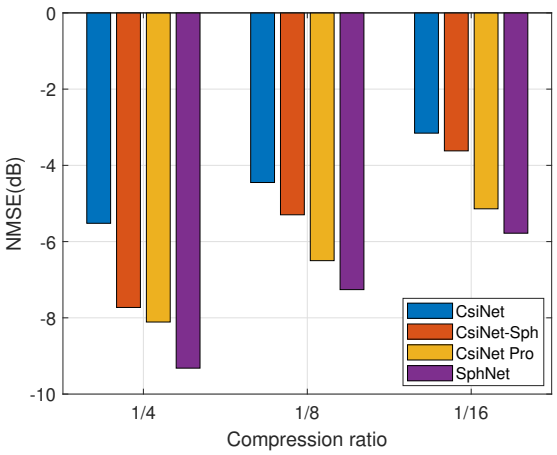
*1) Performance evaluation at $t_1$:* To enable efficient differential CSI feedback, high accuracy CSI feedback is required at $t_1$ to provide a good starting CSI condition for subsequent timeslots. Here, we demonstrate that our proposed spherical CSI feedback framework improves the CSI recovery accuracy for a single time slot compared to different CSI feedback frameworks.

Fig. 8 compares the performance of channel reconstruction from the use of CsiNet [14], CsiNet-Sph (CsiNet with spherical feedback), CsiNet Pro, and SphNet (CsiNet Pro with spherical feedback). As shown in Fig. 8, SphNet achieves the best performance in single shot feedback for CSI recovery without relying on prior CSI knowledge, which means that SphNet can improve the accuracy of prior information for the MarkovNet. On the one hand, CsiNet Pro outperforms CsiNet at all CR in both channel environments, which means the enhanced network structure is effective. On the other hand, we can observe that spherical feedback can provide the most noticeable performance gain to both CsiNet and CsiNet Pro. This establishes the strength of spherical normalization to efficiently capture the CSI data feature.

*2) Overall performance evaluation of MarkovNet:* Every instance of MarkovNet contains two different compression ratios for practical implementation. For the first time slot, we initialize MarkovNet with CR=$1/4$ at timeslot $t_1$ to provide an accurate starting CSI. For all subsequent timeslots ($t_2$ to $t_{10}$), MarkovNet maintains the same CR. For example, in Fig. 9 that follows, "MarkovNet, CR=$1/16$" uses CR=1/16 at timeslots $t_2$ through $t_{10}$ and CR=1/4 at timeslot $t_1$. To evaluate MarkovNet's performance under different levels of




(a) Indoor $\text{NMSE}_{\text{all}}$



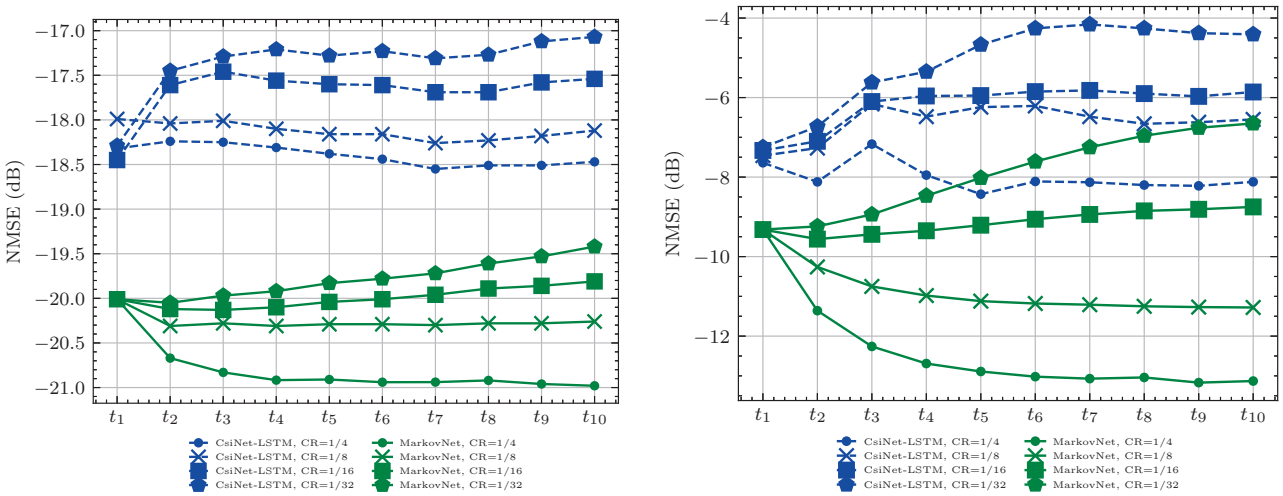

(b) Outdoor $\text{NMSE}_{\text{all}}$

Fig. 8: NMSE of different networks in the first time slot over varying compression ratios (CR).

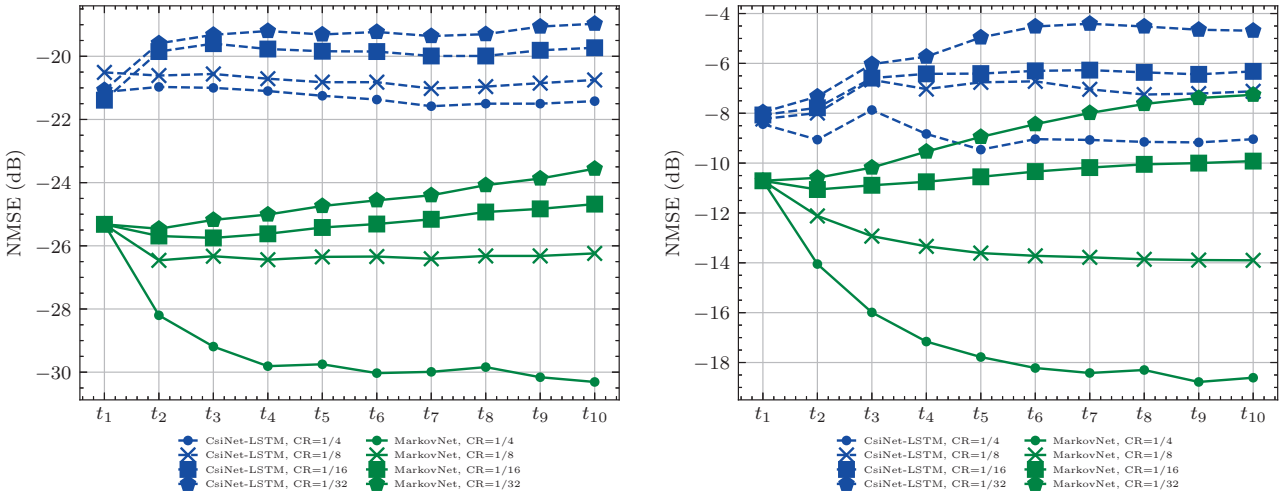
compression, we vary the second CR used in timeslots $t_2$ to $t_{10}$ from $1/4$ to $1/64$ and train each network.

Fig. 9 compares the performances of MarkovNet and CsiNet-LSTM. The benefit of differential CSI encoding in MarkovNet can be seen from its improved CSI recovery accuracy at different compression ratios beyond $t_2$. MarkovNet utilizes a structured approach and consistently achieves higher CSI accuracy than the blackbox CsiNet-LSTM at every CR level. For the indoor channels, MarkovNet can deliver reliable CSI accuracy $\text{NMSE}_{\text{all}}$ of $-20$dB even for CR $= 1/32$, a 3dB improvement over CsiNet-LSTM. Although the outdoor scenario continues to be more challenging, our results show that CR $\in [1/4, 1/8]$ can achieve $\text{NMSE}_{\text{all}}$ of $-13$dB and $-11$dB, respectively, much lower than the NMSE of $-8$dB and $-6.5$dB from CsiNet-LSTM, respectively.

To demonstrate the difference between $\text{NMSE}_{\text{truncated}}$ and $\text{NMSE}_{\text{all}}$, Fig. 9 (c) and (d) show the performance comparsion using the metric $\text{NMSE}_{\text{truncated}}$ used in [14], [16], [18], [19]. On




(a) Indoor $\text{NMSE}_{\text{all}}$ (b) Outdoor $\text{NMSE}_{\text{all}}$




(c) Indoor $\text{NMSE}_{\text{truncated}}$ (d) Outdoor $\text{NMSE}_{\text{truncated}}$

Fig. 9: NMSE comparison of MarkovNet and CsiNet-LSTM at various compression ratios (CR).

the one hand, MarkovNet at a given CR consistently achieves higher CSI accuracy than CsiNet-LSTM at the same CR. On the other hand, compared with $\text{NMSE}_{\text{truncated}}$, the performance improvement in the $\text{NMSE}_{\text{all}}$ is much slower, which means that it is not necessary to insist on optimizing the neural networks to reduce the $\text{NMSE}_{\text{truncated}}$ under the influence of truncation error. Note that in the following subsections, all results reporting NMSE correspond to $\text{NMSE}_{\text{all}}$.

A related metric, i.e., the cosine similarity defined in [19] is also calculated for comparison

$$\rho = \mathbb{E}\left\{\frac{1}{T}\frac{1}{N_f}\sum_{t=1}^{T}\sum_{n=1}^{N_f}\frac{\left|\hat{\mathbf{h}}_{t,m}^{H}\mathbf{h}_{t,m}\right|}{\left\|\hat{\mathbf{h}}_{t,m}\right\|\left\|\mathbf{h}_{t,m}\right\|}\right\}, \tag{22}$$

where $\hat{\mathbf{h}}_{t,m} \in \mathbb{C}^{N_b \times 1}$ denotes the channel vector of the $m-$th subcarrier at time $t$. When the gNB uses $\mathbf{v}_{t,m} = \hat{\mathbf{h}}_{t,m}/\left\|\hat{\mathbf{h}}_{t,m}\right\|$ as a beamforming vector (i.e., as in zero-forcing precoding), cosine similarity can be used to indicate the beamforming gain. Table II compares the cosine similarity for CsiNet-LSTM and MarkovNet. As shown in Table II, and consistent with the NMSE results, MarkovNet achieves higher cosine similarity at every CR level.

TABLE II: Cosine similarity of MarkovNet and CsiNet-LSTM for indoor and outdoor environments at various compression ratios (CR).

| | **Indoor** | | **Outdoor** | |
|---|---|---|---|---|
| **CR** | **CsiNet-LSTM** | **MarkovNet** | **CsiNet-LSTM** | **MarkovNet** |
| $\frac{1}{4}$ | 0.991 | 0.993 | 0.914 | 0.967 |
| $\frac{1}{8}$ | 0.990 | 0.992 | 0.885 | 0.955 |
| $\frac{1}{16}$ | 0.990 | 0.992 | 0.871 | 0.934 |
| $\frac{1}{32}$ | 0.989 | 0.992 | 0.831 | 0.911 |

Fig. 10 displays the magnitude of the reconstructed CSI under CR $= 1/4$ in comparison with the original CSI for a single Outdoor channel sample from the test set. Both MarkovNet and CsiNet-LSTM networks can learn to encode the two major peak magnitude regions of the CSI, but MarkovNet is able to recover a third peak magnitude region where CsiNet-LSTM fails.

*3) Performance and Complexity Trade-off of MarkovNet-CNN:* Fig. 11 shows the performance comparison between MarkovNet and MarkovNet-CNN at different meaningful compression ratios. Since the variations of CSI accuracy over time are similar, we focus on the performance from $t_1$ to $t_7$. For the first time slot $t_1$, we initialize MarkovNet and MarkovNet-CNN with CR=$1/4$ to provide an accurate CSI startup for subsequent time slots. Both MarkovNet and MarkovNet-CNN achieve comparable CSI accuracy at $t_1$, demonstrating that our proposed CNN layer for compression and decompression is not only more efficient in memory and computation, but also delivers similar CSI accuracy. MarkovNet maintains the same CR for all subsequent timeslots ($t_2$ to $t_7$). Interestingly, MarkovNet-CNN achieves modestly higher accuracy beyond $t_2$, for indoor channels when CR=$1/8$ and $1/16$ as shown in Fig. 11(a). This slight edge by MarkovNet-CNN for the indoor channels likely arises from the substantial reduction of redundant weights from the FC layer and the number of local minima, capable of exploiting the more sparse indoor CSI matrices [14]. For outdoor channels, MarkovNet-CNN achieves CSI accuracy comparable to MarkovNet for compression ratio of $1/8$ and $1/16$ while exhibiting a modest loss of accuracy at CR=1/4. Because CSI of outdoor channels is more complex and less sparse relative to indoor channels, networks estimating outdoor CSI could benefit more from a higher number of connectivity in compression and feature extraction layers.

*4) Performance under larger antenna counts:* We demonstrate the performance of MarkovNet for 48 antennas in the Outdoor channel environment (see Fig. 12). We only conduct tests for the Outdoor network since larger antenna arrays are impractical for most realistic indoor channel environments. These results suggest that a larger antenna count does not negatively impact MarkovNet's estimation performance.

### C. Model Size and Computational Complexity

We demonstrate that latent convolutional layers require significantly fewer parameters than FC-layers without loss of performance. Table III compares the model size and computational complexity (respectively) of CsiNet-LSTM, MarkovNet, and MarkovNet-CNN associated with a single timeslot. Among the tested compression ratios, MarkovNet uses $\frac{1}{60}$ of the parameters in comparison to CsiNet-LSTM. More importantly, MarkovNet-CNN further reduces the number of parameters to $\frac{1}{3000}$ of what is needed by CsiNet-LSTM while achieving similar or better CSI recovery accuracy. We further provide the parameters of several related networks that do not exploit temporal correlation (CsiNet [14], CRNet [16], and Deep AE [23]) in Table III for a more comprehensive comparison. We observe MarkovNet uses similar number of parameters, whereas MarkovNet-CNN requires significantly fewer parameters with the help of the proposed CNN-based dimension compression and decompression modules.

Table III also presents the average number of floating point operations (FLOPs) associated with a single timeslot for each learning model [43], [44]. MarkovNet and MarkovNet-CNN can reduce the computation load by more than $\frac{8}{9}$ and $\frac{9}{10}$ in FLOPs, respectively, in comparison with the CsiNet-LSTM for each compression ratio. We also include the FLOPs of CsiNet, CRNet and Deep AE which do not exploit temporal correlation in Table III. We observe that both MarkovNet and MarkovNet-CNN use 6-10 times FLOPs in comparison to the above three networks owing to the sizable CNN kernel. Practically, a number of works have examined the computation complexity of CNN, including convolution factorization [45], depth-wise separable convolution [46], etc. Since MarkovNet focuses on exploiting channel temporal coherence more efficiently, we leave the optimization of CNN factorization part to our future work.

We note that when deploying MarkovNet and MarkovNet-CNN as a cooperative learning mechanism at both UE and gNB, 50% additional parameters and FLOPs are required in comparison with the training phase. This is because the trained decoder must be duplicated at the UE side to generate the decoded CSI for the previous time slot used by the encoder. Despite this additional cost, both MarkovNet and MarkovNet-CNN still can reduce the number of parameters by orders of magnitude, and save over $\frac{5}{6}$ FLOPs in comparison with CsiNet-LSTM.

### D. Network Performance Under Feedback Quantization

To understand the effect of feedback quantization, we apply $\mu$-law companding to the encoded layer of both tested networks. $\mu$-Law companding uses a logarithmic transformation that emphasizes lower magnitude samples. For signal value $x$, the compression portion of the $\mu$-law scheme is written as

$$f(x) = \frac{\text{sgn}(x)\ln(1+\mu|x|)}{\ln(1+\mu)},\ 0 \le |x| \le 1. \tag{23}$$

Uniform quantization is applied to the compressed signal. For signal value $x$, the quantization/dequantization operation produces a value $\hat{x}$, which can be written as $\hat{x} = \Delta \left\lfloor \frac{f(x)}{\Delta} \right\rceil$ for fixed step size $\Delta$. After the quantized feedback is received, then we expand the result using the inverse of (23),

$$F(\hat{x}) = \frac{\text{sgn}(\hat{x})((1+\mu)^{|\hat{x}|}-1)}{\mu},\ -1 \le y \le 1. \tag{24}$$

Fig. 13(a) and Fig. 13(b) show the performance of MarkovNet and CsiNet-LSTM with $\mu$-law companding and fixed quantization step size at two different quantization levels, 6 bits and 4 bits, in comparison to the non-quantized network

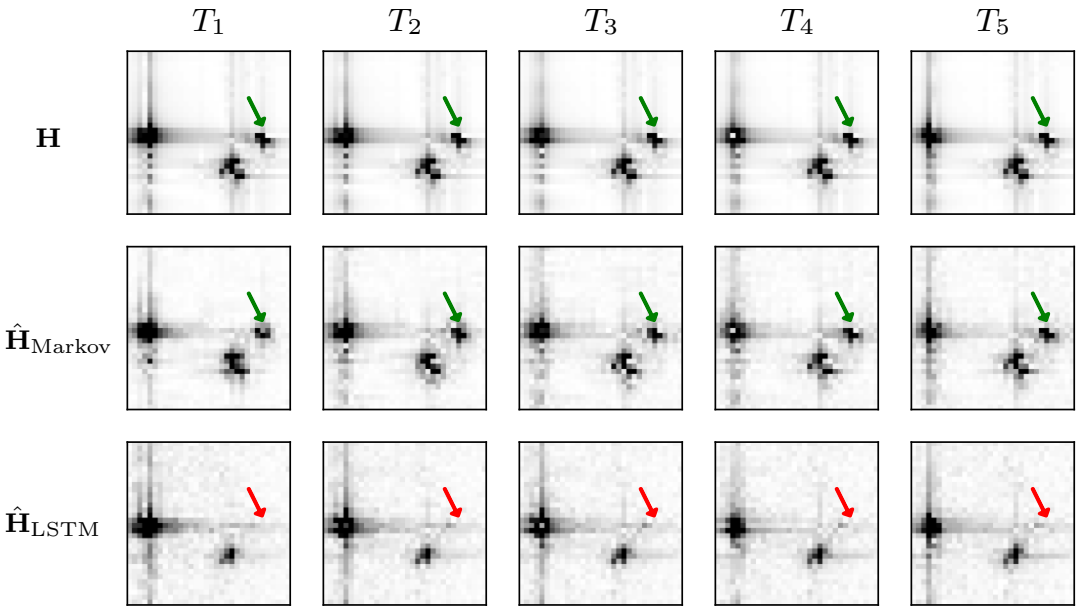
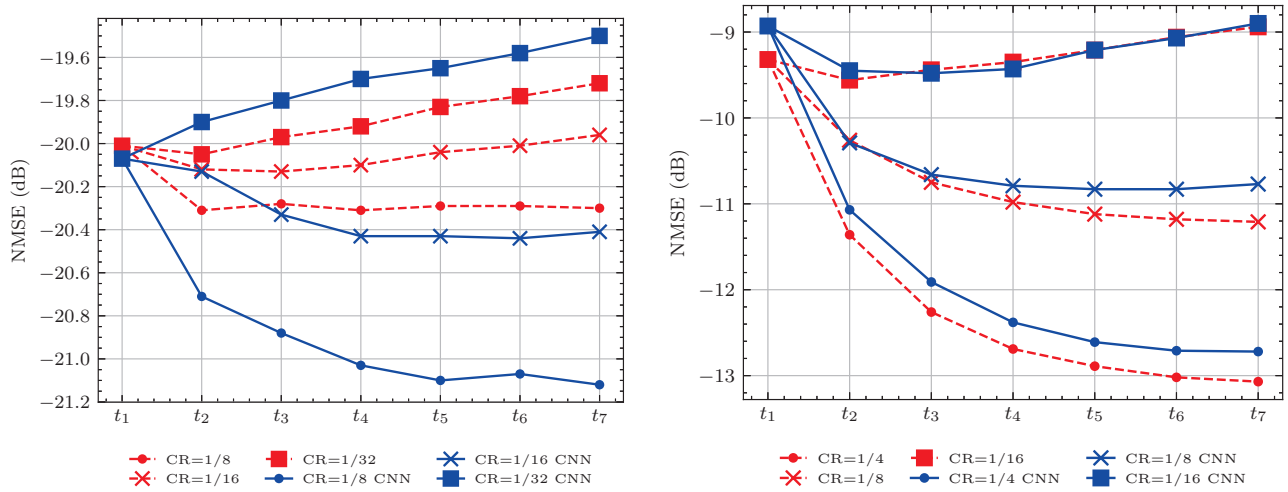



Fig. 10: Ground truth CSI ($\mathbf{H}$), MarkovNet estimates ($\hat{\mathbf{H}}_{\text{Markov}}$), and CsiNet-LSTM estimates ($\hat{\mathbf{H}}_{\text{LSTM}}$) across five timeslots ($T_1$ through $T_5$) on one outdoor channel sample from the test set, using CR $= \frac{1}{4}$. MarkovNet is able to recover a fine-grained region as highlighted by the green arrow while CsiNet-LSTM fails to recover the same region as highlighted by the red arrow.




(a) Indoor

(b) Outdoor

Fig. 11: NMSE comparison between MarkovNet and MarkovNet-CNN over varying CR.

(i.e., 32 bit floating point). The networks with quantized feedback use 8 bit quantization at the first timeslot to establish good intial CSI estimates. Note that the networks are not re-trained or fine-tuned after applying quantization. MarkovNet is more robust to feedback quantization noise than CsiNet-LSTM. For the Indoor environment, MarkovNet maintains NMSE better than -15 dB for 4 bit quantization while CsiNet-LSTM error is above -15 dB. For the Outdoor network, MarkovNet's performance at CR $= \frac{1}{32}$ is close to CsiNet-LSTM's performance at CR $= \frac{1}{4}$, meaning MarkovNet can achieve the same performance under 8 times as much compression.

### *E. Compression Ratio vs. Conditional Entropy*

Based on the conditional entropy $\hat{H}(\mathbf{H}_t|\mathbf{H}_{t-\delta})$ for different feedback intervals, $\delta$, it is intuitive that larger $\delta$ leads to higher conditional entropy, requiring a lower level of CSI compression feedback. Similarly, indoor channels of lower mobility exhibit smaller conditional entropy $\hat{H}(\mathbf{H}_t|\mathbf{H}_{t-\delta})$ than outdoor

TABLE III: Model size and computational complexity of tested networks (CsiNet-LSTM, MarkovNet, MarkovNet-CNN) and comparable networks which do not exploit temporal correlation (CsiNet, CRNet, Deep AE). M: million, K: thousand.

| | **Parameters** | | | | | |
|---|---|---|---|---|---|---|
| | **CsiNet-LSTM** | **MarkovNet** | **MarkovNet-CNN** | **CsiNet** | **CRNet** | **Deep AE** |
| **CR=$\frac{1}{4}$** | 132.7 M | 2.1 M | 34.9 K | 2.1 M | 2.1 M | 3.2 M |
| **CR=$\frac{1}{8}$** | 123.2 M | 1.1 M | 27.8 K | 1.1 M | 1.1 M | 2.9 M |
| **CR=$\frac{1}{16}$** | 118.5 M | 0.5 M | 24.2 K | 0.5 M | 0.5 M | 2.8 M |
| **CR=$\frac{1}{32}$** | 116.1 M | 0.3 M | 22.4 K | 0.3 M | 0.3 M | 2.7 M |
| | **FLOPs** | | | | | |
| | **CsiNet-LSTM** | **MarkovNet** | **MarkovNet-CNN** | **CsiNet** | **CRNet** | **Deep AE** |
| **CR=$\frac{1}{4}$** | 412.9 M | 44.5 M | 41.2 M | 7.8 M | 7.7 M | 6.3 M |
| **CR=$\frac{1}{8}$** | 410.8 M | 42.4 M | 40.7 M | 5.7 M | 5.6 M | 5.8 M |
| **CR=$\frac{1}{16}$** | 409.8 M | 41.3 M | 40.5 M | 4.7 M | 4.5 M | 5.5 M |
| **CR=$\frac{1}{32}$** | 409.2 M | 40.8 M | 40.4 M | 4.1 M | 4.0 M | 5.4 M |

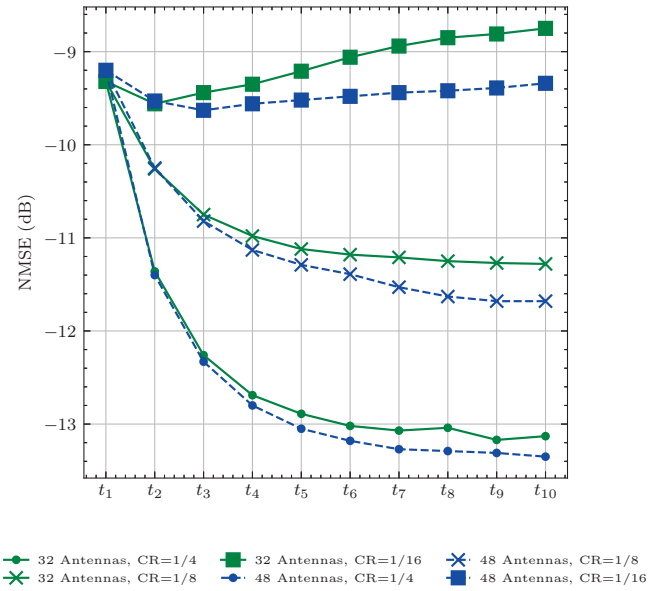



Fig. 12: Influence of antenna number on the CSI feedback performance of MarkovNet.

channels of higher mobility for the same $\delta$ (see Fig. 3). For this reason, MarkovNet demonstrates higher accuracy in CSI recovery for indoor channels than for outdoor channels. In the following example of Fig. 14, we can clearly see that for $\delta =$ 40ms MarkovNet generates substantially smaller NMSE than for larger $\delta = 80$ms. The results are consistent for various compression ratio of $\frac{1}{4}$, $\frac{1}{8}$, $\frac{1}{16}$, and $\frac{1}{32}$.

In future works, we expect that the conditional entropy can provide more valuable design guidelines when selecting the compression ratio and feedback rate for CSI estimation. Naturally, this requires a much more elaborate entropy encoding of the feedback coefficients by the UE.

## VII. Conclusion

To better exploit temporal channel coherence, we provide an information theoretic basis for efficient feedback in forward link CSI estimation. We propose MarkovNet, a CNN-based CSI feedback deep learning framework that leverages conditional entropy in differential encoding, which achieves superior estimation accuracy and lowers computational complexity relative to over-parameterized LSTM approach. We demonstrate that MarkovNet achieves accurate forward link CSI estimates despite a high degree of compression and quantization errors. MarkovNet achieves a substantial reduction in computation power and memory, making it a strong candidate for deployment on low cost mobile devices. In future work, we intend to explore more advanced estimation frameworks (e.g., Kalman filter, extended Kalman filter), different deep learning architectures, and entropy encoding to further reduce feedback payloads and enhance CSI recovery.

## Acknowledgment

The authors thank Prof. S. Jin of Southeast Univ. for kindly providing source codes of [19] and answering related questions in the process of preparing manuscript.

## References

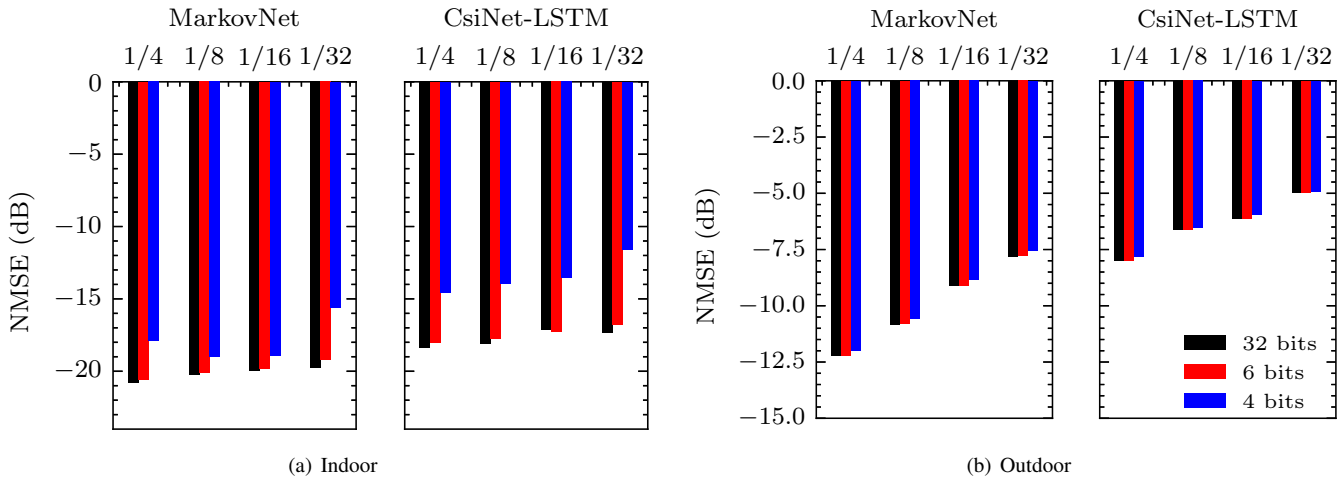
[1] E. P. Simon, L. Ros, H. Hijazi, J. Fang, D. P. Gaillot, and M. Berbineau, "Joint carrier frequency offset and fast time-varying channel estimation for MIMO-OFDM systems," *IEEE Trans. on Vehicular Tech.*, vol. 60, no. 3, pp. 955–965, 2011.

[2] W.-G. Song and J.-T. Lim, "Channel estimation and signal detection for MIMO-OFDM with time varying channels," *IEEE Commu. Letters*, vol. 10, no. 7, pp. 540–542, 2006.

[3] B. Makki and T. Eriksson, "On hybrid ARQ and quantized CSI feedback schemes in quasi-static fading channels," *IEEE Transactions on Communications*, vol. 60, no. 4, pp. 986–997, April 2012.

[4] D. J. Love, R. W. Heath, V. K. N. Lau, D. Gesbert, B. D. Rao, and M. Andrews, "An overview of limited feedback in wireless communication systems," *IEEE Journal on Sel. Areas in Comm.*, vol. 26, no. 8, pp. 1341–1365, 2008.

[5] H. Shirani-Mehr and G. Caire, "Channel state feedback schemes for multiuser MIMO-OFDM downlink," *IEEE Transactions on Communications*, vol. 57, no. 9, pp. 2713–2723, Sep. 2009.




(a) Indoor (b) Outdoor

Fig. 13: NMSE comparison of MarkovNet and CsiNet-LSTM for the (a) Indoor and (b) Outdoor scenarios with feedback subject to $\mu$-law quantization using fixed step size, $\Delta = 2^{b-1}$, for $b$ bits.

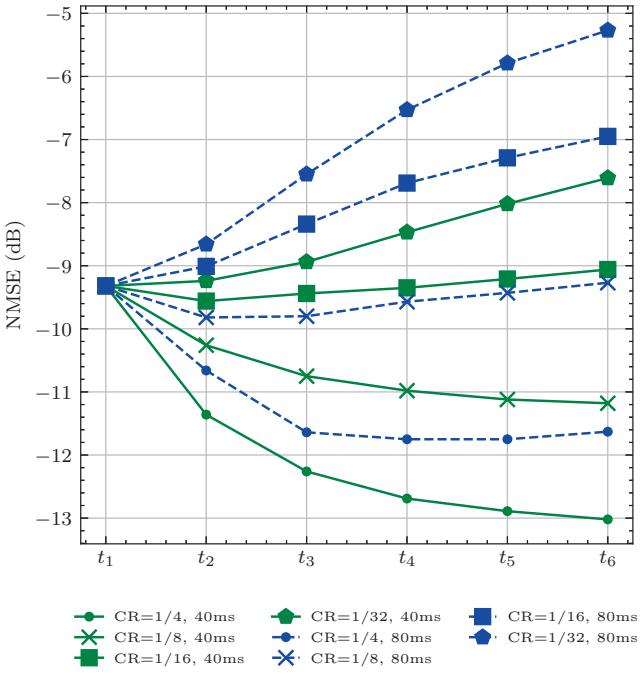



Fig. 14: Performance of MarkovNet on Outdoor channel for 40 ms (green solid lines) and 80 ms (blue dotted lines) feedback intervals for different compression ratios.


[6] A. Hindy, U. Mittal, and T. Brown, "CSI feedback overhead reduction for 5G Massive MIMO systems," in *10th Annual Computing and Communication Workshop and Conference (CCWC)*, 2020, pp. 0116–0120.

[7] X. Rao and V. K. N. Lau, "Distributed Compressive CSIT Estimation and Feedback for FDD Multi-User Massive MIMO Systems," *IEEE Tran. Signal Processing*, vol. 62, no. 12, pp. 3261–3271, June 2014.

[8] M. E. Eltayeb, T. Y. Al-Naffouri, and H. R. Bahrami, "Compressive Sensing for Feedback Reduction in MIMO Broadcast Channels," *IEEE Trans. Comm.*, vol. 62, no. 9, pp. 3209–3222, Sep. 2014.

[9] Z. Gao, L. Dai, S. Han, C. I, Z. Wang, and L. Hanzo, "Compressive sensing techniques for next-generation wireless communications," *IEEE Wireless Comm.*, vol. 25, no. 3, pp. 144–153, 2018.

[10] E. Chen, R. Tao, and X. Zhao, "Channel equalization for OFDM system based on the BP neural network," in *8th International Conference on Signal Processing*, vol. 3. IEEE, 2006.

[11] N. Taşpinar and M. N. Seyman, "Back propagation neural network approach for channel estimation in OFDM system," in *Proc. IEEE Int. Conf. on Wireless Comm., Networking and Info. Security*, 2010, pp. 265–268.

[12] C.-H. Cheng, Y.-P. Cheng *et al.*, "Using back propagation neural network for channel estimation and compensation in OFDM systems," in *7th. Int. Conf. on Complex, Intelligent, and Software Intensive Systems*. IEEE, 2013, pp. 340–345.

[13] K. Hiray and K. V. Babu, "A neural network based channel estimation scheme for OFDM system," in *2016 International Conference on Communication and Signal Processing (ICCSP)*. IEEE, 2016, pp. 0438–0441.

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

[15] 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, no. 3, pp. 889–892, 2019.

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

[17] 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 Comm. Letters*, vol. 9, no. 12, pp. 2192–2196, 2020.

[18] M. Hussien, K. K. Nguyen, and M. Cheriet, "PRVNet: Variational autoencoders for massive MIMO CSI feedback," *arXiv*, 2020.

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

[20] C. Lu, W. Xu, H. Shen, J. Zhu, and K. Wang, "MIMO Channel Information Feedback Using Deep Recurrent Network," *IEEE Commu. Letters*, vol. 23, no. 1, pp. 188–191, Jan 2019.

[21] Y. Liao, H. Yao, Y. Hua, and C. Li, "CSI Feedback Based on Deep Learning for Massive MIMO Systems," *IEEE Access*, vol. 7, pp. 86 810–86 820, 2019.

[22] X. Li and H. Wu, "Spatio-Temporal Representation With Deep Neural Recurrent Network in MIMO CSI Feedback," *IEEE Wireless Comm. Letters*, vol. 9, no. 5, pp. 653–657, 2020.

[23] Y. Jang, G. Kong, M. Jung, S. Choi, and I. Kim, "Deep Autoencoder Based CSI Feedback With Feedback Errors and Feedback Delay in FDD Massive MIMO Systems," *IEEE Wireless Comm. Letters*, vol. 8, no. 3, pp. 833–836, 2019.

[24] A. Milan, S. H. Rezatofighi, A. Dick, I. Reid, and K. Schindler, "Online multi-target tracking using recurrent neural networks," in *AAAI Conference on Artificial Intelligence*, 2017.

[25] I. Sutskever, O. Vinyals, and Q. V. Le, "Sequence to sequence learning with neural networks," *Advances in Neural Information Processing Systems*, vol. 4, no. January, pp. 3104–3112, 2014.

[26] O. Irsoy and C. Cardie, "Opinion mining with deep recurrent neural networks," *Proc. 2014 Conference on Empirical Methods in Natural Language Processing*, pp. 720–728, 2014.

[27] 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.

[28] R. H. Jr and A. Lozano, *Foundations of MIMO communication*. Cambridge University Press, 2018.

[29] M. Hermans and B. Schrauwen, "Training and analysing deep recurrent neural networks," in *Advances in Neural Information Processing Systems 26*, C. J. C. Burges and et al., Eds., 2013, pp. 190–198.

[30] R. Pascanu, C. Gulcehre, K. Cho, and Y. Bengio, "How to construct deep recurrent neural networks," *2nd International Conference on Learning Representations*, no. March 2014, 2014.

[31] Yoav Goldberg, "A Primer on Neural Network Models for Natural Language Processing," *Journal of Artificial Intelligence Research*, vol. 57, pp. 345–420, 2016. [Online]. Available: http://www.jair.org/papers/paper4992.html

[32] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, pp. 1735–80, 12 1997.

[33] Y. Bengio, "Learning deep architectures for AI," *Foundations and Trends in Machine Learning*, vol. 2, no. 1, pp. 1–27, 2009.

[34] D. Tse and P. Viswanath, *Fundamentals of Wireless Communication*. Cambridge university press, 2005.

[35] 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, December 2012.

[36] D. Lombardi and S. Pant, "Nonparametric $k$-nearest-neighbor entropy estimator," *Phys. Rev. E*, vol. 93, p. 013310, Jan 2016. [Online]. Available: https://link.aps.org/doi/10.1103/PhysRevE.93.013310

[37] T. S. Rappaport, *Wireless communications: principles and practice*. prentice hall PTR New Jersey, 1996, vol. 2.

[38] S. Dhanani and M. Parker, "Entropy, predictive coding and quantization," in *Digital Video Processing for Engineers*, S. Dhanani and M. Parker, Eds. Newnes, 2012, pp. 69 – 81.

[39] K. Huber and S. Haykin, "Improved Bayesian MIMO channel tracking for wireless communications: incorporating a dynamical model," *IEEE Trans. Wireless Comm.*, vol. 5, no. 9, pp. 2458–2466, 2006.

[40] A. Goldsmith, *Wireless communications*. Cambridge university press, 2005.

[41] S. Gao, P. Dong, Z. Pan, and G. Y. Li, "Deep Learning Based Channel Estimation for Massive MIMO With Mixed-Resolution ADCs," *IEEE Comm. Letters*, vol. 23, no. 11, pp. 1989–1993, Nov 2019.

[42] Y. LeCun, L. Bottou, G. Orr, and K. Müller, "Efficient backprop," in *Neural networks: Tricks of the trade*. Springer, 2012, pp. 9–48.

[43] P. Molchanov, S. Tyree, T. Karras, T. Aila, and J. Kautz, "Pruning convolutional neural networks for resource efficient inference," *arXiv preprint arXiv:1611.06440*, 2016.

[44] A. Nisar, J. A. Sue, and J. Teich, "Performance comparison between machine leamins based LTE downlink grant predictors," in *Proc. Int. Conf. on Artificial Intelligence (ICAI)*, 2019, pp. 226–232.

[45] M. Wang, B. Liu, and H. Foroosh, "Factorized convolutional neural networks," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV) Workshops*, Oct 2017.

[46] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, "Mobilenets: Efficient convolutional neural networks for mobile vision applications," *arXiv preprint arXiv:1704.04861*, 2017.


**Zhenyu Liu** received his Ph.D. degree from the Beijing University of Posts and Telecommunications in 2020. He is currently with the School of Computer Science (National Pilot Software Engineering School), Beijing University of Posts and Telecommunications as a Postdoctoral Fellow. From 2017 to 2019, he was a visiting Ph.D. student with the Department of Electrical and Computer Engineering at the University of California, Davis. His research interests include deep learning for the physical layer and massive MIMO.

**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 applying information theoretic techniques for learnable feedback quantization to reduce feedback bandwidth and in exploiting temporal coherence and downlink/uplink reciprocity to improve channel estimation accuracy.

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 is a Graduate Writing Fellow with UC Davis' University Writing Program, a Professors for the Future Fellow with UC Davis' GradPathways Institute for Professional Development, and 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.