

CQNet: Complex Input Quantized Neural Network designed for Massive MIMO CSI Feedback

Sijie Ji, Weiping Sun, *Member, IEEE*, Mo Li, *Fellow, IEEE*

Nanyang Technological University

sijie001@e.ntu.edu.sg, wpsun@ntu.edu.sg, limo@ntu.edu.sg

Abstract

The Massive Multiple Input Multiple Output (MIMO) system is a core technology of the next generation communication. With the growing complexity of CSI in massive MIMO system, traditional compressive sensing based CSI feedback has become a bottleneck problem that is limited in practical. Recently, numerous deep learning based CSI feedback approaches demonstrate the efficiency and potential. However, the existing methods lack a reasonable interpretation of the deep learning model and the accuracy of the model decreases significantly as the CSI compression rate increases.

In this paper, from the intrinsic properties of CSI data itself, we devised the corresponding deep learning building blocks to compose a novel neural network CQNet and experiment result shows CQNet outperform the state-of-the-art method with less computational overhead by achieving an average performance improvement of 8.07% in both outdoor and indoor scenarios. In addition, this paper also investigates the reasons for the decrease in model accuracy at large compression rates and proposes a strategy to embed a quantization layer to achieve effective compression, by which the original accuracy loss of 67.19% on average is reduced to 21.96% on average, and the compression rate is increased by 8 times on the original benchmark. Codes are available at github ¹.

Index Terms

Massive MIMO, FDD, CSI feedback, deep learning, complex neural network, attention mechanism, quantization neural network, edge computing.

¹Once the paper is accepted it will be made public

I. INTRODUCTION

The massive multiple-input multiple-output (MIMO) technology is considered one of the core technologies of the next generation communication system, e.g., 5G. By equipping large number of antennas, base station (BS) can sufficiently utilize spatial diversity to improve channel capacity. Especially, by enabling beamforming, a 5G BS can concentrate signal energy to a specific user equipment (UE) to achieve higher signal-to-noise ratio (SNR), less interference leakage and hence, higher channel capacity. However, beamforming is possibly conducted by the BS only when it has the channel state information (CSI) of the downlink at hand [1].

Many research efforts have been devoted to time-division duplexing (TDD) massive MIMO, because the CSI in the TDD mode can be obtained by exploiting channel reciprocity, where the pilot-aided training overhead is independent of the number of antennas.

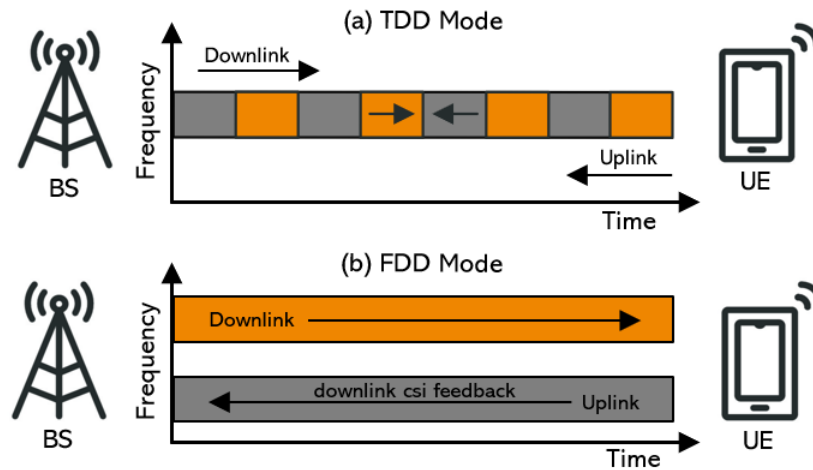


Fig. 1: (a) Time-Division Duplexing (TDD) mode: BS obtain CSI by channel reciprocity. (b) Frequency-Division Duplexing (FDD) mode: BS obtain CSI through the UE feedback transmission.

However, the TDD mode is not efficient enough in terms of time sensitive communication such as live video streaming, vehicular communications, etc. On the other hand, the frequency-division duplexing (FDD) mode uses different frequency bands for uplink and downlink transmissions at the same time, which is more efficient and can support more users simultaneously. Thus, most of contemporary cellular systems operate in FDD mode [2]. From the deployment perspective, adopting massive MIMO in FDD mode therefore attracts greater interest in exploring effective

approaches [3]. The biggest challenge to working with FDD mode is its overhead for CSI acquisition. Unlike TDD mode, in FDD mode, as Figure 1 depicts, the uplink and downlink channels are separated to different frequency bands, and hence the channel reciprocity does not exist. As a consequence, the UE would have to explicitly feed back the knowledge of downlink CSI to the BS, and the pilot-aided training overhead grows quadratically with the number of transmitting antennas which might overturn the benefit of Massive MIMO itself [4].

Fortunately, one important observation for massive MIMO systems helps alleviate the above issue. Some experimental studies of massive MIMO channels [5], [6] show that as the number of transmitting antennas increases, the user channel matrices tend to be sparse due to the limited local scatterers at the BS. Such observation inspires researchers to exploit unobvious sparse representation of CSI. Specifically, the massive MIMO channel has an *approximately* sparse representation in joint angular-delay domain [7], which can be obtained by conducting 2D-DFT on the channel matrix. The angle of arrival (AOA) and the spread delay of the path remain constant for uplink and downlink in the angular-delay domain. Based on the above characteristics, compressive sensing based CSI feedback algorithms have been proposed in recent years, e.g., LASSO [8], TVAL3 [9], and BM3D-AMP [10].

Compressive sensing based CSI feedback methods however relies heavily on channel sparsity and is limited by its efficiency in iteratively reconstructing the signals, the performance of which is highly dependent on the wireless channel, and thus is not a desirable approach considering the diversified use cases of 5G networks.

The recent rapid development of deep learning (DL) technologies provide another possible solution to efficiently feeding back CSI for FDD massive MIMO system. Instead of relying on sparsity, DL approaches utilize the auto-encoder frameworks [11] as an implicit prior constraint for encoding data [12]. The decoder learns a map from the low-dimensional data space to the targeted data distribution by single run to reconstruct the original data, without requiring the labeled data, which naturally overcomes the limit of compressive sensing based approaches in channel sparsity and operation efficiency. Recent studies [13], [14], [15], [16], [17] have demonstrated the feasibility and efficacy that DL can achieve in CSI feedback.

DL based CSI compression method is still at its early development. Most of previous works pay no attention to the complexity of the proposed DL model, which needs significant study. Considering the fact that the CSI compression would be conducted by UE, which has limited computing power and memory resources. Also, previous studies have not fully exploited the

characteristics of complex valued CSI for organic integration of the real and imaginary parts into the real valued neural network models, which limits their performance in accurately representing the wireless channel. Accordingly, in this work, we propose Complex Quantization Net (CQNet), a DL based neural network framework for massive MIMO CSI compression/decompression, which is empowered by forged complex-valued convolution layers and attention mechanisms. The proposed CQNet outperforms the state-of-the-art with higher accuracy in CSI feedback and less computational overhead in operation. At the same time, we propose an effective compression paradigm, which greatly improves the accuracy degradation of the current method in the case of large compression rate, and is able to improve the compression rate by a factor of 8 with the same accuracy. We state the following contributions.

- Signals and CSI are represented in complex envelopes, but at present, the majority of building blocks for DL models are based on real-valued operations and representations. We propose a way to extend existing real valued DL models to support complex valued CSI input, and maintain connections of CSI real and imaginary parts in the model.
- CSI corresponds to channel frequency response, which carries the physical information of the angle of arrival and the path delay can be clearly displayed in angular-delay domain with different resolution of cluster. Thus, we introduce attention mechanism to let the DL model learn information with weights rather than learn equally.
- CQNet is a lightweight network and reduces those operations that hardware-depends requires, such as exponent calculation.
- CQNet embeds quantization as a constraint of the neural network that mitigate quantization loss and achieve effective compression with higher accuracy in CSI compression.

The rest of this paper is organized as follows. Section II reviews related works. Section III introduces the system model and preliminary, including channel model and CSI feedback process. Section IV presents the detailed design of CQNet. Section V evaluates the performance of CQNet and provides experimental details. Section VI concludes the paper.

II. RELATED WORK

The challenge of CSI feedback in massive MIMO systems has motivated plenty of studies. Their main focus is to reduce feedback overhead by using the spatial and temporal correlation of CSI. Current established CSI feedback protocols are based on the concept of Compressive Sensing (CS). Specifically, recover the channel with a sparse vector from an undetermined linear

system [18]. However, CS based algorithms [8], [12] rely heavily on the assumption of channel sparsity and such algorithms [9], [10] need iterative construction, making the reconstruction process very slow.

To solve such limitations, recently, researchers leverage the deep learning technology that relaxes the sparsity assumption while learning the representative transform using data-driven approach. In particular, as Figure 2 illustrates, the designed encoder architecture at UE side to transform the angular-delay domain CSI to a compressed representation called code-words, and decoder architecture at BS side to reconstruct CSI from the code-words. Such architecture do not rely on any sparsity assumptions, instead, learn a non-linear data transform between original data distribution and latent space data distribution. Both side conduct transformation in a non-iterative way that is much faster than traditional compressive sensing based methods.

The first work CsiNet [16] explored and demonstrated the efficiency of deep learning based CSI feedback. They proposed CsiNet based on convolutional neural network (CNN) and carefully designed two sequential RefineNet units in decoder to refine the reconstruction accuracy. The results of CsiNet significantly outperform the traditional methods of CSI feedback (LASSO, BM3D-AMP and TVAL3) under various compression rates. Based on that, CsiNet-LSTM [15] leverage recurrent convolutional neural network (RCNN), combining several channels within coherence time T as a group as an input to the neural network to explore the temporal relationship between channels. CsiNet-LSTM shows the advantage of preserved accuracy under high compression ratio. However, the introduced LSTM increases the computational overhead. CsiNet+ [13] comprehensively surveyed recent deep learning based CSI feedback method and proposed a parallel multiple-rate compression framework focusing on practical storage issue. However, it requires manually switching based on the corresponding compression rate. The state-of-the-art method called CRNet [14] is based on the fact that the density of CSI matrix in angular-delay domain is highly dependent on the channel that they proposed CRBlock to flexibly extract the features in different resolutions. In the end, CRNet outperforms CsiNet under the same computational complexity.

Different from previous works, this work starts from exploring the inherent characteristic of CSI data, and take the practical issues, limited computation resource and limited storage at UE side as consideration to come up with a tailored lightweight DL framework, CQNet, for CSI feedback problem. In addition, this work investigate the potential quantization loss in DL based CSI feedback method and propose effective compression paradigm.

III. SYSTEM MODEL AND PRELIMINARY

A. Massive MIMO OFDM FDD System

Consider a single cell FDD system using massive MIMO with N_t antennas at BS, where $N_t \gg 1$ and N_r antennas at UE side. For simplicity, here we assume N_r equals to 1. The received signal $y \in \mathbb{C}^{N_c \times 1}$ can be expressed as

$$y = \mathbf{A}x + z \quad (1)$$

where N_c indicates the number of subcarriers, $x \in \mathbb{C}^{N_c \times 1}$ indicates the transmitted symbols, and $z \in \mathbb{C}^{N_c \times 1}$ is the complex additive Gaussian noise. \mathbf{A} can be expressed as $\text{diag}(h_1^H p_1, \dots, h_{N_c}^H p_{N_c})$, where $h_i \in \mathbb{C}^{N_t \times 1}$ and $p_i \in \mathbb{C}^{N_t \times 1}, i \in \{1, \dots, N_c\}$ represent downlink channel coefficients and beamforming precoding vector for subcarrier i , respectively. $(\cdot)^H$ here represents conjugate transpose.

In order to derive the beamforming precoding vector p_i , the BS needs the knowledge of corresponding channel coefficient h_i , which is fed back by the UE. Suppose that the downlink channel matrix is $\mathbf{H} = [h_1 \dots h_{N_c}]^H$ which contains $N_c N_t$ elements. The number of parameters that need feed back is $2N_c N_t$, including the real and imaginary parts of the CSI. Note that the amount of feedback parameters is proportional to the number of antennas, meaning in massive MIMO, the extremely large number of antennas will give rise to excessive size of the feedback channel matrix \mathbf{H} .

The channel matrix \mathbf{H} is often sparse in the angular-delay domain. By 2D discrete Fourier transform (DFT), the original form of spatial-frequency domain CSI can be converted into angular-delay domain, such that

$$\mathbf{H}' = \mathbf{F}_c \mathbf{H} \mathbf{F}_t^H \quad (2)$$

where \mathbf{F}_c and \mathbf{F}_t are the DFT matrices with dimension $N_c \times N_c$ and $N_t \times N_t$, respectively. For angular-delay domain channel matrix \mathbf{H}' , every element in \mathbf{H}' corresponds to a certain path delay with a certain angle of arrival (AoA). In \mathbf{H}' , only the first N_a rows contain useful information, while the rest of rows, which represent the paths with larger propagation delays, are made up of near-zero values, can be omitted without much information loss. Let \mathbf{H}_a denote the informative rows of \mathbf{H}' . Although \mathbf{H}_a is already smaller than original CSI matrix, $2N_a N_t$ may still remain large. While \mathbf{H}_a might be sparse enough for the compressive sensing based methods when $N_t \rightarrow \infty$. In practice N_t is limited, thus leading to the sparsity assumption invalid, especially when large compression ratio η is applied.

In this paper, we design CQNet by adopting encoder-decoder network. The channel matrix \mathbf{H} is first converted into angular-delay representation \mathbf{H}' by 2D-DFT. We then remove the near-zero components to obtain \mathbf{H}_a . The encoder of CQNet at UE side compresses \mathbf{H}_a into a codeword vector \mathbf{v} according to a given compression ratio η . \mathbf{v} is then fed back to the BS, which will reconstruct \mathbf{H}_a based on \mathbf{v} using its decoder. \mathbf{H}_a is finally zero filled and reverted to original \mathbf{H} by inverse 2D-DFT.

B. CSI Feedback Process

\mathbf{H}_a is put into UE's encoder to produce codeword \mathbf{v} such that

$$\mathbf{v} = f_{\mathcal{E}}(\mathbf{H}_a, \theta_{\mathcal{E}}) \quad (3)$$

where $f_{\mathcal{E}}$ denotes the encoding process and $\theta_{\mathcal{E}}$ represents a set of parameters of the encoder.

Once the BS receives the codeword \mathbf{v} , the decoder is used to reconstruct the channel by

$$\hat{\mathbf{H}}_a = f_{\mathcal{D}}(\mathbf{v}, \theta_{\mathcal{D}}) \quad (4)$$

where $f_{\mathcal{D}}$ denotes the decoding process and $\theta_{\mathcal{D}}$ represents a set of parameters of the decoder.

Therefore, the entire feedback process can be expressed as

$$\hat{\mathbf{H}}_a = f_{\mathcal{D}}(f_{\mathcal{E}}(\mathbf{H}_a, \theta_{\mathcal{E}}), \theta_{\mathcal{D}}) \quad (5)$$

The goal of CQNet is to minimize the difference between the original \mathbf{H}_a and the reconstructed $\hat{\mathbf{H}}_a$, which can be expressed formally as finding the parameter sets of encoder and decoder satisfying

$$(\hat{\theta}_{\mathcal{E}}, \hat{\theta}_{\mathcal{D}}) = \arg \min_{\theta_{\mathcal{E}}, \theta_{\mathcal{D}}} \|\mathbf{H}_a - f_{\mathcal{D}}(f_{\mathcal{E}}(\mathbf{H}_a, \theta_{\mathcal{E}}), \theta_{\mathcal{D}})\|_2^2 \quad (6)$$

IV. CQNET DESIGN

In this section, we present the design of CQNet and its key components. Figure 2 depicts the overall architecture of CQNet. CQNet is an encoder-decoder deep learning framework which contains four main building blocks tailored to the CSI feedback problem.

CQNet employs a forged complex-valued input layer that takes real and imaginary parts of the CSI and performs multiple filtered 1×1 convolutions to separately represent the full complex-valued channel coefficients of different signal paths (Section IV-A). Following the two different types of attention blocks are applied to devise an informative and lightweight encoder, i.e., the channel-wise attention block which aims at enhancing the effectiveness of complex-valued

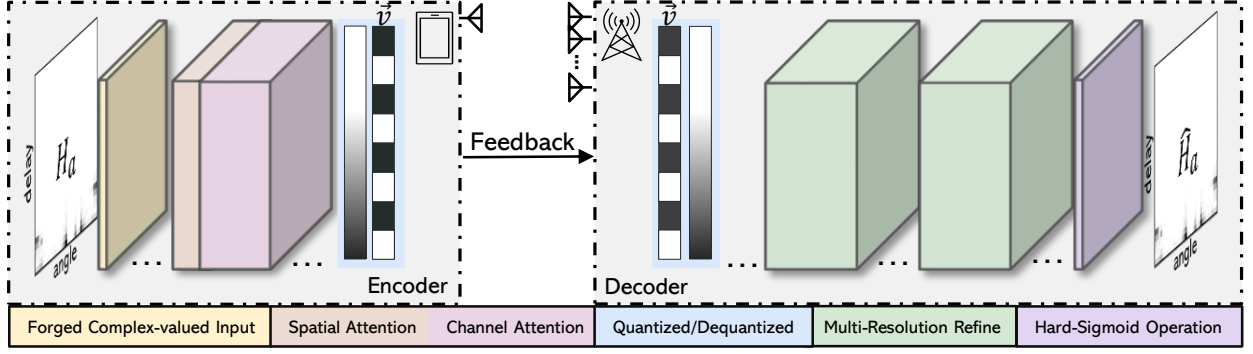


Fig. 2: The encoder and decoder architecture of CQNet. The encoder compresses CSI into a codeword vector \mathbf{v} according to a given compression ratio η . The decoder reconstructs CSI based on received feedback \mathbf{v} .

input layer, and the spatial-wise attention block which aims at making use of the cluster effect in the angular-delay domain (Section IV-B). CQNet keeps the residual refine block and multi-resolution block in the decoder side as previous studies shows their effectiveness [14], [16]. To further reduce computation cost, CQNet modifies the fully connected layer which were adopted in most previous studies to point-wise convolution layer and adopts hard-Sigmoid activation which is more hardware friendly than conventional Sigmoid activation (Section IV-C). In addition, CQNet embeds quantization as layers in neural networks which serves as additional regularization constraints to mitigate quantization loss and improve accuracy under large compression rate in the compression process (Section IV-D). Appendix A, Figure 9 presents the complete CQNet architecture with detailed layer level design for reproducibility.

A. Forged Complex-valued Input

While a typical deep learning neural network is designed based on real-valued inputs, operations, and representations, the input of our problem is based on complex-valued path channel coefficients in \mathbf{H}' . How to best cope with complex-valued inputs is yet an open question in machine learning community [19]. Most existing studies utilizing deep learning for wireless communication or wireless sensing systems separate the real and imaginary parts of the complex-valued signals, take them as two independent channels of an image as input, and perform mixed convolutions around the real and imaginary parts of different values. In such a way, the real and

imaginary parts of the same complex value are decoupled during the convolution process, which may destroy the original characteristics of each complex-valued channel coefficient.

To tackle such an issue, CQNet devises a specific input layer, which utilizes 1×1 point-wise convolution to couple the real and imagery parts of the same channel coefficient. The forged complex-valued input layer employs multiple 1×1 convolutional filters to encode the real and imaginary parts of each complex-valued element in \mathbf{H}_a with respective learnable weights.

Mathematically, $\mathbf{F}_{tr} : \mathbf{H}_a \rightarrow \mathcal{I}$ is a convolutional transformation. Here, $\mathbf{H}_a \in \mathbb{R}^{N_a \times N_a \times 2}$ is a 3D tensor, extended from its 2D version by including an additional dimension to separately express the real and imaginary parts, and $\mathcal{I} \in \mathbb{R}^{N_a \times N_a \times C}$, where C indicates the number of convolutional filters applied to learn different weighted representations. Let $\mathbf{F} = [\mathbf{f}_1, \mathbf{f}_2, \dots, \mathbf{f}_C]$ denote the learned set of filter kernels, where \mathbf{f}_c refers to the learnable parameter of the c -th filter. The output of \mathbf{F}_{tr} is $\mathcal{I} = [\mathbf{i}_1, \mathbf{i}_2, \dots, \mathbf{i}_C]$, $\mathbf{i}_c \in \mathbb{R}^{N_a \times N_a}$, where

$$\mathbf{i}_c[m, n] = \mathbf{f}_c * \mathbf{H}_a = \sum_{i=1}^m \sum_{j=1}^n \sum_{k=1}^2 \mathbf{f}_c[i, j] \mathbf{H}_a^k[m - i, n - j] \quad (7)$$

Here $*$ denotes convolution, and $m, n \in [0, N_a)$. For simplicity, bias terms are omitted. Since the output is produced by a summation of the two channels, the dependency between real and imaginary parts is implicitly embedded in \mathbf{i}_c . Based on the trade-off between accuracy and model size, CQNet adopts $C = 32$ learnable filters. To compare, conventional 3×3 kernel size entangles the real and imaginary parts of neighboring elements in \mathbf{H}_a , and as a result the 9 complex values are interpolated as one synthesized value, Figure 3 (b), thus losing the original physical information carrier by the channel matrix. Figure 3 (a) illustrates the design of the complex-valued input layer, \mathbf{F}_{tr} , the output of which will be directed to the attention mechanism \mathbf{F}_{se} (to be detailed in next section).

B. Attention Mechanism for Informative Encoder

The performance of CSI feedback scheme highly depends on the compression part, the encoder. Due to the limited computing power and storage space of UE, deepening the encoder network design is not practical. Therefore, CQNet adopts attention mechanism to achieve distilled yet informative encoding output.

Attention mechanism assists the neural network to focus on important features and suppress unnecessary ones by assigning different learnable weights. It can be interpreted as a means of biasing the allocation of available computational resources to the most informative components

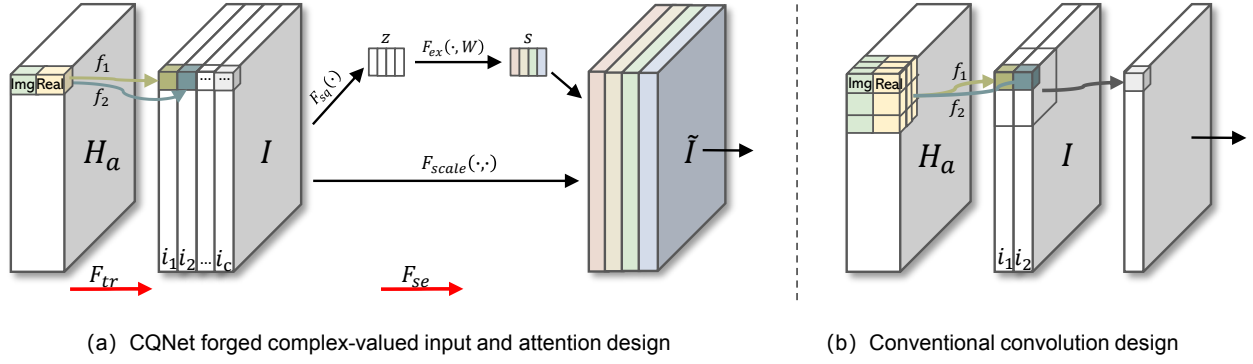


Fig. 3: (a) The forged complex-valued input operation F_{tr} couples the real and imaginary parts of the complex-valued channel matrix and after that **the channel-wise attention** F_{se} is applied to strengthen learning of significant channel coefficients. (b) Conventional convolution entangles the real and imaginary parts with neighboring elements' real parts and imaginary parts.

of a signal that increases the representativeness of the neural network. CQNet imposes two different attention mechanisms for different stages, resulting in a lightweight yet informative encoder.

1) *Channel-Wise Attention*: To stay with the complex annotated H_a , CQNet devises a forged complex-valued input layer. The output of the input layer \mathcal{I} , essentially, is a weighted representation of original H_a . Specifically, $\mathcal{I} \in \mathbb{R}^{N_a \times N_a \times C}$, where the number of channel C corresponds to the learned different weights of H_a , among which, some may be more important than others. Based on this, CQNet introduces the channel-wise attention mechanism, SE block [20], to assist the neural network model with the relationship of the weights so as to focus on important features and suppress unnecessary ones. A diagram of SE block is shown in Figure 3 (a) with annotation F_{se} .

The output \mathcal{I} first goes through F_{sq} transformation by global average pooling to obtain channel-wise statistics descriptor $z \in \mathbb{R}^C$,

$$z_c = F_{sq}(i_c) = \frac{1}{N_a \times N_a} \sum_{i=1}^{N_a} \sum_{j=1}^{N_a} i_c(i, j), \text{ s.t. } c \in \{1, 2, \dots, C\} \quad (8)$$

Here, F_{sq} acts as expanding network receptive field to the whole angular-delay domain to obtain global statistical information, compensating for the shortcoming of insufficient local receptive field of 1×1 convolution.

After that, the channel descriptor \mathbf{z} goes through \mathbf{F}_{ex} transformation, i.e., a gated layer with sigmoid activation to learn the nonlinear interaction as well as non-mutually-exclusive relationship between channels, such that

$$\mathbf{s} = \mathbf{F}_{ex}(\mathbf{z}, \mathbf{W}) = \sigma(g(\mathbf{z}, \mathbf{W})) = \sigma(\mathbf{W}_2 \delta(\mathbf{W}_1 \mathbf{z})), \quad (9)$$

where δ is the ReLU function, $\mathbf{W}_1 \in \mathbb{R}^{\frac{C}{2} \times C}$ and $\mathbf{W}_2 \in \mathbb{R}^{C \times \frac{C}{2}}$. \mathbf{F}_{ex} further explicitly model inter-channel dependencies based on \mathbf{z} and obtain calibrated \mathbf{s} , which is the attention vector that summarizes all the characteristics of channel C , including intra-channel and inter-channel dependencies. Before being fed into the next layer, each channel of \mathcal{I} is scaled by the corresponding attention value, such that

$$\tilde{\mathcal{I}}_{:,i} = \mathbf{F}_{scale}(\mathbf{s}, \mathcal{I}) = \mathbf{s}_i \mathcal{I}_{:,i}, \quad \text{s.t. } i \in \{1, 2, \dots, C\} \quad (10)$$

Channel-wise attention mechanism intrinsically captures dynamics based on the complex-valued input \mathbf{H}_a by learning to weigh the importance of each channel in \mathcal{I} , boost the feature discriminability, and generates more informative $\tilde{\mathcal{I}}$.

2) *Spatial-Wise Attention*: Spatial-wise attention focuses on learning the places of the more informative parts across spatial domain. Specifically, after being converted to angular-delay domain, the channel coefficients exhibit effect of clusters corresponding to the distinguishable paths that arrive with specific delays and AoAs. In order to pay more attention to those clusters, CQNet employs a CBAM block [21] to learn differentiation with weighting in the spatial domain as Figure 4 illustrates.

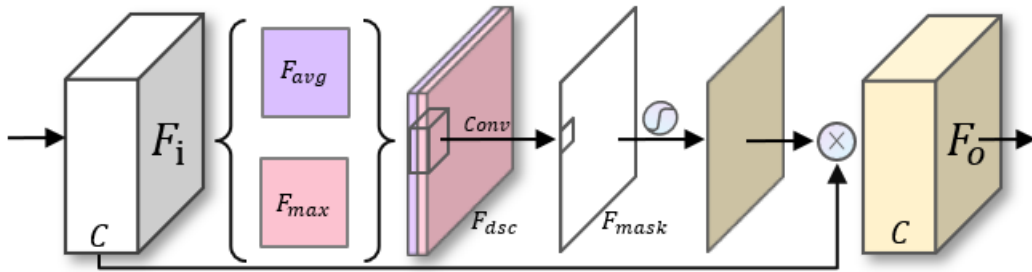


Fig. 4: Based on the cluster effect in angular-delay domain, **spatial-wise attention** uses the generating spatial statistical descriptors as the basis for assigning weights, forcing the network to focus more on the distinguishable propagation paths.

First, two pooling operations, i.e., average-pooling and max-pooling, are adopted across the input \mathbf{F}_i 's channel C to generate two 2D feature maps, $\mathbf{F}_{\text{avg}} \in \mathbb{R}^{N_a \times N_a \times 1}$ and $\mathbf{F}_{\text{max}} \in \mathbb{R}^{N_a \times N_a \times 1}$, respectively. CQNet concatenates the two feature maps to generate a compressed spatial feature descriptor $\mathbf{F}_{\text{dsc}} \in \mathbb{R}^{N_a \times N_a \times 2}$, and convolves it with a standard convolution layer to produce a 2D spatial attention mask $\mathbf{F}_{\text{mask}} \in \mathbb{R}^{N_a \times N_a \times 1}$. The mask is activated by Sigmoid and then multiplied with the original feature maps \mathbf{F}_i to obtain \mathbf{F}_o with spatial-wise attention.

$$\begin{aligned}\mathbf{F}_o &= \text{CBAM}(\mathbf{F}_i) \\ &= \mathbf{F}_i (\sigma(\mathbf{f}_c([\text{AvgPool}(\mathbf{F}_i); \text{MaxPool}(\mathbf{F}_i)]))) \\ &= \mathbf{F}_i (\sigma(\mathbf{f}_c([\mathbf{F}_{\text{avg}}; \mathbf{F}_{\text{max}}])))\end{aligned}\tag{11}$$

With spatial-wise attention, CQNet focuses the neural network to the more informative signal propagation paths in the angular-delay domain.

C. Reduction of the Computation Cost

In practice, UEs are often edge devices with limited computational power, memory and storage, which must be taken into consideration in CQNet design. This section details our efforts in reducing its space and time cost.

1) *Space Cost*: Since the final objective of CQNet is to compress CSI into a fixed length vector \mathbf{v} with compression ratio η , the last layer of encoder is a fully connected layer. The operation of 1×1 convolution is equal to that of fully connected layer, since both of them entail element-wise multiplication. CQNet replaces fully connected layer with 1×1 convolution layer, which greatly reduces the parameters of the network. Taking the input of $\mathbf{H}_a \in \mathbb{R}^{32 \times 32 \times 2}$ which equals to 2048 dimensions, and $\eta=1/4$ as an example, the number of the parameters of the fully connected layer is $32 \times 32 \times 2 \times 512$, while that of 1×1 convolution layer is $32 \times 32 \times 2 \times 1$, 512 times fewer.

2) *Time Cost*: Sigmoid activation function as often used contains exponential operation

$$\sigma(x) = \frac{1}{1 + e^{-x}} = \frac{e^x}{e^x + 1}.\tag{12}$$

In order to reduce time cost in the computation, CQNet uses hard version of Sigmoid, its piece-wise linear analogy function, denoted as $h\sigma$ to replace Sigmoid function [22], [23],

$$h\sigma(x) = \frac{\text{ReLU } 6(x + 3)}{6},\tag{13}$$

where ReLU6 is a clip version of ReLU, which ensures quantization precision in float16 edge device

$$\text{ReLU6}(x) = \min(\max(x, 0), 6). \quad (14)$$

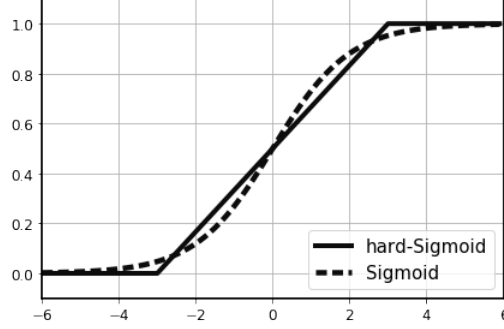


Fig. 5: Comparison between Sigmoid and hard-Sigmoid functions.

Figure 5 compares the excitation curves of the hard-Sigmoid and Sigmoid functions. The hard-Sigmoid induces no discernible degradation in accuracy but benefits from its computational advantage of entailing no exponential calculations. In practice, hard-Sigmoid can fit in most software and hardware frameworks and can mitigate potential numerical quantization loss introduced by different hardware.

D. Effective Quantization in the Neural Network

Unlike commonly used encoder-decoder framework, where the encoder output is fed directly into the decoder to reconstruct the input, in our problem of CSI feedback, the encoder output at UE side needs to be transferred to the BS as a bit-stream through a real communication channel. The output of DL encoder commonly is a 32 bit float-point representation providing us an opportunity to perform further compression, which has not been studied in previous DL-based CSI feedback studies. For instance, representing a 32bit parameter with 4bit quantized number gives a true compression ratio of $\eta * 4/32$ which we call *effective compression ratio* γ .

Direct quantization, however, leads to significant quantization loss, as we will show in our experimental evaluation. Given $\mathbf{H}_a \in \mathbb{R}^{32 \times 32 \times 2}$ which equals to 2048 dimensions, and $\eta=1/4$ we let the output of CRNet [14] encoder \mathbf{v} , with the value range (l, u) , be quantized uniformly by

bit width $\mathcal{B} = 4$, namely, 512 length 32 bit float-point codeword \mathbf{v} is transferred to 2048 0-1 bit-stream for transmission by

$$Q_U(\mathbf{v}) = \text{round}\left(\frac{\mathbf{v}}{\Delta}\right) \Delta, \quad (15)$$

where the range (l, u) is divided into $2^{\mathcal{B}} - 1$ interval $\mathcal{P}_i, i \in (0, 1, \dots, 2^{\mathcal{B}} - 1)$, and $\Delta = \frac{u-l}{2^{\mathcal{B}}-1}$ denotes the interval length.

The converted 2048 bits are dequantized back to 32 length float-point and fed into the decoder. Following the operation, the average NMSE result drops dramatically from -26.64 to -5.99 , which results in more than 4 times performance drop.

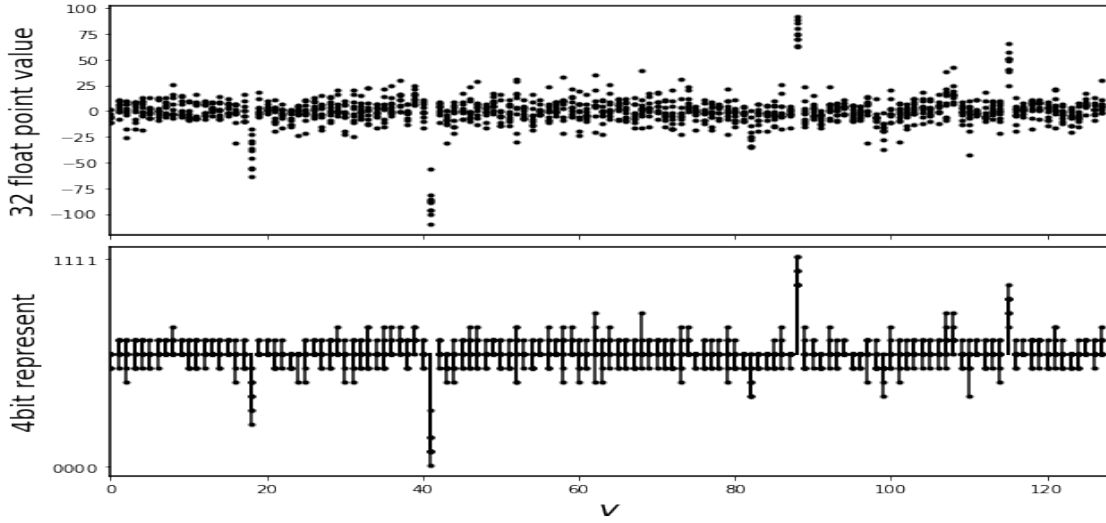


Fig. 6: An example with 4bit quantization: output of an encoder with $\eta = 1/16$

Figure 6 visualizes the quantization process of a batch, 10 codeword of length 128. The upper shows the original output value in 32-bit floating point form, and the bottom shows the corresponding value quantized into 4-bit representation.

To mitigate the quantization loss, CQNet embeds quantization-dequantization process as layers that can be trained together with the whole neural network. Since the quantization operation is not derivable, we set the gradient of the layer to be a constant. Essentially, the layer becomes a regularization term that forces the network to adjust the data distribution according to the quantization method and thus reduce the quantization loss.

Embedding the quantization layer in the deep neural network also offers a room for adaptive quantization. We can either fix the bit width \mathcal{B} as hyper-parameters or set it as a learnable

parameter so the quantization layer can adaptively choose the bit width \mathcal{B} to represent a float number.

V. EVALUATION

In this section, we evaluate the overall performance of CQNet and the efficacy of the key components. The detailed experiment setting is described in section V-A. Section V-B presents the overall performance and computational overhead as compared with state-of-the-art machine learning based CSI feedback approaches. We then conduct ablation study by additively evaluate the forged complex-valued input layer and two attention blocks to assess their efficacy (Section V-C). Finally, we analyze the effect of the new quantization layer and discuss the possibility of adaptive compression in Section V-D.

A. Experiment Setting

1) *Data Generation*: To ensure a fair performance comparison, we use the same dataset as provided in the first work of deep learning based Massive MIMO CSI feedback in [16], which is also used in later studies on this problem [13], [14], [15]. The channel coefficients are generated by COST 2100 channel model [24] with configuration of $N_t = 32$ uniform linear array (ULA) antennas at the BS, $N_r = 1$ antenna at UE and $N_c = 1024$ sub-carriers. There are two types of scenarios. The first one is **indoor pico-cell scenario** operating on 5.3 GHz band. BS is positioned at the center of 20m square area and UEs are randomly positioned within that square. The other is **outdoor rural scenario** operating on 300 MHz band. BS is positioned at the center of a 400m square area and UEs are randomly positioned within that square. The generated CSI matrices are converted to angular-delay domain $\mathbf{H}_a \in \mathbb{R}^{32 \times 32 \times 2}$ by 2D-DFT.

The total 150,000 independently generated CSI are split into three parts, i.e., 100,000 for training, 30,000 for validation, and 20,000 for testing, respectively.

2) *Training Scheme and Evaluation Metric*: As comparison scheme, we use the start-of-the-art method CRNet [14], which significantly outperforms other CSI feedback work. CRNet demonstrates the effectiveness of using cosine annealing learning rate with warming up scheme instead of fixing the learning rate to train, and hence, in CQNet, we adopt the same training scheme. To evaluate the performance, we measure the normalized mean square error (NMSE) between the original \mathbf{H}_a and the reconstructed $\hat{\mathbf{H}}_a$:

$$\text{NMSE} = \text{E} \left\{ \|\mathbf{H}_a - \hat{\mathbf{H}}_a\|_2^2 / \|\mathbf{H}_a\|_2^2 \right\} \quad (16)$$

The model was trained with the batch size of 200 and 8 workers on a single NVIDIA 2080Ti GPU. The epoch is set to 1000, as recommended in previous work [14], [13]. To further ensure fairness, we fix the random seed of the computer in every run.

B. CQNet Overall Performance

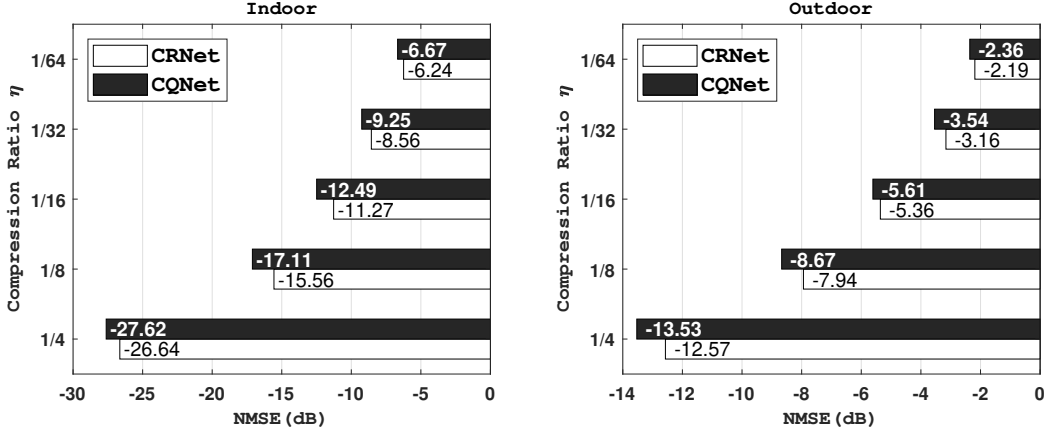


Fig. 7: Normalized Mean Square Error(dB) Comparison between CRNet and CQNet.

Figure 7 shows the overall performance of CQNet as compared with CRNet [14], with the same hardware condition and training scheme. In indoor scenarios, CQNet obtains an average performance increase of 7.88%, with the most significant increase of 10.83% at the compression ratio of $\eta = 1/16$. In outdoor scenarios, the average improvement on NMSE is 8.26%, the most significant increase occurs at the compression ratio of $\eta = 1/8$ with increase of 12.03%. Even in the worst case, CQNet achieves 3.68% ($\eta = 1/64$) and 4.66% ($\eta = 1/16$) improvement in indoor and outdoor scenarios, respectively. The result shows that CQNet consistently outperforms CRNet for all compression ratios in both indoor and outdoor scenarios with 8.07% overall average improvement on NMSE. In addition, we notice that both CQNet and CRNet have lower accuracy in outdoor scenarios, which is probably caused by the data processing. Due to the long propagation distance, the propagation loss and the path delay in outdoor scenarios are larger, leading to part of the information being discarded when calculating \mathbf{H}_a .

At the same time, we also derive the computational cost in flops (floating-point operations per second) of the two models. As Figure 8 indicates, the number of flops of CQNet is 1.6%, 5.0%, 10.2%, 17.6%, 26.6% less than CRNet at compression ratio η of 1/64, 1/32, 1/16, 1/8, 1/4, respectively, which indicate that CQNet yields higher accuracy with less computational

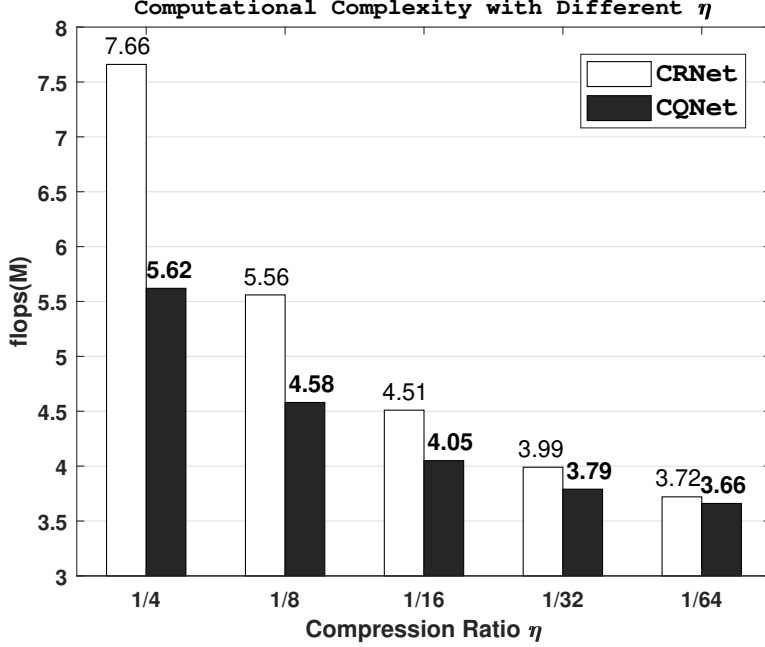


Fig. 8: the Number of flops ($\times 10^6$) of CRNet and CQNet.

complexity. The reduction of the flops is negatively correlated with the compression rate, where the lower compression rate there is, the larger flops reduction there will be. This gain comes from the design of replacing the fully connected layer, whose hyper parameters are proportional to the length of code-word \mathbf{v} .

C. Ablation Study

Considering the limited interpretability of deep neural network, we further conduct the ablation study to better quantify the gain of the proposed forged complex-valued input and attention mechanism. The epochs of ablation studies are set to 500, the rest settings remain the same as discussed in Section V-A.

1) Forged Complex-valued Input Design: To evaluate the forged complex-valued input design, we conduct an ablation study that adds the forged complex-valued input layer to the baseline CRNet without other modifications to it. The result is shown in Table I column B. With the forged complex-valued input layer, the accuracy surpasses baseline CRNet (Table I column A) at all compression ratios with an average improvement of 9.2%, which demonstrates the efficacy of appropriately interpret the complex notation.

TABLE I: NMSE (dB) Comparison of Ablation Study

	A	B	C	D	E
Baseline	✓	✓	✓	✓	✓
Complex-valued Input		✓	✓	✓	✓
Channel-wise Attention			✓		✓
Spatial-wise Attention				✓	✓
$\eta=1/4$	-21.912	-26.880	-26.146	-22.322	-27.212
$\eta=1/8$	-14.048	-15.317	-15.565	-15.164	-15.845
$\eta=1/16$	-10.216	-11.347	-11.130	-10.651	-11.277
$\eta=1/32$	-8.484	-8.744	-8.945	-8.763	-8.974
$\eta=1/64$	-6.063	-6.072	-6.045	-6.064	-6.438

2) *Attention Mechanism Design*: In addition to the complex-valued input design, we further conduct the ablation study with three groups of experiments, i.e., only adopting channel-wise attention (Table I column C), only adopting spatial-wise attention (Table I column D) and adopting both attention mechanisms (Table I column E). Compared to the baseline, either adopting channel-wise or spatial-wise attention can improve the performance, where the accuracy is 8.8% and 3.5% higher for the channel-wise and spatial-wise attention, respectively. Adopting the two attention mechanisms at the same time gives the best performance gain, 11.9%.

However, we notice that the performance of the added attention mechanism shows uncertainties. In some cases, the accuracy may drop compared to that Table I column B with forged complex-valued input layer only. Nevertheless, if adopting two attention mechanisms simultaneously, the overall performance is still 2.5% better than Table I column B that only adding the complex-valued input layer. This is the reason why we finally adopt both attention mechanisms in the CQNet design. In particular, when the compression rate is large, for example, $\eta = 1/64$, adopting both attention mechanisms helps to improve the NMSE(dB) from -6.063 to -6.438 with an improvement of 6.2%, while adding complex-valued input layer improves alone only 0.1%.

TABLE II: NMSE(dB) Result of Quantization Loss & Embedding Quantization Gain.

η	1/4		1/8		1/16		1/32		1/64	
wo/q	-26.64		-15.56		-11.27		-8.56		-6.24	
B	4	8	4	8	4	8	4	8	4	8
γ	1/32	1/16	1/64	1/32	1/128	1/64	1/256	1/128	1/512	1/256
w /q	-5.99	-25.60	-4.72	-15.45	-2.95	-11.20	-6.08	-8.56	-4.75	-6.24
e /q	-20.16	-26.08	-14.36	-15.49	-10.78	-11.19	-8.47	-8.56	-5.87	-6.24

(a) CRNet

η	1/4		1/8		1/16		1/32		1/64	
wo/q	-27.62		-17.11		-12.49		-9.25		-6.67	
B	4	8	4	8	4	8	4	8	4	8
γ	1/32	1/16	1/64	1/32	1/128	1/64	1/256	1/128	1/512	1/256
w /q	-6.43	-25.80	-11.60	-17.07	-7.04	-12.45	-5.36	-9.23	-5.86	-6.63
e /q	-22.21	-26.78	-15.87	-17.48	-11.02	-11.50	-8.89	-8.97	-6.02	-6.63

(b) CQNet

D. Effective Quantization

To evaluate the effect of the add-on quantization layer and the performance with effective compression ratio γ , we conduct experiments under indoor scenarios with different quantization ratio. We use trained CRNet and CQNet with different compression ratio η and let the encoder outputs go through quantization with corresponding bit width B and fed them back to the decoder after dequantization. We follow this procedure for both CRNet and CQNet, and results are shown in Table II denoted as 'wo/q' and 'w/q', for compression without and with add-on quantization layer respectively. We report results for $B = 4$ and 8 respectively. When $B = 16$ the quantization loss is very small so we do not include those results. As the results demonstrate, both CRNet and CQNet have different levels of quantization loss at different compression rates. Among them, the loss is most significant when the $\eta = 1/4$ and $B = 4$. Both CRNet and CQNet suffer from more than 4 times accuracy drops. When the compression rate itself is high, the quantization loss is relatively small, for example, $\eta = 1/64$ and $B = 4$ CRNet and CQNet have accuracy decrease with 23.9% and 12.1% correspondingly. It means that the accuracy loss of the DL model itself is dominant, therefore, by reducing the quantization loss, we can choose the model with less accuracy loss for quantization and thus achieve better results at the same compression rate.

As mentioned in Section IV-D, we embed quantization as a layer of the neural network and set the gradient as constant to regularize the neural network. Results are shown in Table II with denoted 'e/q'. Compared to direct quantization (w/q), the results of the embedded quantization layer improve the accuracy at all η and all \mathcal{B} , with a significant improvement especially for $\mathcal{B} = 4$. When η at the value of $1/4, 1/8, 1/16$ and $1/32$, the improvement of CRNet and CQNet are 236.6% and 245.4%, 204.02% and 36.8%, 265.4% and 56.5%, 39.3% and 65.9%, respectively. The results fully demonstrate the effectiveness of the embed quantization layer and the performance gain orthogonal to neural network architecture.

This design gives us a new way of effective compression that compensates for the reduced accuracy of various current schemes at large compression rates. The best results showing the corresponding effective compression ratio γ based on such a quantization design are bolded in Table II. It can be seen that even compared to the idea case without quantization loss, this effective compression approach can improve the accuracy of both CRNet and CQNet by 131.4% and 114.4%, 135.5% and 140.1%, 130.1% and 137.9% for the original compression ratios of $1/16, 1/32$, and $1/64$, respectively. In addition, we achieve similar accuracy in extremely case with $\gamma = 1/512$ as before $\eta = 1/64$. Pushing the limit of compression ratio from $1/64$ to $1/512$, an 8x improvement.

VI. CONCLUSION

In this paper, we study CSI feedback problem for 5G communication systems, which is supposed to be the bottleneck in massive MIMO operation. With consideration of the physical properties of the CSI data itself, we propose a novel deep learning framework, CQNet, which is based on the attention mechanism with pseudo-complex input. The overall performance of CQNet is superior as compared with the state-of-the-art CRNet with less computation overhead. In addition, we investigate a practical issue, the quantization loss faced in real communication systems, and identify that integrating a quantization layer into the neural network may serve as a constraint to reduce quantization loss. With our proposed effective compression paradigm, we can improve the previous problem of large compression rate accuracy reduction and can further increase the compression rate.

APPENDIX A

DETAILED LAYER LEVEL ARCHITECTURE DIAGRAM FOR REPRODUCIBILITY

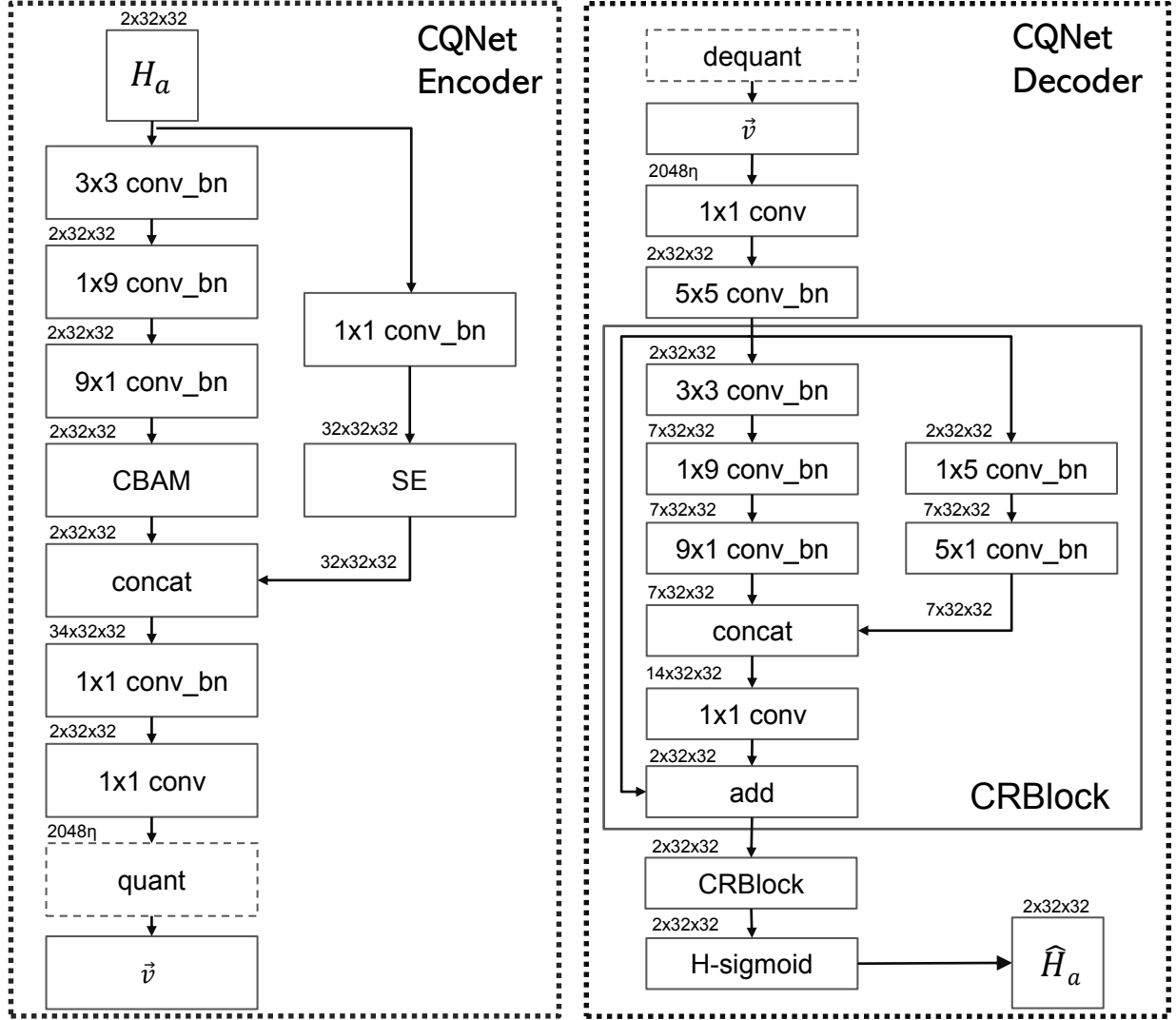


Fig. 9: Detailed encoder and decoder design of the proposed CQNet. All the input feature shape($c \times h \times w$) is given on top of the corresponding block. Conv represents convolutional operation, number in front is the size of the filter, bn represents Batch-norm operation and activation layers are left out for simplicity.

REFERENCES

- [1] T. L. Marzetta, "Noncooperative cellular wireless with unlimited numbers of base station antennas," *IEEE transactions on wireless communications*, vol. 9, no. 11, pp. 3590–3600, 2010.
- [2] T. L. Marzetta, G. Caire, M. Debbah, I. Chih-Lin, and S. K. Mohammed, "Special issue on massive mimo," *Journal of communications and networks*, vol. 15, no. 4, pp. 333–337, 2013.
- [3] E. Björnson, E. G. Larsson, and T. L. Marzetta, "Massive mimo: Ten myths and one critical question," *IEEE Communications Magazine*, vol. 54, no. 2, pp. 114–123, 2016.
- [4] L. Lu, G. Y. Li, A. L. Swindlehurst, A. Ashikhmin, and R. Zhang, "An overview of massive mimo: Benefits and challenges," *IEEE journal of selected topics in signal processing*, vol. 8, no. 5, pp. 742–758, 2014.
- [5] Y. Zhou, M. Herdin, A. M. Sayeed, and E. Bonek, "Experimental study of mimo channel statistics and capacity via the virtual channel representation," *Univ. Wisconsin-Madison, Madison, WI, USA, Tech. Rep.*, vol. 5, pp. 10–15, 2007.
- [6] P. Kyritsi, D. C. Cox, R. A. Valenzuela, and P. W. Wolniansky, "Correlation analysis based on mimo channel measurements in an indoor environment," *IEEE Journal on Selected areas in communications*, vol. 21, no. 5, pp. 713–720, 2003.
- [7] R. Roy and T. Kailath, "Esprit-estimation of signal parameters via rotational invariance techniques," *IEEE Transactions on acoustics, speech, and signal processing*, vol. 37, no. 7, pp. 984–995, 1989.
- [8] I. Daubechies, M. Defrise, and C. De Mol, "An iterative thresholding algorithm for linear inverse problems with a sparsity constraint," *Communications on Pure and Applied Mathematics: A Journal Issued by the Courant Institute of Mathematical Sciences*, vol. 57, no. 11, pp. 1413–1457, 2004.
- [9] C. Li, W. Yin, and Y. Zhang, "User's guide for tval3: Tv minimization by augmented lagrangian and alternating direction algorithms," *CAAM report*, vol. 20, no. 46-47, p. 4, 2009.
- [10] C. A. Metzler, A. Maleki, and R. G. Baraniuk, "From denoising to compressed sensing," *IEEE Transactions on Information Theory*, vol. 62, no. 9, pp. 5117–5144, 2016.
- [11] G. E. Hinton and R. R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *science*, vol. 313, no. 5786, pp. 504–507, 2006.
- [12] D. L. Donoho, A. Maleki, and A. Montanari, "Message-passing algorithms for compressed sensing," *Proceedings of the National Academy of Sciences*, vol. 106, no. 45, pp. 18 914–18 919, 2009.
- [13] J. Guo, C.-K. Wen, S. Jin, and G. Y. Li, "Convolutional neural network-based multiple-rate compressive sensing for massive mimo csi feedback: Design, simulation, and analysis," *IEEE Transactions on Wireless Communications*, vol. 19, no. 4, pp. 2827–2840, 2020.
- [14] 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)*. IEEE, 2020, pp. 1–6.
- [15] T. Wang, C.-K. 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, 2018.
- [16] C.-K. Wen, W.-T. Shih, and S. Jin, "Deep learning for massive mimo csi feedback," *IEEE Wireless Communications Letters*, vol. 7, no. 5, pp. 748–751, 2018.
- [17] Y. Yang, F. Gao, G. Y. Li, and M. Jian, "Deep learning-based downlink channel prediction for fdd massive mimo system," *IEEE Communications Letters*, vol. 23, no. 11, pp. 1994–1998, 2019.
- [18] E. J. Candes, J. K. Romberg, and T. Tao, "Stable signal recovery from incomplete and inaccurate measurements," *Communications on Pure and Applied Mathematics: A Journal Issued by the Courant Institute of Mathematical Sciences*, vol. 59, no. 8, pp. 1207–1223, 2006.

- [19] C. Trabelsi, O. Bilaniuk, Y. Zhang, D. Serdyuk, S. Subramanian, J. F. Santos, S. Mehri, N. Rostamzadeh, Y. Bengio, and C. J. Pal, “Deep complex networks,” *arXiv preprint arXiv:1705.09792*, 2017.
- [20] J. Hu, L. Shen, and G. Sun, “Squeeze-and-excitation networks,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 7132–7141.
- [21] S. Woo, J. Park, J.-Y. Lee, and I. So Kweon, “Cbam: Convolutional block attention module,” in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 3–19.
- [22] A. Howard, M. Sandler, G. Chu, L.-C. Chen, B. Chen, M. Tan, W. Wang, Y. Zhu, R. Pang, V. Vasudevan *et al.*, “Searching for mobilenetv3,” in *Proceedings of the IEEE International Conference on Computer Vision*, 2019, pp. 1314–1324.
- [23] A. Krizhevsky and G. Hinton, “Convolutional deep belief networks on cifar-10,” *Unpublished manuscript*, vol. 40, no. 7, pp. 1–9, 2010.
- [24] L. Liu, C. Oestges, J. Poutanen, K. Haneda, P. Vainikainen, F. Quitin, F. Tufvesson, and P. De Doncker, “The cost 2100 mimo channel model,” *IEEE Wireless Communications*, vol. 19, no. 6, pp. 92–99, 2012.