Beamspace Channel Estimation for Wideband Millimeter-Wave MIMO: A Model-Driven Unsupervised Learning Approach

Hengtao He, Member, IEEE, Rui Wang, Student Member, IEEE, Weijie Jin, Graduate Student Member, IEEE, Shi Jin, Senior Member, IEEE, Chao-Kai Wen, Senior Member, IEEE, and Geoffrey Ye Li, Fellow, IEEE

Abstract—Millimeter-wave (mmWave) communications have been one of the promising technologies for future wireless networks that integrate a wide range of data-demanding applications. To compensate for the large channel attenuation in mmWave band and avoid high hardware cost, a lens-based beamspace massive multiple-input multiple-output (MIMO) system is considered. However, the spatial-wideband effect in wideband mmWave systems makes channel estimation very challenging, especially when the receiver is equipped with a limited number of radio-frequency (RF) chains. Furthermore, the real channel data cannot be obtained before the mmWave system is used in a new environment, which makes it impossible to train a deep learning (DL)-based channel estimator using real data set beforehand. To solve the problem, we propose a model-driven unsupervised learning network, named learned denoising-based generalized expectation consistent (LDGEC) signal recovery network. By utilizing the Stein’s unbiased risk estimator loss, the LDGEC network can be trained only with limited measurements corresponding to the pilot symbols, instead of the real channel data. Even if designed for unsupervised learning, the LDGEC network can be supervisingly trained with the real channel via the denoiser-by-denoiser way. The numerical results demonstrate that the LDGEC-based channel estimator significantly outperforms state-of-the-art compressive sensing-based algorithms when the receiver is equipped with a small number of RF chains and low-resolution ADCs.

Index Terms—mmWave communications, deep learning, model-driven, deep unfolding, spatial-wideband effect, massive MIMO, beamspace channel estimation, unsupervised learning.

I. INTRODUCTION

Millimeter-wave (mmWave) communications have been considered as a promising technology to support the very high data rate in future wireless communications since it can provide a tenfold increase in the bandwidth [1], [2], [3]. However, as the carrier frequency increases, the mmWave signals suffer from much more severe attenuation, which becomes a vital issue in mmWave communications. Leveraging the large antenna arrays employed at the transmitter and receiver, massive multiple-input multiple-output (MIMO) can perform directional beamforming to achieve a high beamforming gain, which helps overcome large pathloss of mmWave signals and guarantees sufficient received signal-to-noise ratio (SNR).

For mmWave channel estimation, many algorithms have been proposed for hybrid, lens-array, and low-resolution ADCs architectures [5], [6], [7], [8], [9], [10], [11], [12], [13], [14], [15], [16], [17], [18], [19], [20], [21], [22]. Specifically, several works utilize compressive sensing (CS) techniques [5], [7], [8], [9], [10], [12] in mmWave band for beamspace channel estimation. The training-based scheme in [10] first scans all the beams and retains only a few strong ones. Then,
the least-square (LS) algorithm is employed for estimating the reduced-dimensional beamspace channel. In [12], a modified version of [10] reduces the overhead of beam training by simultaneously scanning several beams with the help of power splitters at the BS. However, the aforementioned algorithms are not optimized for lens-based mmWave systems because the lens antenna array has energy-focusing capability, and the received signal matrix from the lens antenna array is characterized by sparsity and concentration. The support detection based scheme in [7] further reduces the pilot overhead, which directly estimates the channel support by exploiting the sparsity of the beamspace channel. Then, a novel channel estimation algorithm is proposed to adaptively denoise the channel vectors in the beamspace domain using an adaptive shrinkage procedure [8]. In [9], the channel matrix is regarded as a 2-dimensional (2D) natural image and is then estimated by the cosine analysis approximate message passing (SCAMP) algorithm derived from the image recovery field. The SCAMP algorithm models the channel as a sparse generic L-term Gaussian mixture (GM) probability distribution and uses the expectation-maximization (EM) algorithm to learn the GM parameters from the current estimated data. By considering similar ideas from [9], EM learning has been combined with generalized expectation consistent signal recovery (GEC-SR) [21] and generalized approximate message passing (GAMP) [14] for mmWave channel estimation, respectively, which show the competitive performance.

Previous works only address narrowband mmWave systems. For wideband mmWave massive MIMO systems, several channel estimation approaches have been proposed by assuming the common support for the channel in each subcarrier [6], [20]. However, the physical propagation delays of electromagnetic waves traveling across the whole array will become large and comparable to the time-domain sample period. In such a case, different antenna elements will receive different time-domain symbols, which is known as the spatial-wideband effect in the spatial-time and beam-frequency domain [18]. As a result, the AoAs (AoDs) will become frequency-dependent; thereby, channel estimation becomes very challenging, especially in mmWave band. A super-resolution CS approach is developed to extract the frequency-insensitive parameters of each uplink channel path, i.e., the AoA and the time delay, and the frequency-sensitive parameter, i.e., the complex channel gain for hybrid architecture [22]. Then, the successive support detection (SSD) technique proposed in [19] applies successive interference cancelation to estimate the channel for lens-arry architecture. The main idea is that each sparse path component has frequency-dependent support determined by its spatial direction and is estimated using beamspace windows. However, some important characteristics of mmWave channels, such as sparsity and channel correlation between adjacent antennas and subcarriers, are not considered. These characteristics are significant for performance improvement in channel estimation, but are difficult to be characterized by traditional model-based method.

Recently, deep learning (DL) has been applied to physical layer communications [15], [16], [23], [24], [25], [27], [28], [29] such as channel state information (CSI) feedback [25], signal detection [26], [27], channel estimation [15], [16], [28], precoder design [29], traffic analysis [30], [31], [32], and end-to-end transceiver design [33], [34]. DL-based physical layer communications may be data-driven and model-driven [24]. By incorporating domain knowledge into network design, model-driven DL can reduce the demand for computing resources and training time, which is more attractive for wireless communications [15]. As a promising model-driven DL approach, deep unfolding has been first proposed in [35] and applied to sparse signal recovery [36] and image processing [37]. The main idea is unfolding the iterative algorithm into a deep neural network and adding learnable parameters, which has been successfully applied to physical layer communications [15], [24], [38]. For example, the deep unfolding-based channel estimator has been successfully applied for narrowband beamspace mmWave massive MIMO systems in [15]. It outperforms state-of-the-art CS-based algorithms and can achieve excellent performance even with a small number of RF chains. However, the existing DL-based channel estimator utilizes a supervised way [15], [16], [28] thereby a large number of real channel data are required to train the network, which defeats the point of channel estimation in the first place. This is because we cannot obtain the true channel data for training the network when the system is equipped in a practical environment. Therefore, how to train the DL-based channel estimator without the true channel data is significantly important.

B. Contributions

In this study, we develop a DL-based channel estimator for lens-based mmWave massive MIMO systems. Instead of considering supervised learning for narrowband beamspace channel estimation [15], we investigate unsupervised learning for a wideband system and take the spatial-wideband effect into consideration. To the best of our knowledge, this paper is the first study applies model-driven unsupervised DL network into wideband mmWave beamspace massive MIMO systems and considers the spatial-wideband effect. The main contributions are summarized as follows.

- We first formulate the wideband beamspace channel estimation problem as a compressed image recovery problem. By incorporating an advanced denoising convolutional neural network (DnCNN) into the GEC-SR algorithm [39], [40], [41], we develop a model-driven DL network, named the learned denoising-based GEC (LDGEC) network. The LDGEC network uses the Stein’s unbiased risk estimator (SURE) as the loss function; thereby, it can be trained only with the received signals not the real channel data. By utilizing layer-by-layer training, the LDGEC-based estimator can significantly outperform state-of-the-art CS algorithms even without the real channel data.
Even if designed for unsupervised learning, the LDGEC network can also be supervisingly trained with the real channel data, thereby further improve channel estimation performance with available channel data. In this case, we can train the DnCNN denoiser in the denoiser-by-denoiser way, where the DnCNN denoiser is trained independently without including the whole GEC algorithm, thereby reducing the training complexity significantly.

To further reduce the cost and power consumption, we investigate the LDGEC-based channel estimator for systems with hardware-friendly low-resolution ADCs. Numerical results demonstrate that little performance loss is caused for LDGEC-based channel estimator when the mmWave beamspace system is with low-resolution ADCs and reduced number of RF chains.

Notations: For any matrix $A$, $A^T$ and $\text{tr}(A)$ denote the transpose and the trace of $A$, respectively. In addition, $\text{Diag}(v)$ is the diagonal matrix with $v$ on the diagonal, and $\text{div}(Q)$ is the diagonalization operator, which returns a constant vector containing the average diagonal elements of $Q$. In addition, $\odot$ and $\circ$ denote componentwise vector division and vector multiplication, respectively. Furthermore, $\mathbb{E}\{\cdot\}$ represents the expectation operator. A circular complex Gaussian with mean $\mu$ and covariance $\Omega$ can be described by the probability density function:

$$
\mathcal{N}_c(z; \mu, \Omega) = \frac{1}{\sqrt{\det(\pi\Omega)}} e^{-(z-\mu)^H \Omega^{-1} (z-\mu)}. 
$$

We use $Dz$ to denote the real Gaussian integration measure

$$
Dz = \phi(z)dz, \quad \phi(z) \equiv \frac{1}{\sqrt{2\pi}} e^{-\frac{z^2}{2}},
$$

$Dz_c = \frac{e^{-|s|^2}}{\sqrt{2\pi}}dz$ to denote the complex Gaussian integration measure, $\Phi(x) \equiv \int_{-\infty}^{x} Dz$ to denote the cumulative Gaussian distribution function.

The remaining part of this paper is organized as follows. Section II formulates the wideband beamspace channel estimation as a compressed image recovery problem. Next, a model-driven DL network is provided for beamspace channel estimation using SURE loss in Section III. Furthermore, the network can also be trained with the real channel data in Section IV. Then, numerical results are presented in Section V. Finally, Section VI concludes the paper.

II. SYSTEM MODEL AND PROBLEM FORMULATION

In this section, we first present the lens-based mmWave MIMO-OFDM systems. After introducing the spatial-wideband effect, we formulate the wideband beamspace channel estimation as a compressed image recovery problem.

A. Beamspace Channel Model

As illustrated in Fig.1, we consider an uplink wideband beamspace mmWave MIMO-OFDM system, where the base station (BS) employs an $N$-element lens antenna array and $N_{RF}$ RF chains to simultaneously serve $K$ single-antenna users. The number of OFDM subcarriers is denoted as $M$. Applying the classical Saleh-Valenzuela channel model [42] in wideband systems, the spatial channel $h_m \in \mathbb{C}^{N \times 1}$ at sub-carrier $m$ is given by

$$
h_m = \sqrt{\frac{N}{L}} \sum_{l=1}^{L} \alpha_l e^{-j2\pi \tau_l f_m} a(\phi_{l,m}),
$$

for $m = 1, 2, \ldots, M$, where $L$ is the number of resolvable paths, $\alpha_l$ and $\tau_l$ are the complex gain and the time delay of the $l$-th path, respectively. Furthermore, $a(\phi_{l,m})$ is the array response vector and $\phi_{l,m}$ is the spatial direction at sub-carrier $m$ defined as

$$
\phi_{l,m} = \frac{f_m}{c} d \sin \theta_l,
$$

where $f_m = f_c + \frac{L}{M} (m-1 - M\frac{1}{L})$ is the frequency of sub-carrier $m$ with $f_c$ and $f_s$ representing the carrier frequency and bandwidth, respectively. Furthermore, $c$ is the speed of light, $\theta_l$ is the physical direction, and $d$ is the antenna spacing, which is usually designed according to the carrier frequency as $d = c/2f_c$. Consider a uniform linear lens array in the BS, the array response vector $a(\phi_{l,m})$ is given by

$$
a(\phi_{l,m}) = \frac{1}{\sqrt{N}} [e^{-j2\pi \phi_{l,m}(-\frac{N-1}{2})}, e^{-j2\pi \phi_{l,m}(-\frac{N+1}{2})}, \ldots, e^{-j2\pi \phi_{l,m}(\frac{N-1}{2})}]^T.
$$
The conventional channel in the spatial domain in (1) can be transformed to the beamspace by employing a carefully designed lens antenna array [10], [11], [19], as shown in Fig.1. Specifically, we assume the lens antenna array plays the role of an $N$-element spatial discrete Fourier transform (DFT) matrix $F^2$, which contains the array response vectors of $N$ orthogonal directions (beams) covering the entire space as

$$F = [a(\tilde{\phi}_1), a(\tilde{\phi}_2), \ldots, a(\tilde{\phi}_N)],$$

where $\tilde{\phi}_n = \frac{1}{N}(n - \frac{N+1}{2})$ for $n = 1, 2, \ldots, N$ are the spatial directions pre-defined by the lens antenna array. Accordingly, the wideband beamspace channel $\tilde{h}_m$ at sub-carrier $m$ can be expressed as

$$\tilde{h}_m = F^H h_m = \sqrt{\frac{N}{L}} \sum_{l=1}^{L} \alpha_l e^{-j2\pi f_m c_{l,m}},$$

where $c_{l,m}$ denotes the $l$-th path component at sub-carrier $m$ in the beamspace, and $c_{l,m}$ is determined by $\phi_{l,m}$ as

$$\tilde{c}_{l,m} = F^H a(\phi_{l,m}) = \left[ \Xi(\phi_{l,m} - \tilde{\phi}_1), \Xi(\phi_{l,m} - \tilde{\phi}_2), \ldots, \Xi(\phi_{l,m} - \tilde{\phi}_N) \right]^T,$$

where $\Xi(x) = \frac{\sin N 2\pi x}{\sin \pi x}$ is the Dirichlet sinc function.

**B. Spatial-Wideband Effect**

Before formulating the wideband beamspace channel estimation problem, we introduce the spatial-wideband effect illustrated in beamspace channel [17], [18], [19]. Based on the power-focusing capability of $\Xi(x)$, we know that most of the power of $c_{l,m}$ is focused on only small number of elements. Additionally, due to the limited scattering in the mmWave systems, the number of resolvable paths, $L$, is generally small. However, the beam power distribution of the $l$-th path component will be different at different subcarriers, i.e., $c_{l,m_1} \neq c_{l,m_2}$ for $m_1 \neq m_2$, since $\phi_{l,m}$ is frequency-dependent in wideband mmWave systems (i.e., $f_m \neq f_q$). This effect is termed as spatial-wideband effect in the beam-frequency domain [18], which is a key difference between wideband and narrowband systems.

To show the spatial-wideband effect in beam domain, we present the energy diagram of the beam-frequency channel matrix in narrowband and wideband systems. We consider a beamspace system with $L = 3$, $M = 128$, $N = 32$, $f_c = 28$ GHz. Furthermore, we set $f_s = 4$ GHz in a wideband system and $f_s = 20$ MHz in the narrowband system. As illustrated in Fig.2, the beam power distribution of the $l$-th path component is almost similar at different sub-carriers in the narrowband system. Therefore, the beamspace channel support (the index of non-zero elements) at different frequencies can be assumed to be the same.

Owing to the spatial-wideband effect, the beam power distribution in wideband systems varies significantly over frequency. We denote the beam channel vector corresponding to the $l$-th path and the $m$-th frequency as $\tilde{h}_{l,m}$. As shown in Fig.2, the index of the strongest element in $\tilde{h}_{1,0}$ (i.e., the yellow bar at the bottom of the Fig.2) is 24, while the index of the strongest element in $\tilde{h}_{3,128}$ is 26. Thus, the beamspace channel supports at different frequencies are different. The characteristic of the beam-frequency matrix will bring a significant challenge for wideband beamspace channel estimation. Furthermore, the channel correlation between adjacent antennas and subcarriers is subtle, which is difficult to be characterized by the traditional approaches. Conversely, DL has the powerful capability to learn the correlation from the data, which is more promising for wideband channel estimation involved in the spatial-wideband effect.

**C. Problem Formulation**

In uplink channel estimation, the user devices transmit pilot sequences to the BS, and the channel is assumed to remain unchanged during this period. We use the orthogonal pilot sequence for different users. Therefore, the channel estimation can be performed for each user independently. Considering a specific user, the pilot at sub-carrier $m$ in instant $q$, $s_{m,q}$, is transmitted. The received signal vector $y_{m,q} \in \mathbb{C}^{N \times 1}$ at the
BS is given by

\[ y_{m,q} = W_q \tilde{h}_m s_{m,q} + W_q n_{m,q}, \]

for \( m = 1, 2, \ldots, M \) and \( q = 1, 2, \ldots, Q \), where \( n_{m,q} \sim \mathcal{N}(0, \sigma_n^2 I) \) represents a Gaussian noise vector. \( W_q \in \mathbb{C}^{N_{RF \times N}} \) is the adaptive selection network but fixed for different sub-carriers. We set \( s_{m,q} = 1 \) for convenience as the pilot signal is known at the receiver side. Thus, the received signal \( \tilde{y}_m \) in \( Q \) instants is given by

\[ \tilde{y}_m = [y_{m,1}^T, \ldots, y_{m,Q}^T]^T = \tilde{W} \tilde{h}_m + n_{eq}^m, \]

where

\[ W_q = [(W_1^T, W_2^T, \ldots, W_Q^T)^Q_{RF \times N}], \quad n_{eq}^m = [(W_1 n_{m,1})^T, (W_2 n_{m,2})^T, \ldots, (W_Q n_{m,Q})^T]^T. \]

In this paper, low-cost one-bit phase shifters are utilized in the adaptive selection network \( W_q \). Therefore, the elements of \( \tilde{W} \) are randomly selected from the set \( \{ -1, +1 \} \) with equal probability.

From Fig. 2 and Section II-A, the beamspace channel vectors at different subcarriers, even if different due to spatial-wideband effect in the beam-frequency domain, are correlated through antenna array response vector \( a(\phi_t, m) \), which is highly similar to a 2D natural image. By stacking \( M \) beamspace channel vectors into a matrix, we have the following signal recovery model,

\[ [\tilde{y}_1, \tilde{y}_2, \ldots, \tilde{y}_M] = W [\tilde{h}_1, \tilde{h}_2, \ldots, \tilde{h}_M] + [n_{1eq}^q, n_{2eq}^q, \ldots, n_{Meq}^q]. \]

If we regard beam-frequency matrix \([\tilde{h}_1, \tilde{h}_2, \ldots, \tilde{h}_M]\) as a 2D natural image, many compressed image recovery methods can be borrowed here for beamspace channel estimation, which enables us to develop a model-driven-DL-based channel estimation network.

Before introducing the DL-based channel estimation network, we first obtain a linear transformation model by stacking the \( y_m, \tilde{h}_m \) and \( n_{eq}^m \) into

\[ y = Ah + n, \]

where \( y = [\tilde{y}_1^T, \tilde{y}_2^T, \ldots, \tilde{y}_M^T]^T, \quad h = [\tilde{h}_1^T, \tilde{h}_2^T, \ldots, \tilde{h}_M^T]^T, \quad n = [(n_{1eq}^q)^T, (n_{2eq}^q)^T, \ldots, (n_{Meq}^q)^T]^T, \quad \text{and} \quad A = (I \otimes W). \]

We denote \( \otimes \) as the matrix Kronecker product. The linear transformation model (12) will be utilized in the subsequent discussion.

### III. UNSUPERVISED LEARNING FOR BEAMSPACE CHANNEL ESTIMATION

In this section, we propose a model-driven unsupervised DL network for wideband beamspace channel estimation, named LDGEC-based channel estimator. As in [24], the network is specially designed by unfolding an iterative algorithm, GEC-SR algorithm, with the DL-based denoiser. After introducing the network architecture and DnCNN denoiser, we elaborate the layer-by-layer training approach, which is critical to implementing LDGEC network with unsupervised learning.

#### A. LDGEC-Based Channel Estimator

As illustrated in Fig. 3, the input of the LDGEC network is the received signal vector, \( \tilde{y} \), and the linear transform matrix, \( A \), while the final output is \( \hat{h} \), the estimated channel vector. The LDGEC network consists of \( T \) layers connected in cascade. We replace the posterior mean estimator in the GEC algorithm with the DnCNN denoiser and deep unfold the GEC algorithm into neural network. Each iteration of the GEC algorithm can be interpreted as each layer of the LDGEC network. As each layer of LDGEC has the same structure except the learnable parameters in DnCNN denoiser, we omit the layer index \( t \) in Fig. 3 and Algorithm 1.

As illustrated in the figure, each layer of the LDGEC network has three modules: A, B, and C. The detailed iterative
The main idea of the LDGEC network is based on the multiplication or division of Gaussian message and the moment matching. It is derived from the factor graph with the messages updated and passed between different modules assumed to follow Gaussian distribution. Therefore, only mean and variance need to be passed to characterize the Gaussian messages. We first provide a brief explanation for each module and omit the detailed derivation and principles, which can be found in our previous works [39], [40]. Specifically, module A computes the posterior mean and variance of \( z = \mathbf{A} \mathbf{h} \); module B performs denoising from the noisy signal, \( \mathbf{r}_{1h} \), by using the advanced DnCNN denoiser, and module C provides the framework that constrains the estimation problem into the linear space \( z = \mathbf{A} \mathbf{h} \). In addition, each module uses the turbo principle as in iterative decoding; that is, each module passes the extrinsic messages to its next module. The three modules are executed iteratively until convergence or terminated by a fixed number of layers.

Before introducing the whole iteration of the LDGEC network, we define two auxiliary variables,

\[
P_h = 1 \quad \text{and} \quad P_z = P_h \cdot \text{tr}(\mathbf{A}^H \mathbf{A})/MN_{RF}Q,\]

(13)

which are interpreted as the powers of \( h_n \) and \( z_n \), respectively. \( h_n \) and \( z_n \) denote the \( n \)-th element in \( \mathbf{h} \) and \( \mathbf{z} \), respectively. The \( P_h \) and \( P_z \) are important for network initialization. To better understand the LDGEC network, we provide detailed explanations for each line in Algorithm 1. Specifically, Lines 1–2 compute the posterior mean and variance of \( z_n \) from the measurements \( \hat{y}_n \), and the expectation is w.r.t. the posterior probability

\[
P_Z(z_n|\hat{y}_n) = \frac{P_{\text{out}}(\hat{y}_n|z_n)P_Z(z_n)}{\int P_{\text{out}}(\hat{y}_n|z_n)P_Z(z_n)dz_n},\]

(14)

where \( P_Z(z_n) \) is assumed to be \( \mathcal{N}(z_n; r_{1z,n}, v_{1z,n}) \). \( \hat{y}_n \), \( z_n \), \( r_{1z,n} \), \( v_{1z,n} \) are the \( n \)-th element in \( \hat{y} \), \( z \), \( \mathbf{r}_{1z} \), \( \mathbf{v}_{1z} \), respectively. The posterior probability \( P_Z(z_n|\hat{y}_n) \) is dependent on \( P_{\text{out}}(\hat{y}_n|z_n) \). Therefore, the exact expressions for Lines 1 and 2 will be different if other \( P_{\text{out}}(\hat{y}_n|z_n) \) is considered. To clearly understand Lines 1 and 2 in Algorithm 1, we take the quantized and unquantized channels as two examples.

1) **Unquantized Channel:** If with infinite-resolution ADCs, the received signal at the BS, \( \bar{y} = y \) and the posterior probability \( P_{\text{out}}(\hat{y}_n|z_n) \) is given by

\[
P_{\text{out}}(\hat{y}_n|z_n) = \frac{1}{\pi \sigma_n^2}e^{\frac{|\mathbf{h}_n \cdot \mathbf{z} - z_n|^2}{2 \sigma_n^2}}.\]

(15)

Thus, the explicit expressions of the posterior mean and variance will be

\[
\hat{z}_1 = r_{1z} + \frac{v_{1z}}{v_{1z} + \sigma_n^2}(\hat{y} - r_{1z}),
\]

\[
v_{1z}^{\text{post}} = v_{1z} - \frac{v_{1z}^2}{v_{1z} + \sigma_n^2},
\]

(16)

(17)

2) **Quantized Channel:** If the low-resolution ADCs are used in the BS, the received signal \( \hat{y} = Q_{\chi}(y) \). Then, the explicit expressions of the posterior mean and variance can be derived similar to [47, Appendix A] as

\[
\hat{z}_1 = r_{1z} + \frac{\text{sign}(\hat{y})v_{1z}}{\sqrt{2(\sigma_n^2 + v_{1z})}}\left(\frac{\phi(\eta_1) - \phi(\eta_2)}{\Phi(\eta_1) - \Phi(\eta_2)}\right),
\]

(18)

\[
v_{1z}^{\text{post}} = \frac{v_{1z}^2}{2(\sigma_n^2 + v_{1z})} \cdot \left(\frac{\eta_1 \phi(\eta_1) - \eta_2 \phi(\eta_2)}{\Phi(\eta_1) - \Phi(\eta_2)} + \left(\frac{\phi(\eta_1) - \phi(\eta_2)}{\Phi(\eta_1) - \Phi(\eta_2)}\right)^2\right),
\]

(19)

where

\[
\eta_1 = \frac{\text{sign}(\hat{y})r_{1z} - \min(|\hat{y}|_{\text{low}}, |\hat{y}|_{\text{up}})}{\sqrt{(\sigma_n^2 + v_{1z})/2}}.
\]

(20a)
\[
\eta_2 = \frac{\text{sign}(\tilde{y}) \cdot r_{12} - \max\{|r_{\text{low}}|, |r_{\text{up}}|\}}{\sqrt{(\sigma_n^2 + v_1^2)/2}},
\]

where \(r_{\text{low}}\) and \(r_{\text{up}}\) are the lower and upper thresholds associated with \(\tilde{y}_n\), respectively. For notational convenience, we omit index \(n\) and have

\[
r_{\text{low}} = \begin{cases} \tilde{y} - \frac{\Delta}{2} & \text{for } \tilde{y} \geq -\left(\frac{2\kappa}{2} - 1\right) \Delta, \\ -\infty, & \text{otherwise,} \end{cases}
\]

and

\[
r_{\text{up}} = \begin{cases} \tilde{y} + \frac{\Delta}{2} & \text{for } \tilde{y} \leq \left(\frac{2\kappa}{2} - 1\right) \Delta, \\ \infty, & \text{otherwise.} \end{cases}
\]

In this paper, we mainly focus on a typical uniform midrise channel from equivalent AWGN observations. Expressions (18) and (19) are the estimators for the real and imaginary parts, respectively. For notational convenience, we omit index \(n\) in the aforementioned expression. The estimator for the imaginary part \(\text{Im}(\hat{z}_1)\) can be obtained similarly as (18) and (19) while \(\tilde{y}\) and \(b\) should be replaced by \(\text{Im}(\tilde{y})\) and \(b^r\), respectively.

Then, lines 3–4 compute the extrinsic information of \(\mathbf{z}\) using the turbo principle. Lines 5–6 perform the linear minimum mean-squared error estimate of \(\mathbf{h}\) under the following assumption,

\[
r_{2x} = \mathbf{z}_2 + \mathbf{w}_{2x},
\]

where \(\mathbf{w}_{2x} \sim \mathcal{N}_C(0, \text{Diag}(\mathbf{v}_{2x}))\), \(\mathbf{z}_2 = \mathbf{A}\mathbf{h}_2\), and \(\mathbf{h}_2 \sim \mathcal{N}_C(\mathbf{h}_2; \mathbf{r}_{2h}, \text{Diag}(\mathbf{v}_{2h}))\). Lines 7–8 compute the extrinsic information of \(\mathbf{h}\) and pass it to module B as a prior information. Lines 9–10 estimate the mean, \(\mathbf{h}_1\), and variance, \(\mathbf{v}_{1\text{post}}\), which is assumed to estimate \(\mathbf{h}\) from several AWGN observations, that is,

\[
r_{1h} = \mathbf{h} + \mathbf{w}_{1h},
\]

where \(\mathbf{w}_{1h} \sim \mathcal{N}_C(0, \text{Diag}(\mathbf{v}_{1h}))\). The process of estimating channel \(\mathbf{h}\) from equivalent AWGN observations \(\mathbf{r}_{1h}\) in (24) is equivalent to perform the denoising from \(\mathbf{r}_{1h}\). As channel \(\mathbf{h}\) can be regarded as a 2D natural image, we utilize the advanced DnCNN denoiser [48] in lines 9–10 to recover channel \(\mathbf{h}\) from equivalent noisy observations \(\mathbf{r}_{1h}\). Lines 11–12 compute the extrinsic information of \(\mathbf{h}\) using the turbo principle, and lines 13–16 constrain the estimated problem into a linear space \(\mathbf{z} = \mathbf{A}\mathbf{h}\) which performs the same procedure as Lines 5–6. Lines 17–18 compute the extrinsic information of \(\mathbf{z}\) and pass to module A as the prior information.

\[5\] The extrinsic information of \(\mathbf{z}\) can be interpreted as computing the mean and variance of a Gaussian random variable \(\mathbf{z} = \mathbf{z}_1 \otimes r_{1z}\), which is obtained by a Gaussian random variable \(\mathbf{z}_1\) divided by another Gaussian variable \(r_{1z}\).

Posterior variance \(\mathbf{v}_{1h}^{\text{post}}\) in line 10 is determined by \(\text{div}D_\sigma'(\mathbf{r}_{1h})\text{avg}(\mathbf{v}_{1h})\). The divergence \(\text{div}D_\sigma'(\mathbf{r}_{1h})\) is simply the sum of the partial derivatives with respect to each element of \(\mathbf{r}_{1h}\) given by

\[
\text{div}D_\sigma'(\mathbf{r}_{1h}) = \sum_{n=1}^{MN} \frac{\partial D_\sigma'(\mathbf{r}_{1h})}{\partial r_{1h,n}},
\]

where \(r_{1h,n}\) is the \(n\)-th element of \(\mathbf{r}_{1h}\). Furthermore, \(\text{avg}(\mathbf{v}_{1h}) = \text{mean}(\mathbf{v}_{1h})\text{Diag}(\), which is the function to compute the mean of the vector \(\mathbf{v}_{1h}\), \(\text{mean}(\cdot\)) is the function to compute the mean value of a vector and \(\text{Diag}(\cdot)\) is \(MN\)-dimensional all-ones vector}

Although several denoisers have been proposed for image denoising problem recently [50], they have similar performance with DnCNN denoiser.

4The exact expressions for posterior variance \(\mathbf{v}_{1h}^{\text{post}}\) can be obtained by using the method from [49, 9]. The idea behind is to compute partial derivative of the output from the denoiser, which is often used in many papers to compute the posterior variance from complex denoiser. To avoid numerical instabilities, we average the variance in line 10 to compute the posterior variance \(\mathbf{v}_{1h}^{\text{post}}\).
handle the Gaussian denoising problem with an unknown noise level, which is more accurate and faster than competing techniques. Fig. 4 illustrates the network architecture of the DnCNN denoiser. It consists of 20 convolutional layers. The first convolutional layer uses 64 different 3 × 3 × 1 filters and is followed by a rectified linear unit (ReLU). Each of the succeeding 18 convolutional layers uses 64 different 3 × 3 × 64 filters, each followed by batch-normalization and a ReLU. The final convolutional layer uses one separate 3 × 3 × 64 filters to reconstruct the signal. Instead of learning a mapping directly from a noisy image to a denoised image, learning the residual noise is beneficial.

We plot three pseudo-color images of noisy channel, residual noise, and estimated channel in Fig. 4. The network is given the noisy observation h + w as an input, where w is the AWGN and noise variance σ is uniformly generated from a specific interval. The network produces residual noise ẑ, rather than an estimated channel ĥ, as an output. This method, known as residual learning [51], renders the network to remove the highly structured natural image rather than the unstructured noise. Consequently, residual learning improves both the training times and accuracy of a network. Furthermore, the DnCNN adopts the method of batch normalization, can speed up the training process, and boost the denoising performance [48].

C. Stein’s Unbiased Risk Estimator

Recently, many DL-based channel estimators have been proposed for different communication scenarios [15]. A common limitation of these works is the extensive real channel data should be obtained before training the network because the MSE function \( \text{MSE} = \mathbb{E}\{\|\hat{h} - h\|^2\} \) involves in the real channel data h. These requirements bring significant challenges when the real data cannot be obtained, e.g., the DL-based channel estimator is equipped in a new channel environment and only the received signal ỹ is obtained at the BS. Furthermore, the existing DL-based channel estimator utilizes supervised way and requires a large number of real channel data, which defeats the point of channel estimation. In this circumstance, how to train the network without the real channel data and only with measurements corresponding to the pilot symbols is significantly important. To solve the problem, we introduce the SURE loss function [52], which is a classical approach for learning images from noisy observations and has been extended to linear noisy measurements. It has been applied in medical imaging, microscopy, and astronomy, where the ground truth data is rarely available [53]. The SURE loss is an unbiased estimator of

\[
\text{MSE} = \mathbb{E}_w\left[\frac{1}{P}\|\hat{h} - f(\hat{y}_w)\|^2\right],
\]

where \( \hat{y}_w = h + w \) and \( f(\cdot) \) is an estimator of h from \( y_w \). Therefore, it can be used for training a DL-based denoiser to replace the MSE loss.

The goal of channel estimation is to reconstruct a channel \( h \) from noisy linear observations \( y = Ah + n \) with the known linear transform matrix A. We are given training measurements \( y^1, y^2, \ldots, y^D \) but not the real channel \( h^1, h^2, \ldots, h^D \). Without access to \( h^1, h^2, \ldots, h^D \), the ground truth data, we cannot train the DnCNN denoiser by minimizing the traditional MSE loss function. Fortunately, we can use the SURE loss instead. SURE is a model selection technique first proposed by its namesake in [52]. It provides an unbiased estimate of the MSE for an estimator of the mean of a Gaussian distributed random vector with an unknown mean. Let \( x \) denote a vector we would like to estimate from noisy observations \( r_{1h} = h + w_{1h} \) where \( w_{1h} \sim \mathcal{N}(0, \text{Diag}(w_{1h})) \).

We assume the DnCNN function \( f_\theta(\cdot) \) is a weakly differentiable function parameterized by \( \theta \), which receives noisy observations \( r_{1h} \) as input and provides an estimate of h as output. Then, according to [52] and [53], we can express the expectation of the MSE of the real channel h and equivalent noisy observations \( r_{1h} \) with respect to the random variable \( w_{1h} \) as follows,

\[
\text{MSE} = \mathbb{E}_{w_{1h}}\left[\frac{1}{P}\|h - f_\theta(r_{1h})\|^2\right].
\]

Then, the MSE loss can be computed as follows,

\[
\mathbb{E}_{w_{1h}}\left[\frac{1}{P}\|h - f_\theta(r_{1h})\|^2\right] = \mathbb{E}_{w_{1h}}\left[\frac{1}{P}\|r_{1h} - f_\theta(r_{1h})\|^2\right] - \mathbb{V}_{1h} - \frac{2\mathbb{V}_{1h}}{P}\text{div}(f_\theta(r_{1h})),
\]

where \( P = MN \) and \( \text{div}(\cdot) \) stands for divergence defined as (25). Note that two terms within the SURE loss depend on

![Fig. 4. Network architecture of the DnCNN denoiser.](image-url)
parameter \( \theta \). The first term, \( \mathbb{E}_{w_{1h}}[\frac{1}{2}\|r_{1h} - f_{\theta}(r_{1h})\|^2] \), indicates the difference between estimate \( f_{\theta}(r_{1h}) \) and observation \( r_{1h} \) (bias). The second term, \( \frac{2\sigma^2}{P}\text{div}(f_{\theta}(r_{1h})) \), penalizes the denoiser for varying as the input is changed. Thus, SURE is a natural way to control the trade-off between the bias and variance of a recovery algorithm.

The critical challenge for using SURE in practice is to compute divergence \( \text{div}(f_{\theta}(r_{1h})) \). For the advanced DnCNN denoiser, the divergence is hard or even impossible to express analytically. Therefore, we cannot obtain a closed form for the divergence. Similar to (26), we can use a Monte Carlo method to estimate the divergence \( \text{div}(f_{\theta}(r_{1h})) \). Combining the SURE loss in (32) and the estimate of divergence \( \text{div}(f_{\theta}(r_{1h})) \), we can minimize the MSE loss function of a denoising problem without ground truth data. Note that minimizing SURE loss requires propagating gradients with respect to the Monte Carlo estimate of divergence (25). Although the gradients are challenging to compute by hand, we can resort to TensorFlow’s auto-differentiation capabilities to propagate it.

### D. Layer-by-Layer Training

A significant reason for the LDGEC-based channel estimator trained with SURE loss is layer-by-layer training. From (32), the computational process of the SURE loss requires noisy observations \( r_{1h} \) and equivalent noise variance \( v_{1h} \). This method takes advantage of the fact that each layer of the LDGEC-based channel estimator is to solve a denoising problem with known variance \( v_{1h} \) and noisy observations \( r_{1h} \). As the LDGEC network can decouple the linear model in (12) into several equivalent AWGN models (24) in each layer and \( v_{1h} \) computed in line 7 in Algorithm 1 is accurate enough to describe the variance of \( w_{1h} \). Therefore, we can train the \( t \)-th layer LDGEC network with the SURE loss, estimated variance \( v_{1h}^{t} \) and noisy observations \( r_{1h}^{t} \) through layer-by-layer training.

In the \( t \)-th round of the layer-by-layer training, the loss function is given by

\[
L_{\text{SURE}}^{t}(\theta^{t}) = \arg\min_{\theta^{t}} \sum_{d=1}^{D} \left[ \frac{1}{P}\|r_{1h}^{t,d} - f_{\theta^{t}}(r_{1h}^{t,d})\|^2 - v_{1h}^{2} \right] + \frac{2v_{1h}^{2}}{P}\text{div}(f_{\theta^{t}}(r_{1h}^{t,d})),
\]

where \( D \) is the number of mini-batches in the \( t \)-th round and \( r_{1h}^{t,d} \) is the corresponding noisy observations for sample \( h^{d} \) in the \( t \)-th LDGEC network. \( \theta^{t} \) is the required learnable variables in the \( t \)-th layer. After training the first to \( t \)-th layers, a new \( t+1 \) layer is appended to the LDGEC network and the entire network is trained again for \( D \) mini-batches. Although the objective function is changed, the values of the variables \( \theta^{0}, \ldots, \theta^{t-1} \) of the previous round are taken as the initial ones in the optimization process for the new round. In summary, the layer-by-layer training updates variables \( \theta^{t} \) in a sequential manner from the first layer to the last layer.

#### E. Complexity Analysis

The computational complexity required for the LDGEC network in each layer is dominant by matrix inverse in lines 5 and 13 in Algorithm 1. Generally, the computational complexity of matrix inverse is \( O((MN)^3) \), which cannot be acceptable. As the matrix \( A \) is a block diagonal matrix and can be expressed as

\[
A = (I \otimes \bar{W}) = \begin{bmatrix} \bar{W} & & \\ & \ddots & \\ & & \bar{W} \end{bmatrix}.
\]

Therefore, the matrix inverse in lines 5 and 13 in Algorithm 1 can be computed by matrix inverse with respect to matrix \( \bar{W} \). By inversion of the partitioned matrix, the total complexity can be reduced from \( O((MN)^3) \) to \( O(MN^3) \). We compare the complexity of LDGEC network with other CS-based algorithms. As shown in Table I, the computational complexity of LDGEC is \( O(MN^3) \), while SSD and BEACHES algorithms have the complexity of \( O(MN_{RF}QL^2\Omega^2) \) and \( O(MN\log(N)) \), respectively. Furthermore, the computational complexity of OMP is \( O(MN_{RF}QL^2\Omega^2) \). Generally, \( \Omega \) is the beamspace windows and assumed as \( \Omega = 4 \) when \( N = 256 \) [19], and is much smaller than \( N \). Although LDGEC network has higher computational complexity than CS-based algorithms but can achieve better performance in simulation results.

### IV. SUPERVISED LEARNING FOR BEAMSPACE CHANNEL ESTIMATION

In this section, we investigate the LDGEC-based channel estimator for the mmWave beamspace MIMO systems with supervised learning. After presenting three training methods for the LDGEC network, we briefly introduce the denoiser-by-denoiser training.

#### A. Denoiser-by-Denoiser Training

The LDGEC-based channel estimator can also be trained with supervised learning. We introduce three supervised training methods for the LDGEC-based channel estimator with supervised learning as follows,

* **End-to-end training**: We can train all the weights of the \( T \) layer LDGEC network simultaneously end-to-end. This is the standard method for training a neural network but with high training complexity.
Layer-by-layer training: We can train the LDGEC with layer-by-layer way by utilizing the MSE loss in (31). The training process is introduced in Section III.

Denoiser-by-denoiser training: We can decouple the denoisers from the rest of the network and train the AWGN denoising problems at different noise levels. The principle of the denoiser-by-denoiser is to train the DnCNN denoiser for the denoising problem solely instead of including the whole GEC algorithm into network training, thereby reducing the computational time in the training stage. Note that the DnCNN denoiser in the LDGEC network is trained for a specific noise level interval. As equivalent noise variance  

\[ \hat{\sigma}^2_t = \text{mean}(v_{1b}) \]

is different for each layer in the LDGEC, we need to deploy different DnCNN denoisers for different layers. To address the issue, we decouple the denoisers from the rest of the network and train each on an AWGN denoising problem at different noise levels. In particular, we scale noise level \( \hat{\sigma}^2 \) by multiplying 255 as \( \hat{\sigma}^2 = 255\hat{\sigma}^2 \) and divide \( \hat{\sigma}^2 \) into intervals \([0,10), [10,20), [20,40), [40,60), \ldots \]. For each noise interval, we generate noise variance \( \bar{\sigma}^2 \) uniformly and train a corresponding DnCNN denoiser. After training the DnCNN denoiser, we deploy the trained DnCNN denoiser into the LDGEC network to perform channel estimation. As illustrated in Fig. 5, we use a selector to choose the corresponding DnCNN denoiser according to the equivalent noise variance \( \hat{\sigma}_t \) for each layer, e.g., we use the denoiser for noise standard deviations between 40 and 60, if \( \hat{\sigma}_t^2 * 255 = 55 \).

**B. MMSE Optimal Performance**

In [54], the layer-by-layer and denoiser-by-denoiser training for LDAMP are proven to achieve MMSE optimal performance. On the contrary, the LDGEC network can also achieve the MMSE optimal performance under the following three conditions,

- The matrix \( A \) is right-orthogonally invariant.
- The noise, \( n \), is also i.i.d. Gaussian.
- The denoisers, \( D_{\sigma^2} (\cdot) \), at each layer are Lipschitz continuous.

Similar to LDAMP network in [54], LDGEC trained with denoiser-by-denoiser performs slightly worse than the end-to-end and layer-by-layer trained networks since the discretization of the noise levels is ignored in our theory. This gap can be reduced by using a finer discretization of the noise levels or by using deeper denoiser networks to handle a range of noise levels. Although matrix \( A \) in system model (12) is block-diagonal, rather than right-orthogonally invariant, the LDGEC network can still show promising performance with different training methods.

**V. NUMERICAL RESULTS**

In this section, we provide numerical results to show the performance of the proposed model-driven DL network for wideband beamspace channel estimation. First, we elaborate on the implementation details. Then, the performances of the LDGEC-based channel estimator trained with denoiser-by-denoiser and layer-by-layer are presented. Finally, we investigate the performance of the LDGEC-based channel estimator with a reduced number of RF chains and low-resolution ADCs.

**A. Implementation Details**

In the simulation, we consider the Saleh-Valenzuela channel model [42] shown in (1), which has been used to characterize the mmWave channel in several works involved in wideband mmWave channel estimation [17], [18], [19]. As the spatial wideband effect demonstrated in wideband mmWave systems, different subcarriers will experience distinct AoAs for the same physical path in wideband systems shown in Fig. 2. The simulation parameters about system settings and channel model are listed in Table II. We use the normalized

\[ \| D(x_1) - D(x_2) \| \leq L\| x_1 - x_2 \|^2 \]

and the Lipschitz continuity of the convolutional denoiser can be ensured by using weight clipping and gradient norm penalization method [55].
TABLE II
SIMULATION PARAMETERS

| Simulation parameters       | Value |
|----------------------------|-------|
| Number of Paths (L)        | 5     |
| Number of antennas (N)     | 32    |
| Number of RF chains (N_{RF}) | 8    |
| Carrier frequency (f_c)    | 28 GHz|
| Bandwidth (f_s)            | 4 GHz |
| Number of subcarriers (M)  | 64, 128|
| Complex gain (β_l)         | N_C(0, 1) |
| Angle (θ_l)                | U(−π/2, π/2) |
| Maximum Delay (τ_{max})    | 20 ns |
| Delay (τ_l)                | U(0, τ_{max}) |

MSE (NMSE) to quantify the accuracy of channel estimation for each user, which is defined as

\[ NMSE = E \left\{ \left\| \hat{h}^{\text{out}} - h \right\|^2 / \left\| h \right\|^2 \right\}, \quad (35) \]

Furthermore, the DL-based channel estimation network is implemented in Tensorflow by using a PC with GPU NVIDIA GeForce GTX 1080 Ti. The training, validation, and testing sets contain 19200, 6400, and 12800 samples, respectively, and are obtained from the Saleh-Valenzuela channel model in (1). The batch size is set to 16, and epoch is equal to 50. We generate the same adaptive selection network, W, for the channel sample in each batch while generate W independently for different batches. The LDGEC network is trained using the stochastic gradient descent method and the Adam optimizer. The training rate is set to be 0.001 initially and then dropped to 0.0001. As existing deep learning APIs are mostly devoted to processing the real-valued data, we consider equivalent real-valued representation for the system model in (12). We set damping factor β = 0.8 except additional notes. The code will be available at https://github.com/hehengtao/LDGEC.

B. Convergence Analysis

1) With real channel data: Fig. 6(a) investigates the convergence of the LDGEC network with denoiser-by-denoiser training. SNR = 0, 10, and 15 dB are considered. From the figure, the LDGEC network with denoiser-by-denoiser training converges within 6 layers, and more layers are required when the SNR is increasing. 2) Without real channel data: Fig. 6(b) demonstrates the convergence of the LDGEC-based channel estimator with layer-by-layer training. Specifically, the LDGEC-MSE means training the LDGEC network with MSE loss function while LDGEC-SURE indicates training LDGEC network with SURE loss. From Fig. 6(b), the NMSE performance of LDGEC-SURE is close to that of LDGEC-MSE when SNR = 10 dB. On the contrary, the performance gap is approximately 2.5 dB when SNR = 0 dB because the estimate of equivalent noise variance \( v_{1h} \) and Monte-Carlo approximation of divergence (26) are not accurate enough in the low-SNR regime, thereby degrading the channel estimation performance.

C. Performance Comparison

Fig. 7 compares the performance of the LDGEC network with other channel estimation algorithms. For LDGEC network with denoiser-by-denoiser, we set the number of layers \( T = 20 \) for all SNR. For the LDGEC-SURE and LDGEC-MSE, we set the number of layers \( T = 20 \) for SNR \( \leq 10 \) dB while \( T = 40 \) for SNR \( > 10 \) dB, because the LDGEC with layer-by-layer training needs more layers to converge in higher SNR. \( M = 64 \) and 128 are considered in the simulation. From the figure, the LDGEC-based channel estimator can outperform the traditional CS-based algorithms with different training methods, such as OMP [5], SSD [19], BEACHES [8]. Note that the LDGEC with layer-by-layer training can outperform that with denoiser-by-denoiser training if the MSE is considered as the loss function because we need to divide equivalent noise variance \( \sigma^2 \) into several intervals and train one DnCNN denoiser for each interval in denoiser-by-denoiser training, respectively. Instead of using the coarse intervals, the layer-by-layer training employs the accurate equivalent noise variance estimate, \( v_{1h} \), in each layer, thereby improves the denoising performance.

D. Impact of Measurement Ratio

In Section II, the measurement ratio, defined by \( \delta = Q N_{RF} / N \) and involved in the number of RF chains, \( N_{RF} \), and pilot length \( Q \), influences the performance of channel estimation and related to the system overhead. Fig. 8(a) illustrates...
the performance of the LDGEC network with denoiser-by-
denoiser training versus different measurement ratios. Since 
SSD algorithm cannot work when \( \delta \leq 1 \), we consider \( \delta = 2 \). 
From the figure, the performance of the LDGEC network 
improves as the measurement ratio increases. Interestingly, the 
LDGEC algorithm with \( \delta = 1 \) outperforms the SSD algorithm 
with \( \delta = 2 \). Furthermore, the performance of the LDGEC 
network with \( \delta = 1 \) is close to that with \( \delta = 2 \), which 
demonstrates the strong robustness to the reduced number 
of RF chains. As the measurement ratio is determined by 
\( N_{RF} \) and \( Q \), we can decrease the number of RF chains 
\( N_{RF} \) by increasing the number of pilot length \( Q \), which 
can reduce hardware cost and power consumption of the 
system significantly. Fig. 8(b) illustrates the performance of the 
LDGEC network with layer-by-layer training versus different 
measurement ratios. From the figure, we have similar conclu-
sions to that of LDGEC with denoiser-by-denoiser training.

E. Low-Resolution ADC

The lens-based beamspace mmWave system can decrease 
the hardware cost by reducing the number of RF chains. 
However, a common limitation of the architectures is that 
the receiver RF chains include high-resolution ADCs, which 
are power-hungry devices, especially when large bandwidth is 
involved. The power consumption of a typical ADC roughly 
scales linearly with the bandwidth and grows exponentially 
with the quantization bits [56]. Many researchers have studied 
the mmWave massive MIMO systems with low-resolution 
ADCs [13], [40], [57]. In this subsection, we investigate the 
LDGEC-based wideband beamspace channel estimation with 
low-resolution ADCs.

To improve the robustness of LDGEC network in quantized 
systems, we use the damping method presented in [41] where 
damping factor \( \beta = 0.1^t \) is exponentially decreased. Fig. 9(a) 
comparing the performance of the LDGEC channel estimator 
with low-resolution ADCs. From the figure, the LDGEC 
channel estimator with two-bit ADCs outperforms the SSD 
algorithm with infinite-bit ADCs when SNR < 10 dB. There-
fore, the LDGEC channel estimator can accurately estimate 
the channel from the quantized signal, thereby reducing the 
hardware cost of the systems.

Fig. 9(b) illustrates the performance of the LDGEC network 
with SURE loss and layer-by-layer training. From the figure, 
we have similar conclusions to that of LDGEC with denoiser-
denoiser training.
with low-resolution ADC architecture even without real channel data.

VI. CONCLUSION

We have developed a novel model-driven unsupervised DL network for wideband mmWave beamspace channel estimation. This network inherits the superiority of iterative signal recovery algorithms and the advanced DL-based denoiser, and thus presents excellent performance. The LDGEC network is easy to train and can be applied to a variety of selection networks. Furthermore, by utilizing the SURE loss, the LDGEC network can be trained without real channel data, enables the system to apply in a new environment. Simulation results demonstrate that the LDGEC-based channel estimator significantly outperforms state-of-the-art CS-based algorithms even for the receiver is equipped with a small number of RF chains and low-resolution ADCs. For future work, it will be interesting to apply the SURE technology to other DL-based wireless communication applications, such as CSI feedback and hybrid beamforming problems, to achieve unsupervised learning.

REFERENCES

[1] A. L. Swindlehurst, E. Ayanoglu, P. Heydari, and F. Capolino, “Millimeter-wave massive MIMO: The next wireless revolution?” IEEE Commun. Mag., vol. 52, no. 9, pp. 56–62, Sep. 2014.

[2] T. Bai and R. W. Heath, Jr., “Coverage and rate analysis for millimeter-wave cellular networks,” IEEE Trans. Wireless Commun., vol. 14, no. 2, pp. 1100–1114, Feb. 2015.

[3] R. W. Heath, N. González-Prelcic, S. Rangan, W. Roh, and A. M. Sayeed, “An overview of signal processing techniques for millimeter wave MIMO systems,” IEEE J. Sel. Topics Signal Process., vol. 10, no. 3, pp. 436–453, Apr. 2016.

[4] J. Brady, N. Behdad, and A. Sayeed, “Beamspaces MIMO for millimeterwave communications: System architecture, modeling, analysis, and measurements,” IEEE Trans. Antennas Propag., vol. 61, no. 7, pp. 3814–3827, Jul. 2013.

[5] A. Alkhateeb, O. El Ayach, G. Leus, and R. W. Heath, Jr., “Channel estimation and hybrid precoding for millimeter wave cellular systems,” IEEE J. Sel. Topics Signal Process., vol. 8, no. 5, pp. 831–846, Oct. 2014.

[6] K. Venugopal, A. Alkhateeb, N. González-Prelcic, and R. W. Heath, Jr., “Channel estimation for hybrid architecture-based wideband millimeter wave systems,” IEEE J. Sel. Areas Commun., vol. 35, no. 9, pp. 1996–2009, Sep. 2017.

[7] X. Gao, L. Dai, S. Han, I. Chih-Lin, and X. Wang, “Reliable beamspace channel estimation for millimeter-wave massive MIMO systems with lens antenna array,” IEEE Trans. Wireless Commun., vol. 16, no. 9, pp. 6010–6021, Sep. 2017.

[8] S. H. Mirfarshbafan et al., “Beamspaces channel estimation for massive MIMO mmWave systems: Algorithm and VLSI design,” IEEE Trans. Circuits Syst. I, Reg. Papers, vol. 67, no. 12, pp. 5482–5495, Dec. 2020.

[9] L. Yang, C.-K. Wen, S. Jin, and F. Gao, “Beamspace channel estimation in mmWave systems via cosparsse image reconstruction technique,” IEEE Trans. Commun., vol. 66, no. 10, pp. 4767–4782, Oct. 2018.

[10] J. Hogan and A. Sayeed, “Beam selection for performance-complexity optimization in high-dimensional MIMO systems,” in Proc. Annu. Conf. Inf. Sci. Syst. (CISS), Mar. 2016, pp. 337–342.

[11] Y. Zeng and R. Zhang, “Millimeter wave MIMO with lens antenna array: A new path division multiplexing paradigm,” IEEE Trans. Commun., vol. 64, no. 4, pp. 1557–1571, Apr. 2016.

[12] L. Yang, Y. Zeng, and R. Zhang, “Efficient channel estimation for millimeter wave MIMO with limited RF chains,” in Proc. IEEE Int. Conf. Commun., May 2016, pp. 1–6.

[13] J. Mo, P. Schniter, and R. W. Heath, Jr., “Channel estimation in broadband millimeter wave MIMO systems with few-bit ADCs,” IEEE Trans. Signal Process., vol. 66, no. 5, pp. 1141–1154, Mar. 2018.

[14] F. Bellili, F. Sohrabi, and W. Yu, “Generalized approximate message passing for massive MIMO mmWave channel estimation with Laplacian prior,” IEEE Trans. Commun., vol. 67, no. 5, pp. 3205–3219, May 2019.

[15] H. He, C.-K. Wen, S. Jin, and G. Y. Li, “Deep learning-based channel estimation for beamspace mmWave massive MIMO systems,” IEEE Wireless Commun. Lett., vol. 7, no. 5, pp. 852–855, Oct. 2018.

[16] P. Dong, H. Zhang, G. Y. Li, I. Gaspar, and N. NaderiAlizadeh, “Deep CNN-based channel estimation for mmWave massive MIMO systems,” IEEE J. Sel. Topics Signal Process., vol. 13, no. 5, pp. 989–1000, Sep. 2019.

[17] B. Wang, F. Gao, S. Jin, H. Lin, and G. Y. Li, “Spatial-and frequency wideband effects in millimeter-wave massive MIMO systems,” IEEE Trans. Signal Process., vol. 66, no. 13, pp. 3393–3406, Jul. 2018.

[18] B. Wang et al., “Spatial-wideband effect in massive MIMO with application in mmWave systems,” IEEE Commun. Mag., vol. 56, no. 12, pp. 134–141, Dec. 2018.

[19] X. Gao, L. Dai, S. Zhou, A. M. Sayeed, and L. Hanzo, “Wideband beamspace channel estimation for millimeter-wave massive MIMO systems relying on lens antenna arrays,” IEEE Trans. Signal Process., vol. 67, no. 18, pp. 4809–4824, Sep. 2019.

[20] Z. Gao, C. Hu, L. Dai, and Z. Wang, “Channel estimation for millimeterwave massive MIMO with hybrid precoding over frequency-selective fading channels,” IEEE Commun. Lett., vol. 20, no. 6, pp. 1259–1262, Jun. 2016.

[21] R. Wang, H. He, S. Jin, X. Wang, and X. Hou, “Channel estimation for millimeter wave massive MIMO systems with low-resolution ADCs,” in Proc. IEEE 20th Int. Workshop Signal Process. Adv. Wireless Commun. (SPAWC), Jul. 2019, pp. 1–5.

[22] B. Wang, M. Jian, F. Gao, G. Y. Li, and H. Lin, “Beam squint and channel estimation for wideband mmWave massive MIMO-OFDM systems,” IEEE Trans. Signal Process., vol. 67, no. 23, pp. 5893–5908, Dec. 2019.

[23] Z. Qin, H. Ye, G. Y. Li, and B. H. F. Juang, “Deep learning in physical layer communications,” IEEE Wireless Commun., vol. 26, no. 2, pp. 93–99, Apr. 2019.
HE et al.: BEAMSPACE CHANNEL ESTIMATION FOR WIDEBAND MILLIMETER-WAVE MIMO

[24] H. He, S. Jin, C.-K. Wen, F. Gao, G. Y. Li, and Z. Xu, “Model-driven deep learning for physical layer communications,” IEEE Wireless Commun., vol. 26, no. 5, pp. 77–83, Oct. 2019.

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

[26] H. Ye, G. Y. Li, and B.-H. Juang, “Power of deep learning for channel estimation and signal detection in OFDM systems,” IEEE Wireless Commun. Lett., vol. 7, no. 1, pp. 114–117, Feb. 2018.

[27] H. He, C.-K. Wen, S. Jin, and G. Y. Li, “Model-driven deep learning for MIMO detection,” IEEE Trans. Signal Process., vol. 68, pp. 1702–1715, 2020.

[28] C. Qi, P. Dong, W. Ma, H. Zhang, Z. Zhang, and G. Y. Li, “Acquisition of channel state information for mmWave massive MIMO: Traditional and on-line training-based approaches,” Sci. China Inf. Sci., vol. 64, no. 8, pp. 1–16, Aug. 2021.

[29] W. Xia, G. Zheng, Y. Zhu, J. Zhang, J. Wang, and A. Petropulu, “A deep learning framework for optimization of MISO downlink beamforming,” IEEE Trans. Commun., vol. 68, no. 3, pp. 1866–1880, Mar. 2020.

[30] G. Aceto, D. Ciuonzo, A. Montieri, and A. Pescapé, “Mobile encrypted traffic classification using deep learning: Experimental evaluation, lessons learned, and challenges,” IEEE Trans. Netw. Service Manag., vol. 14, no. 2, pp. 485–498, Jun. 2019.

[31] A. Rago, G. Piro, G. Boggia, and P. Dini, “Multi-task learning at the mobile edge: An effective way to combine traffic classification and prediction,” IEEE Trans. Veh. Technol., vol. 69, no. 9, pp. 10362–10374, Sep. 2020.

[32] G. Aceto et al., “MIMETIC: Mobile encrypted traffic classification using multimodal deep learning,” Comput. Netw., vol. 165, Dec. 2019, Art. no. 106944.

[33] F. Ait Aoudia and J. Hoydis, “End-to-end learning of communications systems without a channel model,” Dec. 2018, arXiv:1804.02276.

[34] H. Ye, L. Liang, G. Y. Li, and B.-H. Juang, “Deep learning based end-to-end wireless communications systems with GAN as unknown channel,” IEEE Trans. Wireless Commun., vol. 19, no. 5, pp. 3133–3143, May 2020.

[35] J. R. Hershey, J. Le Roux, and F. Weninger, “Deep unfolding: Model-based inspiration of novel deep architectures,” Sep. 2014, arXiv:1409.2574.

[36] D. Ito, S. Takabe, and T. Wadayama, “Trainable ISTA for sparse signal recovery,” IEEE Trans. Sig. Process., vol. 67, no. 12, pp. 3113–3125, Jun. 2019.

[37] V. Monga, Y. Li, and Y. C. Eldar, “Algorithm unrolling: Interpretable, efficient deep learning for signal and image processing,” 2019, arXiv:1912.04533.

[38] A. Balatsoukas-Stimming and C. Studer, “Deep unfolding for communications systems: A survey and some new directions,” 2019, arXiv:1906.05774.

[39] H. He, C.-K. Wen, and S. Jin, “Generalized expectation consistent signal recovery for nonlinear measurements,” in Proc. IEEE Int. Symp. Inf. Theory (ISIT), Jun. 2017, pp. 2333–2337.

[40] H. He, C.-K. Wen, and S. Jin, “Bayesian optimal data detector for hybrid mmWave MIMO-OFDM systems with low-resolution ADCs,” IEEE J. Sel. Topics Signal Process., vol. 12, no. 3, pp. 469–483, Jun. 2018.

[41] C. Wang, C.-K. Wen, S. Tsai, and S. Jin, “Decentralized expectation consistent signal recovery for phase retrieval,” IEEE Trans. Signal Process., vol. 68, pp. 1484–1499, 2020.

[42] A. A. M. Saleh and R. A. Valenzuela, “A statistical model for indoor multipath propagation,” IEEE J. Sel. Areas Commun., vol. SAC-5, no. 2, pp. 128–137, May 1987.

[43] S. M. Perera, A. Madanayake, A. Ogle, D. Silverio, and J. Q. Huang, “A fast algorithm to solve delay Vandermonde systems in phased-array digital receivers,” IEEE Trans. Aerosp. Electron. Syst., vol. 57, no. 4, pp. 2288–2297, Aug. 2021.

[44] S. M. Perera, A. Madanayake, and R. Cintra, “Efficient and self-reconfigurable delay Vandermonde algorithm for multi-beam antenna arrays,” IEEE Open J. Signal Process., vol. 1, pp. 64–76, 2020.

[45] S. M. Perera et al., “Wideband N-beam arrays with low-complexity algorithms and mixed-signal integrated circuits,” IEEE J. Sel. Topics Signal Process., vol. 12, no. 2, pp. 368–382, May 2018.

[46] S. M. Perera, A. Madanayake, and R. J. Cintra, “Radix-2 self-reconfigurable sparse factorizations of delay Vandermonde matrices for wideband multi-beam antenna arrays,” IEEE Access, vol. 8, pp. 25498–25508, 2020.

[47] C. K. Wen, C. J. Wang, S. Jin, K. K. Wong, and P. Ting, “Bayes-optimal joint channel-and-data estimation for massive MIMO with low-precision ADCs,” IEEE Trans. Signal Process., vol. 64, no. 10, pp. 2541–2556, May 2016.

[48] K. Zhang, W. Zuo, Y. Chen, D. Meng, and L. Zhang, “Beyond a Gaussian denoiser: Residual learning of deep CNN for image denoising,” IEEE Trans. Image Process., vol. 16, no. 9, pp. 3142–3155, Jul. 2017.

[49] H. D. Pfister. (2014). Compressed Sensing Supplemental Material for Graphical Models and Inference. [Online]. Available: http://pfister.ee.duke.edu/courses/ece59010/cs.pdf

[50] C. Tian, L. Fei, W. Zheng, Y. Xu, W. Zuo, and C.-W. Lin, “Deep learning on image denoising: An overview,” 2019, arXiv:1912.13171.

[51] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit., Jun. 2016, pp. 770–778.

[52] C. M. Stein, “Estimation of the mean of a multivariate normal distribution,” Ann. Statist., vol. 9, no. 6, pp. 1135–1151, Nov. 1981.

[53] C. A. Metzler, A. Mousavi, R. Heckel, and R. G. Baraniuk, “Unsupervised learning with Stein’s unbiased risk estimator,” 2018, arXiv:1805.10551.

[54] C. A. Metzler, A. Mousavi, and R. G. Baraniuk, “Learned D-AMP: Principled neural network based compressive image recovery,” 2017, arXiv:1704.06625.

[55] I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin, and A. Courville, “Improved training of Wasserstein GANs,” 2017, arXiv:1704.00028.

[56] H.-S. Lee and C. G. Sodini, “Analog-to-digital converters: Digitizing the analog world,” Proc. IEEE, vol. 96, no. 2, pp. 323–334, Feb. 2008.

[57] J. Zhang, L. Dai, Z. He, S. Jin, and X. Li, “Performance analysis of mixed-ADC massive MIMO systems over Rician fading channels,” IEEE J. Sel. Areas Commun., vol. 35, no. 6, pp. 1327–1338, Jun. 2017.
Shi Jin (Senior Member, IEEE) received the B.S. degree in communications engineering from Guilin University of Electronic Technology, Guilin, China, in 1996, the M.S. degree from Nanjing University of Posts and Telecommunications, Nanjing, China, in 2003, and the Ph.D. degree in information and communications engineering from the Southeast University, Nanjing, in 2007. From June 2007 to October 2009, he was a Research Fellow at the Adastral Park Research Campus, University College London, London, U.K. He is currently a Faculty Member with the National Mobile Communications Research Laboratory, Southeast University. His research interests include space time wireless communications, random matrix theory, and information theory. He serves as an Associate Editor for the IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS, IEEE COMMUNICATIONS LETTERS, and IET Communications. He and his coauthors have been awarded the 2011 IEEE Communications Society Stephen O. Rice Prize Paper Award in the field of communication theory and the 2010 Young Author Best Paper Award from the IEEE Signal Processing Society.

Chao-Kai Wen (Senior Member, IEEE) received the Ph.D. degree from the Institute of Communications Engineering, National Tsing Hua University, Taiwan, in 2004. He was with the Industrial Technology Research Institute, Hsinchu, Taiwan, and MediaTek Inc., Hsinchu, Taiwan, from 2004 to 2009. Since 2009, he has been with National Sun Yat-sen University, Taiwan, where he is a Professor with the Institute of Communications Engineering. His research interests include optimization in wireless multimedia networks.

Geoffrey Ye Li (Fellow, IEEE) is currently a Chair Professor with Imperial College London, U.K. He was a Professor at the Georgia Institute of Technology, USA, for 20 years and a Principal Technical Staff Member at AT&T Laboratories-Research in New Jersey, USA, for five years. He has published over 600 journals and conference papers in addition to over 40 granted patents and several books in his research areas. His publications have been cited over 50,000 times with an H-index over 100 and he has been recognized as a Highly Cited Researcher, by Thomson Reuters, almost every year. His general research interests include statistical signal processing and machine learning for wireless communications. He was awarded IET fellow for his contributions to signal processing for wireless communications. He was a recipient of several prestigious awards, including the Donald G. Fink Overview Paper Award from IEEE Signal Processing Society in 2017; the James Evans Avant Garde Award in 2013 and Jack Neubauer Memorial Award in 2014 from IEEE Vehicular Technology Society; and the Stephen O. Rice Prize Paper Award in 2013, the Award for Advances in Communication in 2017, and the Edwin Howard Armstrong Achievement Award in 2019 from IEEE Communications Society. He also received the 2015 Distinguished ECE Faculty Achievement Award from Georgia Tech. He has organized and chaired many international conferences, including the Technical Program Vice-Chair of the IEEE ICC’03 and the General Co-Chair of the IEEE GlobalSIP’14, the IEEE VTC’19, Fall, and the IEEE SPAWC’20. He has been involved in editorial activities for over 20 technical journals, including the Founding Editor-in-Chief of IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS (JSAC) Special Series on ML in Communications and Networking.