Signal Processing 239 (2026) 110180

Contents lists available at ScienceDirect

Signal Processing

journal homepage: www.elsevier.com/locate/sigpro

Compressive sensing based downlink channel estimation for mmWave systems using deep learning: Centralized or decentralized

Usman Aslam, Rukhsana Ruby, Dian Zhang, Kaishun Wu, Lu Wang *,1

School of Computer Science and Software Engineering, Shenzhen University, Nanshan Yuehai Campus, Shenzhen, 518060, Guangdong, China

ARTICLE INFO

Keywords:
Compressive sensing
Channel estimation for mmWave massive MIMO systems
Deep learning
Federated learning

ABSTRACT

Thanks to the availability of abundant bandwidth, millimeter-wave (mmWave) massive multi-input multi-output (MIMO) systems have numerous applications in wireless communications, in which downlink channel estimation is one of the crucial problems. Through the adaptation of lens antenna array, mmWave channels have a sparsity feature in the beamspace domain, which results in the requirement of a reduced number of RF chains for such a system. Using the sparsity of beamspace channels, the beamspace channel estimation problem can be formulated as a sparse signal recovery problem, which can be solved by the greedy OMP algorithm, where dominant beamspace channel entries are found sequentially one by one, iteratively. To improve the performance of this algorithm, we propose a deep learning (DL)-based OMP algorithm, namely LOMP, in which a previously trained DL model is adopted to choose a set of beamspace dominant entries simultaneously instead of a single dominant entry. As we consider the downlink channel estimation, constructing a DL model well-accepted by all users requires them to send received signals to a centralized entity with high computational power. However, this incurs huge communication burden owing to the signal transmission by all users. To tackle this issue of communication burden as well as preserve the privacy of received signals, we propose a decentralized federated learning (FL)-based OMP algorithm, namely FL-OMP, in which the peer-to-peer (P2P) FL technique is adopted to construct the DL model in the LOMP algorithm, where users share gradients of their local learning models among themselves. Extensive simulations have been conducted to verify the effectiveness of the proposed LOMP and FL-OMP algorithms while comparing with other works in the existing literature.

1. Introduction

Millimeter wave (mmWave) is one of the promising communication technologies in 5G and the forthcoming 6G system due to the availability of abundant bandwidth [1]. Owing to the placement at the high-frequency domain, mmWave channels suffer high path loss, which can be mitigated by equipping beamforming technologies, such as a massive multi-input multi-output (MIMO) array at the transmitter [2]. The phase shifter network or lens antenna array is a well-known technology to convert the spatial channels of such a system to beamspace channels [3]. Thanks to the nature of high-frequency mmWave channels, there are only a few dominant propagation paths with large gain, which causes sparsity in the beamspace channel [4]. Consequently, the radio frequency (RF) chain requirements connected to the digital baseband of such a system can be considerably reduced through the selection of dominant beams. The selection of appropriate beams requires the accurate channel state information (CSI) in the beamspace domain [5], which is computationally intensive because of the high channel dimension, especially when the system has much less number of RF chains compared to the number of antennas.

At the beginning, researchers used least-square (LS) [6,7] and pilot-based techniques [8] to acquire CSI from wireless channels. Due to the pilot contamination issue owing to the exchange of huge volume of pilot signal, the accuracy as well as the system efficiency [9,10] of pilot-based techniques is not good. Consequently, compressive-sensing (CS)-based techniques allow for accurate channel estimation with fewer pilot symbols [11] by exploiting the sparsity of the channel which is beneficial for high-frequency channel with sparse representation [12]. Based on different structures, CS has different algorithms for channel estimation [13]. For example, Orthogonal Matching Pursuit (OMP) is a well-known CS-based greedy algorithm [14], which is computationally efficient and effective for real-time applications. Approximate

* Corresponding author.
E-mail address: wanglu@szu.edu.cn (L. Wang).

[1] The research was supported in part by the China NSFC Grant (No. 62372307, No. U2001207), Guangdong NSF (No. 2024A1515011691), Shenzhen Science and Technology Program (No. RCYX20231211090129039), Shenzhen Science and Technology Foundation (No. JCYJ20230808105906014), Guangdong Provincial Key Lab of Integrated Communication, Sensing and Computation for Ubiquitous Internet of Things (No. 2023B1212010007), the Project of DEGP (No. 2023KCXTD042).

https://doi.org/10.1016/j.sigpro.2025.110180
Received 23 December 2024; Received in revised form 20 May 2025; Accepted 17 June 2025
Available online 29 July 2025

Message Passing (AMP) is also a CS powerful technique used for channel estimation [15]. This is an iterative sparse channel recovery technique [16], particularly where sparsity is a defining characteristic [17]. Sparse Bayesian Learning (SBL) is a CS-based algorithm to help estimate sparse channels in wireless systems. [18] proposes a modified off-grid SBL that directly estimates continuous parameters, boosting accuracy and efficiency for large-scale Reconfigurable Intelligent Surfaces (RIS) setups but requires active sensors at the RIS, increasing system complexity and cost. On the other hand, the approach multi-layer SBL [19] estimates mmWave channels by starting with a low-resolution estimate and iteratively refining this using updated dictionaries and beam selections to reduce computational demands for real-time applications. A variational Bayesian (VB) [20] learning approach was introduced for multi-user mmWave systems to enhance robustness against hardware impairments by leveraging hierarchical priors and supporting set sharing across subcarriers. In another direction, a deep unfolding method, learned iterative shrinkage and thresholding algorithm (LISTA) [21], and its optimized version sensing matrix optimization (LISTA-SMO) [22], were applied to dynamic metasurface antenna (DMA)-based systems, demonstrating improved estimation performance by learning both the algorithmic structure and the sensing matrix. Additionally, an alternating direction method of multiplier (ADMM) CS-based [23] sparse recovery algorithm was proposed for underwater acoustic channels to enhance resilience against impulsive noise through robust optimization. Despite the requirement of fewer pilot signals, the accuracy of these techniques is not well-fit for the advanced applications in wireless communications. Consequently, there is a pressing need for a more robust channel estimation technique that can reliably estimate the dominant beamspace channel entries while avoiding the pitfalls of sequential estimation [24].

Recently, machine learning (ML) has received widespread attention due to its effectiveness in almost every sector of our lives [25]. The study referenced [26] was the first to apply Deep Learning (DL) technique for channel estimation in wireless energy transfer systems, aiming to enhance the accuracy of channel estimation in multi-carrier wireless communication environments. In [27], the authors used a Deep Neural Network (DNN) for joint pilot and data-aided channel estimation, where the pilot-aided estimation process is implemented using a 2-layer neural network (NN) combined with a DNN, while another DNN is utilized for the data-aided estimation process. DL offers a transformative scheme by modeling complex patterns effectively. Recent advancements in end-to-end DL frameworks enhance channel estimation and offer a groundbreaking alternative of using NN to process received signals and directly recover transmitted data. In [28], the authors proposed a hybrid network integrating pilot design and channel estimation in MIMO-OFDM systems, improving accuracy with fewer pilots. [29] innovatively models the time-frequency response of channels as a 2D image, applying super-resolution denoising techniques to achieve remarkable estimation accuracy. [30] utilized a super-resolution algorithm to refine LS estimation in mine settings to achieve MMSE precision with reduced pilot overhead. In [31], the authors proposed a super-resolution channel estimation scheme based on DL for mmWave massive MIMO systems, which utilizes a DNN to estimate the direction-of-arrival (DOA). By utilizing spatial correlation among different antennas [32], the DL model refines the estimated supports, leading to better overall performance compared to traditional methods. Hybrid model-based approaches [33–35] adopted optimization algorithms to renovate a DNN as well as simple NN, respectively, by tuning their parameters to improve data communication performance via accurate channel estimation. DL continued to dominate, allowing for more accurate channel estimation by leveraging large datasets and complex patterns without the need for explicit mathematical models [36]. In [37], NN is leveraged to improve the accuracy and efficiency of estimating CSI, which is a crucial task in ensuring reliable and efficient data transmission. In all these data-driven works, it is shown that the channel estimation performance is much better with the DL techniques compared to the conventional algorithms [38] at the cost of huge computational complexity. The application of CS techniques, such as OMP and AMP for channel estimation in mmWave massive MIMO systems, has been extensively studied [39]. While these methods have demonstrated success in leveraging the sparse nature of mmWave channels [40], they are often limited by their sequential and greedy approach, which can lead to suboptimal results due to the lack of global optimality. As the complexity of communication environments gradually increases, joint CS and DL techniques emerge. For example, the hybrid model-based DL approach [33] combines traditional CS-based estimation techniques with DL models to improve channel estimation performance. Following this strategy, another two works are [17,41], in which a model-driven DL technique is integrated with the AMP algorithm to learn the intermediate variables of the iterative procedure for mmWave and Tera-Herz (THz) communications, respectively. Most of these model-driven CS-based channel estimation techniques are suitable for the uplink direction, and our work focuses on the downlink channel estimation in mmWave systems. In [42], a predefined DL model as well as the essence of CS-based OMP algorithm are combined to estimate channel in the high-frequency domain. Similar to [42], we leverage the OMP algorithm integrated with a pre-trained DL model to design a channel estimation scheme for mmWave massive MIMO systems. However, the complexity of the DL model of [42] is much higher compared to ours due to different design of output vector. On the other hand, [42] is ignorant of the residual concept, and thus its achieved channel estimation accuracy is much lower as the sparsity level of the channel increases. Furthermore, their formulation for channel estimation is for the uplink direction, and we have considered the downlink one in this paper.

In our proposed CS-based channel estimation algorithm, a DL model is required that is trained offline by the collected received signals in a centralized entity. In order to deal with high communication overhead of data collection, recently federated learning (FL) schemes, including the centralized and decentralized peer-to-peer (P2P)-based, have received widespread attention [43]. In the decentralized P2P-based FL, users only exchange the model updates, i.e., gradients of the model parameters, and thus the communication overhead is considerably reduced. Motivated by the applications of FL in the resource management for wireless sensor networks [44], the trajectory optimization for UAV networks [45], task optimization of vehicular networks [46] and beamforming design for massive MIMO systems [47], researchers have considered the estimation of high-frequency channels, such as mmWave and THz, using FL. For example, in [48,49], the authors have proposed channel estimation schemes using FL for mmWave massive MIMO systems. In the similar manner, the channel estimation for THz communications is considered in [22], and in [50,51], the same problem has been studied for RIS-aided communication systems and cell-free massive MIMO systems, respectively. All these works estimate channels based on the raw received signals. However, in [52], the authors proposed a model-driven learning approach using FL, in which intermediate variables of some CS-based algorithm (i.e., AMP) are learned using FL. Similar to this work, we also learn our designed DL model using the P2P-based decentralized FL technique for the sake of our proposed iterative channel estimation algorithm.

Despite the availability of some works on the DL-inspired CS-based channel estimation works for mmWave massive MIMO systems, a thorough study is still required to justify their effectiveness in terms of performance, computation and communication overhead. To this end, we propose a DL-inspired OMP-based downlink channel estimation scheme, namely LOMP, for mmWave massive MIMO systems aiming to achieve higher performance with low computational complexity compared to existing schemes (e.g., DLCS [42]). The structure of the LOMP algorithm is as same as that in the conventional OMP algorithm with the modification of the selection of the dominant entries in the beamspace channel, which is conducted using the previously trained DL model instead of the correlation function. This DL model is trained

in an offline environment, for which the receivers are required to send their received signals to a centralized entity, and thus incurs high communication overhead. To mitigate this communication overhead as well as to preserve the privacy of received signals at each individual user, we employ the FL technique to train the DL model, and we name the resultant channel estimation scheme as FL-OMP. To summarize, the contributions of this work are listed as follows.

- Leveraging the idea of CS-based greedy OMP algorithm and DL technique, we propose an iterative CS-based downlink channel estimation scheme, namely LOMP, for mmWave massive MIMO systems. The structure of each iteration is as same as that in the OMP algorithm. However, the proposed algorithm chooses a set of dominant beamspace channel indexes through a DL model, which was trained offline beforehand. Although the concept of DLCS [42] has somewhat similarity with ours, this algorithm suffers high computational complexity in the offline training as the output of the DL model is the amplitude of the entire channel vector (i.e., the length of which is N), whereas the output of our DL model is only a set of indexes of the actual channel vector, the cardinality of which is at most κ ($\kappa \ll N$). On the other hand, DLCS is ignorant of the residual concept after one iteration, and the algorithm requires just only one iteration. However, with the increasing sparsity of the beamspace channel, this algorithm suffers significant performance degradation, and we overcome this in our proposed LOMP scheme.
- In the LOMP algorithm, the DL model is constructed in a centralized manner, for which the receivers are required to send the received signals to a centralized entity, which incurs high communication overhead. Consequently, to mitigate this communication overhead issue as well as to preserve the privacy of user-specific pilot signals, we propose a FL-based decentralized CS-inspired channel estimation scheme, namely FL-OMP, in which the receivers construct a DL model with their individual dataset and then exchange the gradients of the DL model among themselves through the P2P networking idea.
- Extensive simulations have been conducted under various realistic settings to justify the effectiveness of our proposed channel estimation scheme compared to DLCS and OMP algorithm. The results demonstrate that our proposed LOMP algorithm outperforms the DLCS one with much lower computational complexity. On the other hand, FL-OMP achieves close performance to the LOMP one with much lower communication overhead.

The rest of the paper is organized as follows. Followed by the system description and problem formulation in Section 2, we propose the solution for downlink channel estimation of mmWave massive MIMO systems based on both the centralized and decentralized concepts in Section 3. We evaluate the performance of the proposed channel estimation schemes in Section 4. Finally, we draw the conclusion of the paper in Section 5.

The anatomy of the notations presented in this paper is described as follows. a is a vector, $\boldsymbol{a}$ is a vector, and $\boldsymbol{A}$ is a matrix. Moreover, $\boldsymbol{A}^T$ is the transpose of the matrix $\boldsymbol{A}$, and $\boldsymbol{A}^H$ is the conjugate transpose of the matrix $\boldsymbol{A}$. On the other hand, $< \boldsymbol{a}, \boldsymbol{b} >$ is the correlation function between the vectors $\boldsymbol{a}$ and $\boldsymbol{b}$, and $\mathbf{DL}(\boldsymbol{a})$ is the output of the DL model given the input vector $\boldsymbol{a}$.

2. System model and problem formulation

Followed by the detailed description of the multi-user mmWave massive MIMO systems, in this section, we formulate the channel estimation problem as a CS problem to estimate the sparse channel in the beamspace domain. In Table 1, we summarize the mathematical nottaions used in this paper.

2.1. System model

We consider downlink multi-user mmWave massive MIMO communication system, in which the base station (BS) consists of N_{BS} antennas and N_{RF} RF chains. There are U users in the system served by the BS, while each user is equipped with a single antenna. The antennas in the BS have a uniform linear array (ULA) structure with d inter-antenna spacing. We employ a hybrid analog-digital architecture at the BS that operates in a frequency-selective multi-path environment with single-antenna users, where the number of RF chains is significantly fewer than the number of antennas (i.e., $N_{\mathrm{RF}} \ll N_{\mathrm{BS}}$). In case the number of users U is larger than N_{RF}, the BS only turns on U RF chains while turning the remaining $U - N_{\mathrm{RF}}$ off in order to save the power consumption. The BS performs hybrid precoding along the downlink direction, which is essentially the multiplexing operation between the baseband and analog domains. U data streams for digital precoding can be transmitted from the BS to the users. The received signal of all U users, denoted by $y \in \mathbb{C}^U$, can be represented as

$$\boldsymbol{y} = \boldsymbol{H}\boldsymbol{F}_{\mathrm{RF}}\boldsymbol{F}_{\mathrm{BB}}\boldsymbol{s} + \boldsymbol{n}, \tag{1}$$

where $\boldsymbol{F}_{RF} \in \mathbb{C}^{N_{BS} \times U}$ denotes the analog precoder and $\boldsymbol{F}_{\mathrm{BB}} \in \mathbb{C}^{U \times U}$ is the digital precoder. To satisfy the power constraint of the hybrid precoder, we set $\|\boldsymbol{F}_{\mathrm{RF}}\boldsymbol{F}_{\mathrm{BB}}\|_F = U$. $\boldsymbol{s} \in \mathbb{C}^U$ is the signal vector, for which $\mathbb{E}\{\boldsymbol{s}\boldsymbol{s}^H\} = \boldsymbol{I}_U$ holds, and $\boldsymbol{n} \in \mathbb{C}^U$ is the additive white Gaussian noise (AWGN) vector satisfying $\boldsymbol{n} \sim \mathcal{CN}(0, \sigma^2 \boldsymbol{I}_U)$. The channel matrix from the BS to U users is denoted by

$$\boldsymbol{H} \triangleq \left[\boldsymbol{h}_1, \ldots, \boldsymbol{h}_U\right]^T \in \mathbb{C}^{U \times N_{\mathrm{BS}}}. \tag{2}$$

2.2. Channel model

In this section, we discuss the mmWave channel model used for multi-user communications in massive MIMO system, where each user has a mmWave channel vector, $\mathbf{h}_u \in \mathbb{C}^{N_{\mathrm{BS}} \times 1}$, which follows the Saleh-Valenzuela (SV) model [42,53] and is sparse in the beamspace domain due to the limited number of propagation paths and the ability to capture the sparse multipath characteristics of the environment. Let us assume that the number of propagation paths for user u is L_u, $g_{u,i}$ is the complex gain of the ith path, which follows the complex Gaussian distribution with zero mean and unit variance. Moreover, $\boldsymbol{a}_T(N_{BS}, \theta_{u,i})$ is the steering vector at the BS corresponding to the angle-of-arrival (AoA) $\theta_{u,i}$. Thus, for user u, the corresponding channel from the BS is modeled as

$$\mathbf{h}_u = \sqrt{\frac{N_{\mathrm{BS}}}{L_u}} \sum_{i=1}^{L_u} g_{u,i}\alpha(N_{\mathrm{BS}}, \theta_{u,i}). \tag{3}$$

We consider the ULA antenna arrangement with the inter-antenna distance $d = \lambda/2$ (λ is the wave length), the steering vector $\boldsymbol{a}_T(N_{BS}, \theta_{u,i})$ for an antenna array with N_{BS} elements is given by

$$\boldsymbol{a}_T(N_{BS}, \theta_{u,i}) = \frac{1}{\sqrt{N_{BS}}}\left[1, e^{j\pi\sin(\theta)}, e^{j2\pi\sin(\theta)}, \ldots, e^{j(N_{BS}-1)\pi\sin(\theta)}\right]^T.$$

This vector model represents the phase shifts across the antenna elements corresponding to the angle θ. Each entry of the vector represents the phase difference across adjacent antennas for a signal arriving at an angle θ. The array steering vector effectively captures the spatial structure of the received signal and is the key to beamforming operations (see Table 1).

2.3. Problem formulation

In mmWave massive MIMO communication systems, efficient channel estimation is crucial and challenging due to the large number of antennas and sparse nature of mmWave channel in the beamspace domain. For example, the good estimation of $\boldsymbol{H}$ is required to design precoding matrices $\boldsymbol{F}_{\mathrm{RF}}$ and $\boldsymbol{F}_{\mathrm{BB}}$ aiming to achieve the best possible

Table 1
Definition of notations used in this paper.

Symbol	Definition	Symbol	Definition
N_{BS}	Number of antennas at the BS	N_{RF}	Number of RF chains at the BS
U	Number of users	$\mathbb{C}^{U}$	Complex vector space of dimension U
$\boldsymbol{y}$	Received signal vector of all users	$\boldsymbol{H}$	Downlink channel matrix from BS to all users
$\boldsymbol{F}_{\mathrm{RF}}$	Analog precoding matrix	F_{BB}	Digital baseband precoding matrix
$\boldsymbol{s}$	Transmit signal vector	n	AWGN vector
$\mathbb{C}^{N_{\mathrm{BS}}\times U}$	Complex matrix of size $N_{\mathrm{BS}} \times U$	$\mathbb{C}^{U\times U}$	Complex matrix of size $U \times U$
$\mathbb{E}\{\boldsymbol{s}\boldsymbol{s}^{H}\}$	Expectation of signal covariance (identity matrix)	$\boldsymbol{I}_{U}$	$U \times U$ identity matrix
$\boldsymbol{n} \sim \mathcal{CN}(0, \sigma^2 \boldsymbol{I}_{U})$	Noise vector that follows complex normal distribution	$\boldsymbol{h}$	Channel vector for a single user
$\mathbb{C}^{U\times N_{\mathrm{BS}}}$	Complex matrix space for $U \times N_{\mathrm{BS}}$	$\mathbb{C}^{N_{\mathrm{BS}}\times 1}$	Channel vector dimension
L_u, $g_{u,i}$	Number of paths and complex gain of the ith path for user u	$\theta_{u,i}$	Angle of arrival of the ith path for user u
g_u	Composite gain for user u	λ	Wavelength
Λ	Index set of selected channel entries	$\boldsymbol{x}$	Optimization variable/estimated signal
$\boldsymbol{a}_T(N_{BS}, \theta_{u,i})$	Array steering vector	$e^{j\pi \sin(\theta)}$	Phase term in steering vector
$e^{j2\pi \sin(\theta)}$	Phase term in array vector	$e^{j(N_{BS}-1)\pi \sin(\theta)}$	Last term of steering vector
$\boldsymbol{P}$	Pilot matrix	M	Number of pilot transmissions
$\tilde{\boldsymbol{b}}$	Processed received signal after pilot projection	T	Number of iterations
$\boldsymbol{A}$	Beamspace dictionary matrix	$\boldsymbol{F}$	Concatenated precoding matrix
$\boldsymbol{\Phi}$	Measurement matrix in the beamspace	G	Grid resolution in the beamspace dictionary
κ	Number of dominant indices selected in each iteration	$\boldsymbol{r}$	Residual vector in the iterative algorithm
$\mathcal{L}$	Loss function for DL model	$\boldsymbol{\gamma}$	Ground truth output vector for DL model
D	Number of training samples	dl	Digital layer/delay variable
η	Learning rate for model training	Δ	Perturbation/noise term in FL update
∇	Gradient operator	β	Smoothness parameter in convergence analysis

performance. Given the sparsity of the mmWave channel, channel estimation can be formulated as a CS problem in the beamspace domain. Let denote the pilot matrix generated by the BS for all U users $\boldsymbol{P} \in \mathbb{C}^{U\times U}$, which is U mutually orthogonal pilot sequences, the length of each is U. For the downlink transmission, we adopt M different analog and digital precoding matrices, denoted by $\boldsymbol{F}^{m}_{\mathrm{RF}} \in \mathbb{C}^{N_{BS}\times U}$ and $\boldsymbol{F}^{m}_{\mathrm{BB}} \in \mathbb{C}^{U\times U}$, respectively, for $m = 1, \ldots, M$. The pilot signal received at the uth user for the mth transmission is given by

$$\boldsymbol{y}^{m}_{u} = \boldsymbol{h}^{T}_{u} \boldsymbol{F}_{\mathrm{RF}} \boldsymbol{F}_{\mathrm{BB}} \boldsymbol{P} + \boldsymbol{n}^{m}_{u}, \tag{4}$$

where $\boldsymbol{n}^{m}_{u}$ is the AWGN vector for the uth user at the mth pilot transmission phase, the distribution of which is $\mathcal{CN}(0, \sigma^2 \boldsymbol{I}_{U})$. Since each entry of the pilot matrix $\boldsymbol{P}$ is orthogonal, i.e., $\boldsymbol{P}\boldsymbol{P}^{H} = \boldsymbol{I}_{U}$, we multiply the received signal vector of the uth user $\boldsymbol{y}^{m}_{u}$ by $\boldsymbol{P}^{H}$, and thus the resultant received signal is given by

$$\tilde{\boldsymbol{b}}^{m}_{u} = \boldsymbol{y}^{m}_{u} \boldsymbol{P}^{H} = \boldsymbol{h}^{T}_{u}(\boldsymbol{F}^{m}) + \tilde{\boldsymbol{n}}^{m}_{u},$$

where

$$\boldsymbol{F}^{m} = \boldsymbol{F}^{m}_{\mathrm{RF}} \boldsymbol{F}^{m}_{\mathrm{BB}} \in \mathbb{C}^{N_{BS}\times U} \text{ and}$$
$$\tilde{\boldsymbol{n}}^{m}_{u} = \boldsymbol{n}^{m}_{u} \boldsymbol{P}^{H} \in \mathbb{C}^{1\times U}.$$

After the complete transmission of M pilot signals from the BS to all users, $\tilde{\boldsymbol{b}}^{m}_{u}$ can be stacked as

$$\boldsymbol{b}_{u} = [\tilde{\boldsymbol{b}}^{1}_{u}, \ldots, \tilde{\boldsymbol{b}}^{M}_{u}]^{T} = \boldsymbol{F}^{T} \boldsymbol{h}_{u} + \tilde{\boldsymbol{n}}_{u},$$

where

$$\boldsymbol{F} = [(\boldsymbol{F}^{1})^{T}, \ldots, (\boldsymbol{F}^{M})^{T}]^{T} \in \mathbb{C}^{N_{BS}\times UM} \text{ and}$$
$$\tilde{\boldsymbol{n}}_{u} = [\tilde{\boldsymbol{n}}^{1}_{u}, \ldots, \tilde{\boldsymbol{n}}^{M}_{u}]^{T} \in \mathbb{C}^{UM\times 1}.$$

Having notice that mmWave channels have sparsity nature in the beamspace domain, the beamspace channel vector is defined as

$$\mathbf{h}^{b}_{u} = \mathbf{A}^{\mathrm{H}} \mathbf{h}_{u},$$

where $\mathbf{A} \in \mathbb{C}^{N_{\mathrm{BS}}\times G}$ is the beamspace dictionary matrix with columns corresponding to steering vectors over a quantized AoA grid and this can be represented as

$$\mathbf{A} = \left[\mathbf{a}(\phi_1), \mathbf{a}(\phi_2), \ldots, \mathbf{a}(\phi_G)\right],$$

where $\phi_g = -1 + \dfrac{2(g-1)}{G}$ holds, over $g = 1, 2, \ldots, G$. Another characteristic of $\boldsymbol{A}$ is such that $\boldsymbol{A}^{H}\boldsymbol{A} = G\boldsymbol{I}_{N_{BS}}/N_{BS}$ holds. Consequently, after the consecutive transmission of the M pilot signals, the received signal at the uth user can be given by

$$\mathbf{b}_{u} = \frac{N_{\mathrm{BS}}}{G} \mathbf{F}^{T} \mathbf{A}^{H} \mathbf{h}^{b}_{u} + \tilde{\mathbf{n}}_{u}.$$

We further define

$$\boldsymbol{\Phi} = \frac{N_{BS}}{G} \boldsymbol{F}^{T} \boldsymbol{A}^{H} \in \mathbb{C}^{UM\times G}.$$

Consequently, the channel estimation problem for the uth user can be written $\boldsymbol{b}_{u} = \boldsymbol{\Phi}\boldsymbol{h}^{b}_{u} + \tilde{\boldsymbol{n}}_{u}$, which is essentially a sparse recovery problem. In practical environments, mmWave channels may not be sparse in the beamspace domain due to the beamspace power leakage. This happens when the true AoA of the signal paths do not align correctly with the discretized angular grid used to form the beamspace dictionary matrix $\boldsymbol{A}$. Furthermore, even if the angular grid is considered finite, the original signal paths may be interrupted between grid points, causing the energy of these paths to leak into multiple neighboring grid entries, resulting in non-zero values in entries that are expected to be zero. The dictionary matrix A is constrained by the number of grid points G, and higher resolution dictionaries with more grid points can reduce the sparsity impairment but at the cost of increased computational complexity. For practical systems with limited grid resolution, this trade-off leads to an approximation error in the sparse recovery process, which contributes to the sparsity impairment of $\mathbf{h}^{b}_{u}$.

3. Channel estimation proposal

Based on the idea of DL, which can be trained both in the centralized and decentralized manners, in this section, we propose two versions of channel estimation schemes for mmWave massive MIMO systems.

3.1. Centralized proposal

Upon the CS-based formulation of the channel estimation problem in the previous section, we notice that the dimension of the received signal vector is much lower than that of the channel vector. Despite this is a linear programming problem, which can be solved by the well-known convex optimization solutions, suffers huge computational complexity. Consequently, exploiting the sparsity of mmWave channel, in the existing literature, there are many algorithms with low complexity. One of such algorithms is OMP, the major steps of which are summarized in Algorithm 1. Omitting the user index as well as the beamspace superscript in $\boldsymbol{h}_u^b$, the algorithm provides the solution for the problem $\boldsymbol{b} = \boldsymbol{\Phi h}$. The algorithm runs in an iterative procedure, and the iterations continue until some intermediate variable $\boldsymbol{r}_k$, namely residual, reaches close to zero. There is another intermediate variable $\boldsymbol{\Lambda}$, that holds the selected indices of the channel vector, to be non-zero, over the iterations. At the initialization phase, this variable is set to $\emptyset$, whereas $\boldsymbol{r}_k$ is set to $\boldsymbol{b}$. Step 5 of the algorithm is crucial, which computes the correlation between $\boldsymbol{r}_{k-1}$ and each column vector (i.e., $\boldsymbol{\phi}_j$) of $\boldsymbol{\Phi}$. λ_k holds an index of the channel vector (i.e., $\boldsymbol{h}$) which provides the highest correlation. In step 6, $\boldsymbol{\Lambda}_k$ is updated with the previously obtained indices in the former iteration. With this new set $\boldsymbol{\Lambda}_k$, the convex optimization technique is applied over the linear programming problem $\boldsymbol{b} = \boldsymbol{\Phi}_{\boldsymbol{\Lambda}_k}\boldsymbol{x}$ to obtain the vector $\boldsymbol{x}$, the $\boldsymbol{\Lambda}_k$ indices of which are non-zero and the remaining entries are zero. Note that since only the $\boldsymbol{\Lambda}_k$ entries are considered non-zero, the resultant complexity of this convex optimization problem is considerably lower compared to the case when the sparsity assumption of the variable $\boldsymbol{x}$ is omitted. Subsequently, with the obtained solution $\boldsymbol{x}$, the residual of the original problem is updated in step 9. This process is continued until the residual reaches close to zero.

Algorithm 1 The conventional OMP algorithm.

1: **Input:** $\boldsymbol{\Phi}, \boldsymbol{b}$.
2: **Output:** $\boldsymbol{h}$.
3: **Initialization:** $k \leftarrow 1$, $\boldsymbol{r}_0 \leftarrow \boldsymbol{b}$ and $\boldsymbol{\Lambda}_0 \leftarrow \emptyset$.
4: **repeat**
5: $\quad \lambda_k \leftarrow \underset{j \notin \boldsymbol{\Lambda}_{k-1}}{\operatorname{argmax}} | < \boldsymbol{\phi}_j, \boldsymbol{r}_{k-1} > |$.
6: $\quad \boldsymbol{\Lambda}_k \leftarrow \boldsymbol{\Lambda}_{k-1} \cup \lambda_k$.
7: $\quad \boldsymbol{x}_k(i \in \boldsymbol{\Lambda}_k) \leftarrow \underset{\boldsymbol{x}}{\operatorname{argmin}} ||\boldsymbol{\Phi}_{\boldsymbol{\Lambda}_k}\boldsymbol{x} - \boldsymbol{b}||$, $\boldsymbol{x}_k(i \notin \boldsymbol{\Lambda}_k) = 0$.
8: $\quad \hat{\boldsymbol{b}}_k \leftarrow \boldsymbol{\Phi x}$.
9: $\quad \boldsymbol{r}_k \leftarrow \boldsymbol{b} - \hat{\boldsymbol{b}}_k$ and $k \leftarrow k + 1$.
10: **until** $||\boldsymbol{r}_k|| \approx 0$

In step 5 of this algorithm, we notice that λ_k is the single index of the channel vector. Since this algorithm chooses the non-zero indices of the channel vector one by one in a sequential manner, it is very likely that the algorithm traps in the local optima, and thus the optimal solution is not achievable. On the other hand, the number of iterations gradually increases with the increasing sparsity level of the problem. Intuitively, if we can select multiple dominant entries in step 5, there is less possibility that the algorithm gets stuck in local optima as well as it requires less iterations to reach the convergence. On the other hand, the concept of choosing indices based on the sole value of correlation is also greedy. In reality, the higher the correlation not necessarily the higher dominant entries. Sometimes, in order to avoid local optima, the algorithm may be required to choose the indices of the channel vector that correspond to a lower correlation. However, the identification of such indices is not straightforward. On the other hand, if we have sample datasets of $\boldsymbol{b}$, $\boldsymbol{\Phi}$ and $\boldsymbol{h}$, we can construct a DL model to provide the indices of the dominant entries in the channel vector to replace the step of 5 of the algorithm. Instead of one index, we propose a DL model that provides a number (say κ) of indices in the channel vector that are apparently dominant. This DL model is trained using the collected data from all the users. With the aid of the constructed DL model, we propose our own channel estimation scheme, namely LOMP, for mmWave systems. The schematic diagram of our LOMP algorithm is provided in Fig. 1. The components of the DL model as well as the features of input and output in each data sample are described as follows.

INPUT: The input vector corresponding to each sample of the DL model is the correlation between $\boldsymbol{\Phi}$ and $\boldsymbol{b}$ as follows.

$$\boldsymbol{c} = \boldsymbol{\Phi}^H \boldsymbol{b}, \quad \boldsymbol{c} \in \mathbb{C}^{G\times 1}.$$

Each entry of $\boldsymbol{c}$ is an imaginary number, and hence we split the real and imaginary part. Consequently, the input vector to the DL model is a vector with size $2G \times 1$.

OUTPUT: There is some connection between the correlation calculated above and the dominant beamspace channel entries. Apparently, the entry of $\boldsymbol{c}$ which has higher value corresponds to higher beamspace channel amplitude. However, as we saw in the OMP algorithm, greedily choosing dominant entries based on the correlation can trap the algorithm to the local optima. As a result, we plan to choose the output vector of each data sample as κ indices in the channel vector $\boldsymbol{h}$, which are dominant. Let denote the index set of the channel vector as $\boldsymbol{\alpha}$. Consequently, the output of each data is given as

$$\boldsymbol{\gamma} = \left[|\boldsymbol{h\alpha}_1|, |\boldsymbol{h}(\boldsymbol{\alpha}_2)|, \ldots, |\boldsymbol{h}(\boldsymbol{\alpha}_\kappa)|\right] \in \mathbb{R}^{\kappa\times 1}.$$

such that $|\boldsymbol{h\alpha}_1| \geq |\boldsymbol{h}(\boldsymbol{\alpha}_2)| \geq \cdots \geq |\boldsymbol{h}(\boldsymbol{\alpha}_\kappa)|$ holds.

LOSS FUNCTION: If the estimated outcome after each training round is $\tilde{\boldsymbol{\gamma}}$ and the original output vector in each data sample is given by $\boldsymbol{\gamma}$, the loss function for each data sample is given by

$$\mathcal{L}(\boldsymbol{\gamma}, \tilde{\boldsymbol{\gamma}}) = \frac{1}{\kappa}\sum_{i}^{\kappa}(\boldsymbol{\gamma}(i) - \tilde{\boldsymbol{\gamma}}(i))^2.$$

Learning Model: We adopt the NN as of the DL model, which has one hidden layer and one fully connected layer. As depicted in Fig. 2, we adopt the NN as part of the DL model, which consists of one input layer, two hidden layers, and one fully connected output layer. Each hidden layer includes a fully connected (FC) layer followed by a batch normalization layer. The number of neurons in the hidden layers are 512, and 256, respectively, enabling hierarchical feature extraction and dimensionality reduction for accurate sparse beamspace support estimation. We adopt the activation function in the FC layer as the ReLu one [54], the structure of which is given by $f_{\mathrm{Re}}(x) = \max(x, 0)$. As of the optimizer to train the NN, we consider the adaptive moment estimation (Adam) [55], which has been implemented by TensorFlow. We train the NN over 1000 epochs, and each epoch is associated with 50 mini batches. The step function is considered for the learning rate, and this rate decreases as the training advances towards the convergence. The initial value for the learning rate is set to 0.01, which gradually decreases 5-fold every 400 epochs. Taking the feed-forward and back-propagation phases into account, the training complexity of each iteration can be written as $2G{\times}512 + 512{\times}256 + \kappa 256$. Considering the entire training epochs, including batch size on the entire data set, the total complexity of training the DL model can be written as $\mathrm{O}(1000{\times}50D(2G{\times}512 + 512{\times}256 + \kappa 256))$, where D is the size of the training data set.

Once the DL model is trained with a sufficient volume of data collected from all the users, we adopt an iterative procedure similar to *Algorithm* 1 to estimate channel. Two steps of this algorithm are required to be altered in order to leverage our constructed DL model. Instead of having one single atom of the channel vector in step 5, we obtain κ atoms which are predicted from the DL model, and thus the altered step 5 is

$$\lambda_k \leftarrow \mathrm{DL}(\boldsymbol{\Phi}, \boldsymbol{r}_{k-1}).$$

Here λ_k is a vector of κ atoms instead of a single index. Consequently, step 6 is updated as follows.

$$\boldsymbol{\Lambda}_k \leftarrow \boldsymbol{\Lambda}_{k-1} \cup \lambda_k.$$

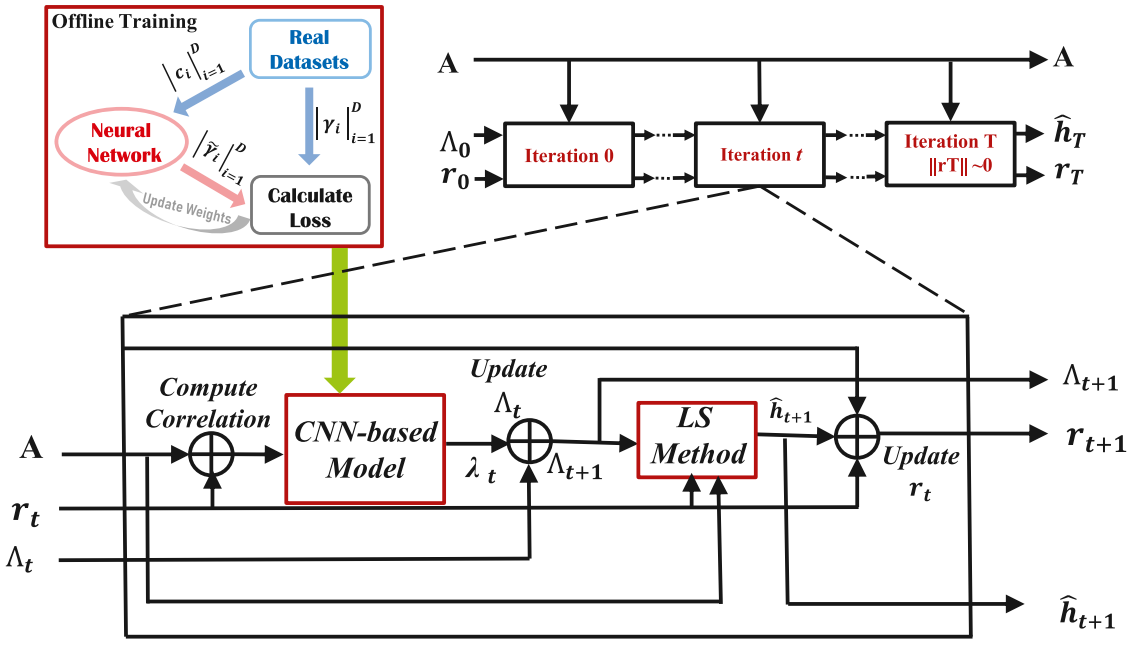


Fig. 1. A schematic diagram in realizing the mechanism of the LOMP algorithm.

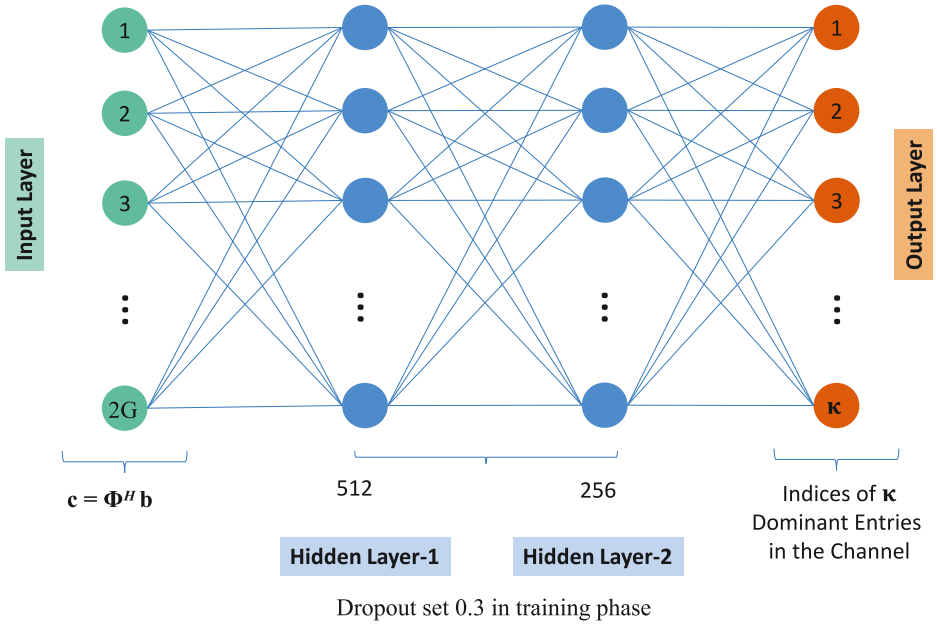


Fig. 2. The NN architecture to construct the DL model for the LOMP algorithm.

The following steps of this algorithm remain un-altered. Upon applying the LS technique, we calculate the residual at the end of each iteration and this process is continued until the residual reaches close to zero.

Computational complexity comparison between LOMP and DLCS [42] schemes: The structure and the dimension of the parameter set in a NN (i.e., the complexity of the NN) with nearly accurate prediction ability depends on the dimension of input and output vector and the number of neurons in the hidden layers. The number of hidden layers as well as the number of neurons in the hidden layers are further dependent on the dimension of input and output vectors to obtain an effective training performance from the NN. The higher the dimension of input and output vectors, the higher the complexity in the hidden layers to achieve reasonable prediction ability from the NN. If we compare the computational complexity between DLCS and LOMP, we notice that the dimension of the input vector (i.e., $2G$) is the same for both the channel estimation schemes. However, the dimension of the input vector for ours is κ, which is much smaller than N_{BS} (the dimension of the output vector of DLCS), i.e., $N_{BS} \gg \kappa$. As a result, the requirement of the number of hidden layers as well as their neurons for the LOMP algorithm is much lower compared to the DLCS scheme with the similar prediction ability. This is evident from the structure of the NN that we have adopted has much lower complexity compared to the DLCS scheme (three hidden layers and one fully connected layer with a large number of neurons compared to ours). This statement will further be justified in Section 4.

3.2. FL-based decentralized proposal

Upon the description of training the DL model in the LOMP algorithm, in this section, we explain the training process of the DL model

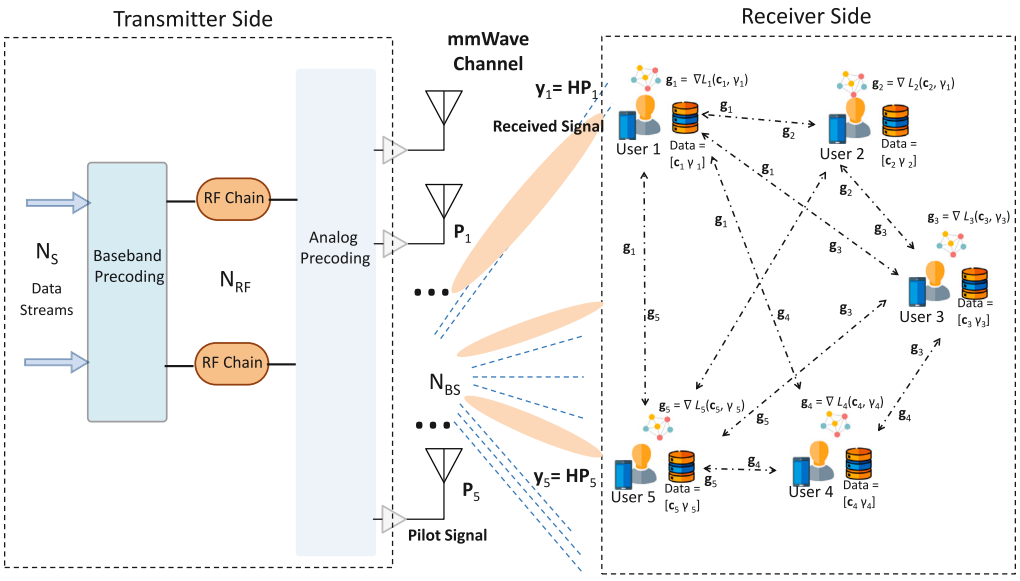


Fig. 3. A sample illustration of decentralized channel estimation in mmWave massive MIMO systems using FL.

in a decentralized setting. Instead of transmitting dummy large volume of pilot signals, the BS may intend to transmit important information in the pilot signals to each user. Since the pilot signals are user-specific and important, each user may intend to keep these pilot signals to itself and does not want to share with other users. To achieve this privacy preservation objective of the pilot signals, the decentralized FL process is an appropriate method to construct the DL model, and thus we have adopted this technique to construct the DL model for the downlink channel estimation scheme, which we name as FL-OMP. A glimpse of the architecture for the FL-OMP scheme is provided in Fig. 3.

In the LOMP algorithm, the training of the DL model is performed by collecting the local dataset $\{\mathcal{D}_u\}_{u\in\mathcal{U}}$ from the users, where $\mathcal{U}$ is the set holding the received signals from all the users. Upon collection of the whole dataset $\mathcal{D} = \sum_{u\in\mathcal{U}} \mathcal{D}_u$ from all the users, the centralized entity trains the DL model aiming to solve the following problem.

$$\min_{\theta} \ \mathcal{L}(\theta)$$
$$\text{s.t. } f\left(\boldsymbol{c}^{(i)}|\theta\right) = \boldsymbol{\gamma}^{(i)}, i = 1, \ldots, D, \tag{5}$$

where $D = |\mathcal{D}|$ is the number of training samples and the loss function $\mathcal{L}(\theta)$ in the training process is defined as

$$\mathcal{L}(\theta) = \frac{1}{D}\sum_{i=1}^{D} \|f\left(\boldsymbol{c}^{(i)}|\theta\right) - \boldsymbol{\gamma}^{(i)}\|_2^2, \tag{6}$$

which is the mean square error (MSE) between the actual output label $\boldsymbol{\gamma}^{(i)}$ and the predicted one by the DL model, $f\left(\boldsymbol{c}^{(i)}|\theta\right)$. On the other hand, in the FL-OMP algorithm, the local datasets $\mathcal{D}_{u\in\mathcal{U}}$ stored at the users are not transmitted to the centralized entity. Hence, each user u trains its own local DL model aiming to solve the following problem

$$\min_{\theta} \ \tilde{\mathcal{L}}(\theta) = \frac{1}{U}\sum_{u=1}^{U} \mathcal{L}_u(\theta)$$
$$\text{s.t. } f\left(\boldsymbol{c}_u^{(i)}|\theta\right) = \boldsymbol{\gamma}_u^{(i)}, i = 1, \ldots, D_k, \ u \in \mathcal{U}, \tag{7}$$

where $\mathcal{L}_u(\theta) = \frac{1}{D_u}\sum_{i=1}^{D_u} \|f\left(\boldsymbol{c}_u^{(i)}|\theta\right) - \boldsymbol{\gamma}_u^{(i)}\|_2^2$ holds. Note that the training for the problem in (7) is conducted at user u and the DL training in the LOMP algorithm for the problem in (5) is tackled at the centralized entity. We employ the gradient descent (GD) technique to solve these problems in an iterative manner. In the LOMP algorithm, we compute the gradient over the whole dataset as $\boldsymbol{g}(\theta_t) = \nabla\mathcal{L}(\theta_t)$, and update the parameter θ by the rule as follows.

$$\theta_{t+1} = \theta_t - \eta \boldsymbol{g}(\theta_t), \tag{8}$$

where η is the learning rate. In the FL-OMP algorithm, for constructing the DL model, each user computes the gradient individually as $\boldsymbol{g}_u(\theta_t) = \nabla\mathcal{L}_u(\theta_t)$ while solving the problem in (7), and then sends this to other users following the P2P networking concept. Over the wireless communication media, actual gradient vector $\boldsymbol{g}_u(\theta_t)$ of user u becomes corrupted, and we denote this corrupted gradient vector as $\tilde{\boldsymbol{g}}_u(\tilde{\theta}_t)$. Consequently, using the received corrupted gradients from other users, the model parameter at user u becomes corrupted. We denote this corrupted model parameter at user u in the tth iteration as $\tilde{\theta}_t$. Upon receiving the corrupted $\tilde{\boldsymbol{g}}_{u'}(\tilde{\theta}_t)$, $u' \in \mathcal{U}, u' \neq u$, user u updates the model parameter as follows.

$$\tilde{\theta}_{t+1} = \tilde{\theta}_t - \eta \boldsymbol{g}_u(\tilde{\theta}_t) - \eta\frac{1}{U-1}\sum_{u'=1,u'\neq u}^{U} \tilde{\boldsymbol{g}}_{u'}(\tilde{\theta}_t), \tag{9}$$

We define $\boldsymbol{g}_{u'}(\tilde{\theta}_t)$ and $\tilde{\boldsymbol{g}}_{u'}(\tilde{\theta}_t)$ as follows.

$$\tilde{\theta}_t = \theta_t + \Delta\theta_t, \tag{10}$$
$$\boldsymbol{g}_{u'}(\tilde{\theta}_t) = \boldsymbol{g}_{u'}(\theta_t) + \Delta\boldsymbol{g}_{u'}(\theta_t), \tag{11}$$
$$\tilde{\boldsymbol{g}}_{u'}(\tilde{\theta}_t) = \boldsymbol{g}_{u'}(\tilde{\theta}_t) + \Delta\boldsymbol{g}_{u'}(\tilde{\theta}_t), \tag{12}$$

where $\boldsymbol{g}_{u'}(\tilde{\theta}_t)$ is the corrupted gradient vector of user u' computed based on $\tilde{\theta}_t$ and $\tilde{\boldsymbol{g}}_{u'}(\tilde{\theta}_t)$ is the corrupted gradient vector of user u' received at user u. $\Delta\theta_t$, $\Delta\boldsymbol{g}_{u'}(\theta_t)$ and $\Delta\boldsymbol{g}_{u'}(\tilde{\theta}_t)$ are the bubble terms appended to θ_t, $\boldsymbol{g}_{u'}(\theta_t)$ and $\boldsymbol{g}_{u'}(\tilde{\theta}_t)$, respectively. Consequently, the model update rule of user u in (9) can be elaborated as

$$\begin{aligned}\tilde{\theta}_{t+1} &= \left[\theta_t + \Delta\theta_t\right] - \eta[\boldsymbol{g}_u(\theta_t) + \Delta\boldsymbol{g}_u(\theta_t)] - \\ &\eta\sum_{u'=1,u'\neq u}^{U} \frac{\boldsymbol{g}_{u'}(\theta_t) + \Delta\boldsymbol{g}_{u'}(\theta_t) + \boldsymbol{g}_{u'}(\tilde{\theta}_t)}{U} \\ &= \theta_t - \eta\sum_{u=1}^{U} \frac{\boldsymbol{g}_u(\theta_t)}{+}\Delta\theta_t - \eta\Delta\boldsymbol{g}_u(\theta_t) - \\ &\eta\sum_{u'=1,u'\neq u}^{U} \frac{\Delta\boldsymbol{g}_{u'}(\theta_t) + \boldsymbol{g}_{u'}(\tilde{\theta}_t)}{U-1} \\ &= \theta_{t+1} + \Delta,\end{aligned}$$

where Δ is the total noise term added to θ_{t+1} at user u. Generally, the noise terms owing to the wireless communication media follow the AWGN distribution, and thus $\Delta\theta_t$ and $\Delta\boldsymbol{g}_{u'}(\tilde{\theta}_t)$ have the AWGN distribution with variance σ_θ^2 and $\tilde{\sigma}_{u'}^2$, respectively. On the other hand, because of the linearity feature of gradients and NN layers, $\Delta\boldsymbol{g}_u(\theta_t)$ has also the AWGN distribution with variance σ_u^2. Therefore, the total

noise term Δ has the AWGN distribution with the variance $\sigma_{\Delta}^2 = \sigma_{\theta}^2 + \sigma_u^2 + \frac{\eta}{U-1}\sum_{u'=1,u'=u}^{U}[\sigma_{u'}^2 + \tilde{\sigma}_{u'}^2]$. With these all noise terms, we define a corrupted loss function $\tilde{\mathcal{L}}_u(\theta)$ at user u as

$$\tilde{\mathcal{L}}_u(\theta) = \mathcal{L}_u(\theta) + \sigma_{\Delta}^2\|g_u(\theta)\|, \tag{13}$$

We apply the first-order Taylor expansion to $\mathbb{E}\{\|\mathcal{L}_u(\theta+\Delta)\|\}$ for approximating (13), which can be written as

$$\begin{aligned}
&\mathbb{E}\{\|\mathcal{L}_u(\theta+\Delta)\|\} \approx \mathbb{E}\{\|\mathcal{L}_u(\theta) + \Delta\nabla\mathcal{L}_u(\theta)\|^2\},\\
&\approx \mathbb{E}\{\|\mathcal{L}_u(\theta)\|^2\} + \mathbb{E}\{\|\Delta\|^2\}\mathbb{E}\{\|\mathcal{L}_u(\theta)\|^2\},\\
&\approx \mathbb{E}\{\|\mathcal{L}_u(\theta)\|^2\} + \sigma_{\Delta}^2\|g(\theta)\|^2,
\end{aligned} \tag{14}$$

where the former term is the actual minimized value of the loss function without any corruption and the latter term is the noise-induced extra cost. With the corrupted loss function in (13), each user u trains the local DL model aiming to solve the following optimization problem

$$\begin{aligned}
&\min_{\theta}\ \tilde{\mathcal{L}}(\theta) = \frac{1}{U}\sum_{u=1}^{U}\tilde{\mathcal{L}}_u(\theta)\\
&\text{s.t. } f\left(c_u^{(i)}|\theta\right) = \gamma_u^{(i)}, i = 1,\ldots,D_u,\ u \in \mathcal{U},
\end{aligned} \tag{15}$$

Similar to (9), each user applies the GD technique to update noisy model parameter as follows.

$$\tilde{\theta}_{t+1} = \tilde{\theta}_t - \eta\nabla\tilde{\mathcal{L}}(\tilde{\theta}),$$

where $\nabla\tilde{\mathcal{L}}(\tilde{\theta}) = g_u(\tilde{\theta}_t) + \frac{1}{U-1}\sum_{u'=1,u'\neq u}^{U}\tilde{g}_{u'}(\tilde{\theta}_t)$ and $\tilde{g}_{u'}(\tilde{\theta}_t) = \nabla\tilde{\mathcal{L}}_u(\tilde{\theta}_t) = \nabla[\mathcal{L}_u(\tilde{\theta}_t) + \sigma_{\Delta}^2\|g_u(\tilde{\theta}_t)\|^2]$. However, corrupted gradient transmission over the wireless communication media, $\tilde{\mathcal{L}}(\theta)$ converges slowly compared to $\mathcal{L}(\theta)$. Under a structured P2P-based topology, in Algorithm 2, we summarize the steps to compute the DL model in the FL-OMP scheme. Moreover, in the following theorem, we provide a statement about the convergence of $\tilde{\mathcal{L}}(\theta)$.

Algorithm 2 The schematic diagram to construct the DL model in the FL-OMP channel estimation scheme.

1: Set the synchronization interval T_{thresh} to some user-specified value.
2: Set the separate communication channel for each user $u \in \mathcal{U}$.
3: $t \leftarrow 1$ and initialize the model $\tilde{\theta}_t$ at the end of each user $u \in \mathcal{U}$.
4: **repeat**
5: User u trains its own model θ_t using its own dataset D_u and computes gradient $g_u(\tilde{\theta}_t)$, $\forall u \in \mathcal{U}$.
6: After the interval T_{thresh}, user u broadcasts the computed gradient $g_u(\tilde{\theta}_t)$ over its designated communication channel, $\forall u \in \mathcal{U}$.
7: User u updates its model according to the gradients received from other users $\forall u' \in \mathcal{U}, u' \neq u$ following (9).
8: $t \leftarrow t+1$.
9: **until** The convergence is achieved.

Theorem 1. *The convergence process of the training of the DL in the FL-OMP scheme at user u can be written as follows.*

$$\tilde{\mathcal{L}}(\theta_t) - \tilde{\mathcal{L}}(\theta_*) \leq \|\theta_0 - \theta_*\|^2\frac{1}{2\eta t}, \tag{16}$$

where θ_0 and θ_* are the initial and optimal model parameters, respectively, and the learning rate is $\eta \leq \frac{1}{(1+\sigma_{\Delta}^2)^2\beta}$ for some $\beta \geq 0$.

Proof. Please refer to Appendix.[2]

[2] Basically, based on the condition whether the NN loss function is convex or non-convex, the convergence proof would be different. In this paper, we have provided a complete proof of the convergence process for the decentralized P2P-based FL technique based on the assumption that the NN loss function is convex. For the case that the loss function is non-convex, we would need to compute the lower bound of the expression $\min_{t\in[T]}\|\nabla\tilde{\mathcal{L}}(\theta_t)\|_2^2$ instead of the expression in Theorem 1. Due to the space constraint, we have skipped this case in this paper.

4. Performance evaluation

Followed by the description of the methodology, in this section, we conduct extensive experiments to verify the effectiveness of the proposed LOMP and FL-OMP downlink channel estimation schemes for mmWave massive MIMO systems compared to the OMP and DLCS schemes.

4.1. Simulation settings

In our simulation, we consider a mmWave massive MIMO multi-user communication system, where the BS equipped with N_{BS} antennas, with the ULA arrangement, serves multiple users simultaneously. The BS is configured with $N_{BS} = 64$ and $N_{RF} = 4$. Unless otherwise specified, the BS serves $U = 6$ users simultaneously, each of which is equipped with a single antenna. The training data was generated using a geometric SV channel model, where the channel of each user comprises L multipath components with complex Gaussian path gains $g_{u,i} \sim \mathcal{CN}(0,1)$. The AoA are uniformly distributed in $[-\pi/2, \pi/2]$, and the array response vectors are constructed assuming a ULA pattern. Channel realizations are independent across users and samples, and the sparse beamspace representation is obtained by projecting received pilot signals onto a predefined dictionary. No explicit path loss or spatial correlation models are included, however, such extensions (e.g., 3GPP-compliant models) can be considered in future work. For the channel model of the considered system, we set the number of propagation paths L_u to 2, capturing both the Line-of-Sight (LOS) and Non-Line-of-Sight (NLOS) components. For the downlink pilot transmission, we set $\mathbb{F}_{BB} = \mathbb{I}_{N_{BS}}$. Therefore, the hybrid precoding matrix $\mathbb{F}$ is equivalent to the analog precoding matrix, which is randomly generated. The beamspace dictionary grid is discretized into $G = 128$ points in the beamspace domain. The resultant beamspace dictionary matrix $\boldsymbol{A}$ of size $N_{BS} \times G$ is constructed using the array steering vectors over a grid of G angles. The analog phase shifters at the BS have a quantization resolution of $Q = 4$ bits, reflecting practical hardware constraints. During the pilot transmission aiming to estimate channel, we allocate $M = 8$ time slots. For the DLCS scheme [42], we examine the sparsity level of $J = 7$ to represent the estimated dominant atoms. Unless otherwise specified, for our proposed LOMP method, we set the number of selected atoms for the constructed DL model to $\kappa = 6$. Moreover, for both the LOMP and DLCS schemes, the number of training samples for each user D_u, to construct the DL model, is set to 5000. The SNR in the simulation ranges from −5 dB to 20 dB, incremented in the step of 5 dB, to evaluate the performance under various channel conditions.

4.2. Simulation results

4.2.1. Centralized channel estimation results

In Fig. 4, we demonstrate the performance of the proposed LOMP algorithm compared with the DLCS and OMP algorithms in terms of normalized-mean-square-error (NMSE) under various values of κ. By definition, we can write the following

$$\text{NMSE} = \frac{\sum_{u=1}^{U}\|\hat{\boldsymbol{h}} - \boldsymbol{h}_u\|_2^2}{\sum_{u=1}^{U}\|\boldsymbol{h}_u\|_2^2}. \tag{17}$$

From the figure, it is obvious that the higher the number of selected dominant atoms by the pre-specified DL model, the better the performance of the LOMP scheme. At an SNR of 20 dB, LOMP (κ=6) outperforms DLCS with an improvement of 21.2% in NMSE. This is because the more the atoms are selected, the less the chance that the LOMP scheme falls in the local optima. On the other hand, although the

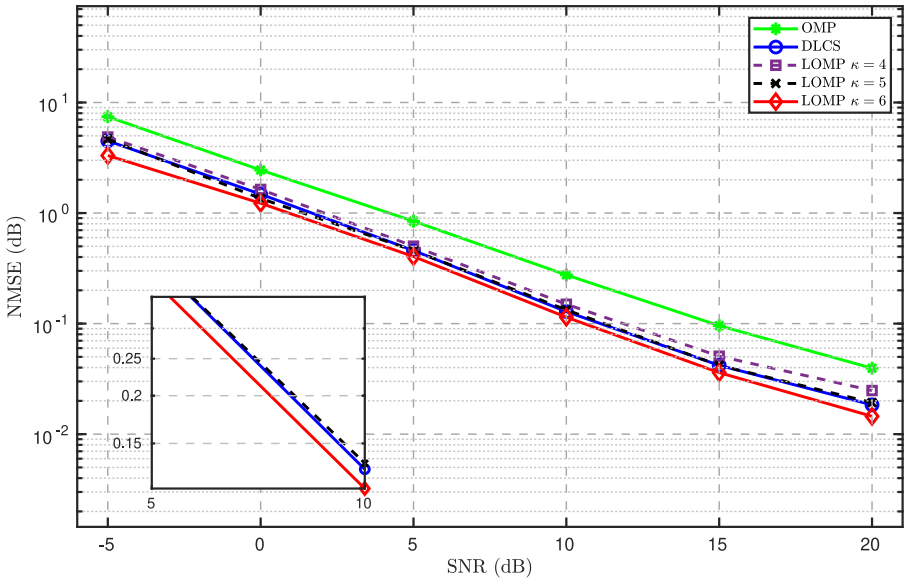


Fig. 4. NMSE comparison among various channel estimation schemes with the increasing SNR.

Table 2
Comparison between LOMP and DLCS.

LOMP	DLCS
The loop associated with the lines 4 – 10 of *Algorithm 1* runs multiple times until the absolute value of the residual reaches close to zero, i.e., $\|\|r_k\|\| \approx 0$.	The loop associated with the lines 4 – 10 of *Algorithm 1* runs only once no matter the value of the residual is.
The output dimension of the DL model is $\kappa(\kappa \ll N_{BS})$.	The output dimension of the DL model is N_{BS}.
Training complexity of the DL model: O(1000×50D(2G×512 + 512×256 + κ256)).	Training complexity of the DL model: O(1000×50D(2G×1024 + 1024×512 + 512×256 + N_{BS}256)).

DLCS scheme selects the highest number of dominant atoms compared to the proposed LOMP scheme, the performance of this scheme is lower compared to ours. This is because DLCS scheme is ignorant of the residual value after selecting the dominant atoms through the previously learned DL model. Whereas, our proposed LOMP scheme continues the selection of dominant atoms and residual calculation in an iterative procedure until the residual reaches close to zero. Therefore, as the sparsity level of the channel increases, with the DLCS scheme, only the pre-specified dominant atoms (i.e., J) in the candidate channel have non-zero values calculated through the LS method. The complexity of the LS method increases with the increasing J, and hence the DLCS scheme chooses a moderate value of J, which may cause the increasing residual with the increasing sparsity level of the channel. For this reason, even though the parameter κ of the LOMP scheme is lower than J of the DLCS scheme, ours one achieves higher performance. As presented in Table 2, the performance of our proposed LOMP scheme is achieved with much reduced computational complexity compared to the DLCS one. Finally, the OMP scheme considers the beamspace dominant entries one by one sequentially, which increases the chance that the algorithm becomes trapped at the local optimal point, and thus has worse performance.

In Fig. 5, we present the performance of all the schemes in terms of spectral efficiency. Despite there are several off-the-shelf hybrid precoding schemes for mmWave massive MIMO systems, similar to [56], we would like to compare the upper bound of the downlink spectral efficiency, which can only be achievable through the fully digital precoding scheme. We denote the estimated downlink channel matrix between the BS and all users is given by $\hat{\boldsymbol{H}} = [\hat{\boldsymbol{h}}_1, \ldots, \hat{\boldsymbol{h}}_U] \in C^{U \times N}$. Consequently, the zero-forcing (ZF) precoding matrix [57] can be represented by $\boldsymbol{F}^{dl} = (\hat{\boldsymbol{H}}^* \hat{\boldsymbol{H}}^T)^{-1} \hat{\boldsymbol{H}}^*$. In order to meet the total power budget, the normalized uth row of $\boldsymbol{F}^{\text{dl}}$, i.e., $\boldsymbol{f}_u^{\text{dl}}, u = 1, 2, \ldots, U$ should satisfy $\|\boldsymbol{f}_u^{\text{dl}}\| = 1$. Given this, the spectral efficiency of the system can be given by

$$R = \log_2 \left(\boldsymbol{I}_U + \hat{\boldsymbol{H}} (\boldsymbol{F}^{\text{dl}})^T \boldsymbol{F}^{\text{dl}} \hat{\boldsymbol{H}}^T \right). \tag{18}$$

The better the channel estimation, the more correct the constructed ZF precoding matrix, and thus the spectral efficiency. According to the discussion in the previous paragraph, the channel estimation performance of our proposed LOMP scheme is the best, and thus the constructed ZF precoding matrix by ours is more accurate. Therefore, intuitively, the resultant spectral efficiency by our proposed scheme should outperform the others, and so thus we observe in this figure.

In Fig. 6, we compare the bit-error-rate (BER) performance among all the schemes with the increasing SNR. According to the definition, BER is defined by

$$\text{BER} = \frac{1}{U} \sum_{u=1}^{U} \frac{L_u^b - \hat{L}_u^b}{L_u^b}, \tag{19}$$

where L_u^b is the originally transmitted bits by the BS to user $u, u = 1, \ldots, U$, and $\hat{L}_u^b$ is the correctly received bits by user u that was transmitted by the BS. Reception of bits to each user is quite dependent on the employed equalization as well as the coding and modulation techniques at user. Transmitted signal patterns are recovered from the received signal through equalization, which requires channel estimation. Then, the receiver can recover the originally transmitted signal through applying an appropriate demodulation technique, e.g., OFDM. The better the channel estimation, the higher the likelihood of correct bit retrieval by each user. In the previous discussion, we notice that our LOMP scheme achieves the best channel estimation performance, and

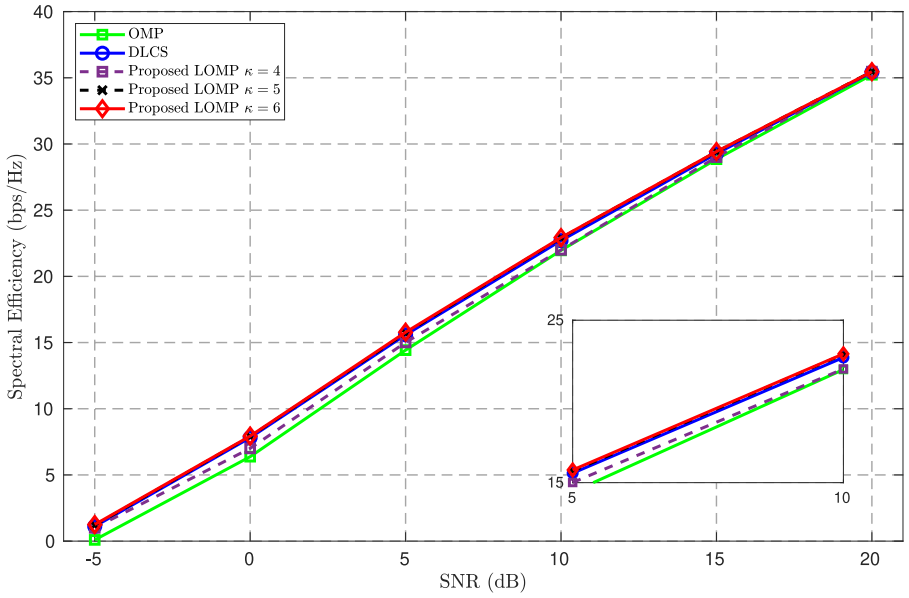


Fig. 5. Spectral efficiency comparison among various channel estimation schemes with the increasing SNR.

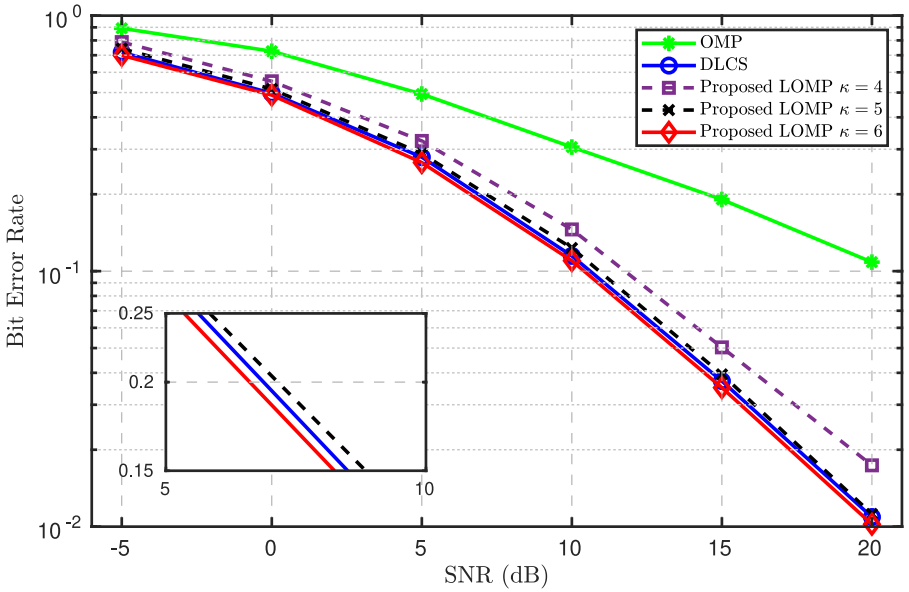


Fig. 6. BER comparison among various channel estimation schemes with the increasing SNR.

hence intuitively, the BER performance of this scheme is the highest compared to other schemes. Other reasoning of the increasing BER performance with the increasing κ is as same as that discussed in the previous paragraphs. In Fig. 7, we present the NMSE results with the increasing number of antennas. In this figure, SNR is set to 10 dB. The more the antennas, the more the propagation samples are available to develop a more accurate channel estimation scheme, and so thus we observe in this figure. The reasons of achieving better performance by the LOMP algorithm compared to other existing scheme are the same as that presented in Fig. 4.

In Fig. 8, we compare the performance of all the schemes in terms of NMSE with the increasing number of training samples. The higher the samples, the constructed DL model gets more opportunity to tune its parameters, such that this is more prone to provide more accurate results, and thus we see the decreasing NMSE for both the DLCS and LOMP schemes with the increasing number of samples. Similar to the reasons as described in Fig. 4, the performance of our scheme has the highest NMSE performance, and that for the OMP scheme is the worst.

4.2.2. Comparison between centralized and decentralized channel estimation results

In Figs. 9(a) and 9(b), we compare the performance of our proposed LOMP and FL-OMP scheme in terms of NMSE and communication overhead, respectively, with the increasing SNR. Here, the number of training samples of each user D_u for both the LOMP and FL-OMP schemes is set to 5000, and the number of users U is set to 8. The communication overhead is defined by the volume of bits sent over the communication medium. For the centralized LOMP scheme, all the users are required to send the received signals as well as the dictionary matrices to the centralized entity, the collective value of which is much smaller than the sum of transmitted data volume by the users equipped with the FL-OMP channel estimation scheme. For the FL-OMP scheme, the users only exchange the gradients of the locally constructed model following the P2P networking idea, the collective volume of which is constant no matter the values of SNR and κ. As the DL model by the FL-OMP scheme is constructed following a decentralized

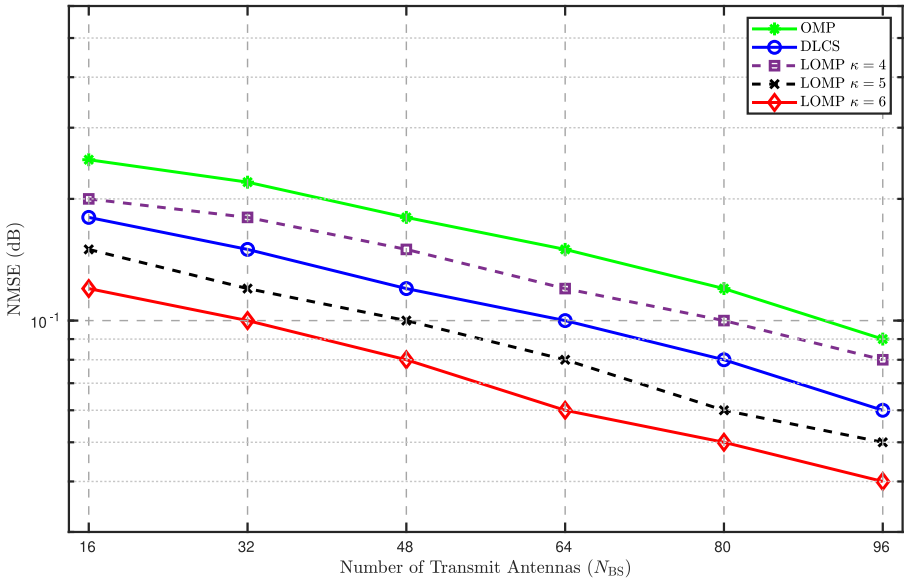


Fig. 7. NMSE comparison among various channel estimation schemes with the increasing number of antennas.

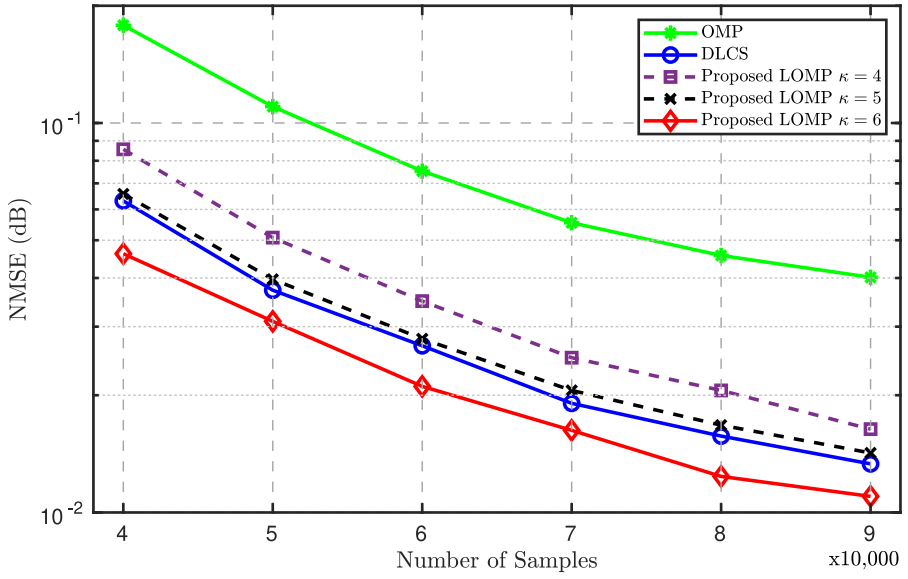
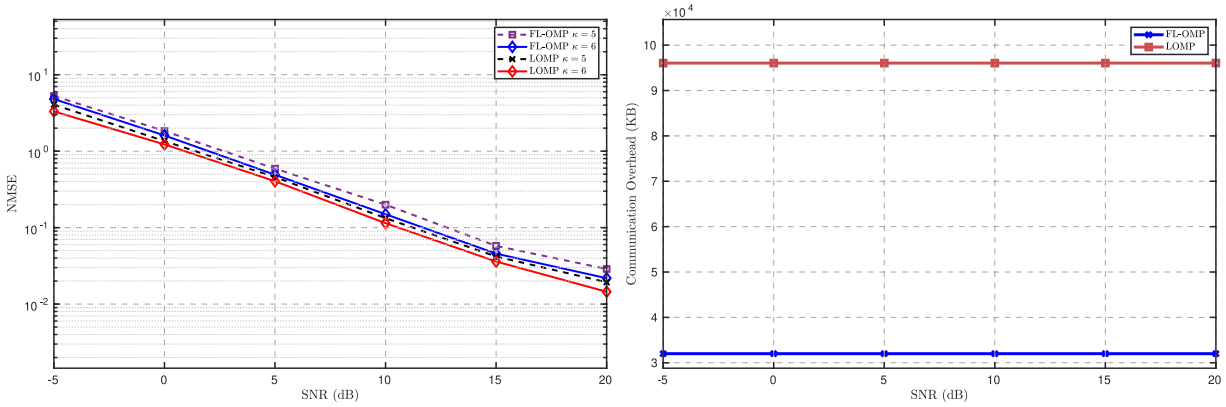
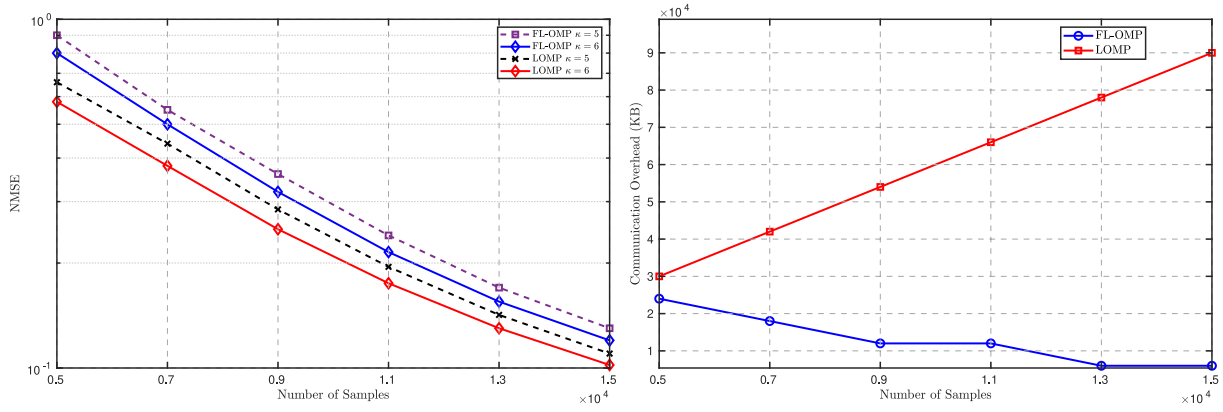


Fig. 8. NMSE comparison among various channel estimation schemes with the increasing number of training samples.



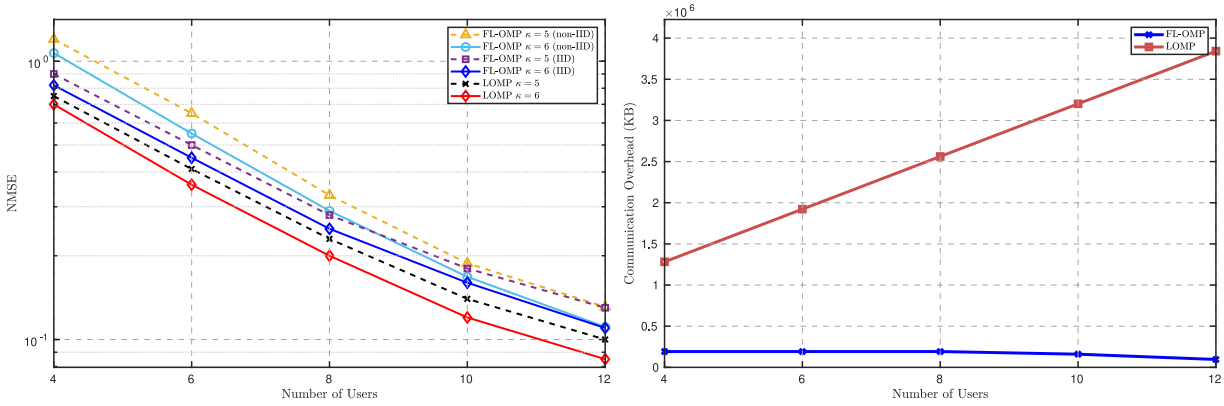
(a) NMSE (b) Communication Overhead

Fig. 9. Comparison between centralized (LOMP) and decentralized (FL-OMP) channel estimation schemes with the increasing SNR.



(a) NMSE (b) Communication Overhead

Fig. 10. Comparison between centralized (LOMP) and decentralized (FL-OMP) channel estimation schemes with the increasing number of samples.



(a) NMSE (b) Communication Overhead

Fig. 11. Comparison between centralized (LOMP) and decentralized (FL-OMP) channel estimation schemes with the increasing number of users.

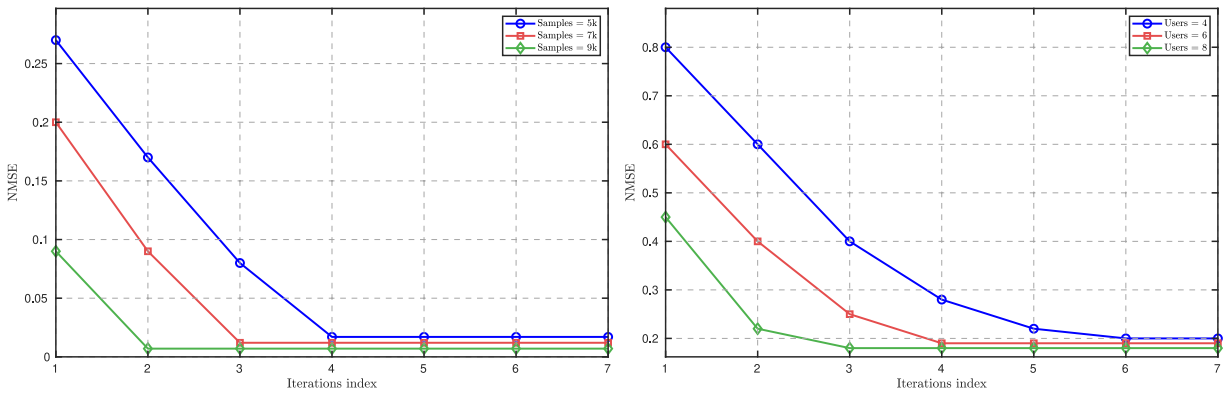
manner, the achieved performance by the FL-OMP scheme is slightly lower compared to the LOMP scheme. The reasonings of higher NMSE performance with higher value of κ is as same as that is described in the previous subsections.

In Figs. 10(a) and 10(b), we present the NMSE results as well as the communication overhead, respectively, for both the LOMP and FL-OMP schemes with the increasing number of data samples. With the increasing data samples, NMSE performance of the LOMP scheme increases with the exchange of the increasing communication overhead. However, for the FL-OMP scheme, the communication overhead for each FL round is constant with the increasing number of data samples as users are only required to exchange gradients of the DL model among themselves. However, with the increasing samples, the number of FL training rounds required to achieve convergence decreases, as depicted in Fig. 12(a). Consequently, we see a decrease in communication overhead with the increasing training samples at each user. With the increasing volume of data samples, for the similar reason as that of the LOMP scheme, the locally constructed model at each user has more opportunity to tune its parameters so that its prediction output becomes closer to reality. Consequently, the performance of FL-OMP scheme is improved with the increasing number of data samples.

In Fig. 11, we present the results (Fig. 11(a) for NMSE and Fig. 11(b) for communication overhead) of all the schemes with an increasing number of users. The communication overhead is directly connected to the convergence speed of the FL-OMP scheme. As depicted in Fig. 12(b), the higher the number of users, the lesser the number of model exchanges among the users required to achieve convergence in the FL-OMP scheme. The lesser the number of model exchanges among the users, the lesser the communication overhead for the FL-OMP scheme. Consequently, we see the non-increasing communication overhead with the increasing number of users. However, for the LOMP scheme, with the increasing number of users, a more data volume is required to be transmitted over the communication medium, and thus the communication overhead is increasing with the increasing number of users. On the other hand, with the increasing number of users, the collective data volume increases and thus the constructed DL model by the LOMP scheme mimics closer to reality. As proved in various research works related to FL, the accuracy of the constructed DL model increases with the increasing number of FL users, and this is what we observe in this figure. Furthermore, the accuracy of the constructed DL model is further dependent on the nature of the data (i.e., IID or non-IID). The more users, the more data volume is available for all the features in the constructed DL model. Therefore, the performance gap between the IID and non-IID cases decreases with the increasing number of users, as presented in Fig. 11(a).

5. Conclusion

Inspired by the DL idea, in this paper, we considered the CS-based downlink channel estimation for mmWave massive MIMO systems based on both the centralized and decentralized concepts. Using a pre-specified DL model, which is trained in the offline manner, we leveraged the idea of the traditional OMP algorithm to build a learning-based downlink channel estimation scheme, namely LOMP, which aims to demonstrate better performance in terms of both the channel estimation accuracy as well as the computational complexity compared to



(a) NMSE vs. Data Samples (b) NMSE vs. Users

Fig. 12. Convergence nature of the decentralized channel estimation scheme (FL-OMP).

the existing DL-based channel estimation scheme. As this centralized LOMP algorithm incurs huge communication overhead owing to the transmission of received signals by all the users to a centralized entity, we leveraged decentralized FL to design a downlink channel estimation scheme, namely FL-OMP, in which the users exchange the gradients of the DL model among themselves following the P2P networking concept. Extensive experiments had been conducted to justify the effectiveness of the proposed LOMP and FL-OMP schemes in terms of various performance metrics (i.e., NMSE, spectral efficiency and BER) as well as the communication overhead. The results exhibited that the LOMP scheme outperforms the existing DL-based channel estimation (i.e., DLCS) in terms of both accuracy and computational complexity. Moreover, the FL-OMP scheme achieves close performance to the LOMP one with much less communication overhead.

CRediT authorship contribution statement

Usman Aslam: Writing – original draft. **Rukhsana Ruby:** Supervision. **Dian Zhang:** Writing – review & editing. **Kaishun Wu:** Writing – review & editing. **Lu Wang:** Funding acquisition.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Appendix

For the sake of this proof, we have some assumptions, such as $\mathcal{L}(\theta)$ is convex and $\nabla\bar{\mathcal{L}}(\theta)$ is a β-smooth function. With the assistance of these assumptions, we further have Lemma 1.

Lemma 1. *$\bar{\mathcal{L}}(\theta)$ is $\bar{\beta}$-smooth, i.e., $\|\nabla\bar{\mathcal{L}}(\theta)-\nabla\bar{\mathcal{L}}(\theta')\|\leq\bar{\beta}\|\theta-\theta'\|$, where $\bar{\beta}=(1+2\sigma_{\Delta}^2)^2\beta$.*

Proof. From (13), we have

$$\begin{aligned}
&\|\nabla\bar{\mathcal{L}}(\theta)-\nabla\bar{\mathcal{L}}(\theta')\|\\
&=\|\nabla\left(\mathcal{L}(\theta)+\sigma_{\Delta}^2\parallel\nabla\mathcal{L}(\theta)\|^2\right)\\
&\quad-\nabla\left(\mathcal{L}(\theta')+\sigma_{\Delta}^2\|\nabla\mathcal{L}(\theta')\|^2\right)\|\\
&=\|\nabla\bar{\mathcal{L}}(\theta)-\nabla\bar{\mathcal{L}}(\theta')+\sigma_{\Delta}^2\Big(\nabla\|\nabla\mathcal{L}(\theta)\|^2\\
&\quad-\nabla\|\nabla\mathcal{L}(\theta')\|^2\Big)\|^2\|\\
&=\|\nabla\bar{\mathcal{L}}(\theta)-\nabla\bar{\mathcal{L}}(\theta')+2\sigma_{\Delta}^2(\nabla\bar{\mathcal{L}}(\theta)-\nabla\bar{\mathcal{L}}(\theta'))\|\\
&=(1+2\sigma_{\Delta}^2)^2\|\nabla\bar{\mathcal{L}}(\theta)-\nabla\bar{\mathcal{L}}(\theta')\|\\
&\leq\bar{\beta}\|\theta-\theta'\|^2.
\end{aligned}$$

The derivation in the last line follows from the β-smooth nature of $\mathcal{L}(\theta)$ function. The conclusion of *Lemma 1* further justifies $\nabla^2\bar{\mathcal{L}}(\theta)\leq\bar{\beta}\boldsymbol{I}_P$, where P is the dimension of the model parameter vector, i.e., $P=|\theta|$. Given that $\theta_t=\theta_{t-1}-\eta\nabla\bar{\mathcal{L}}(\theta)$, taking the second degree Taylor expansion of $\bar{\mathcal{L}}(\theta)$ yields

$$\begin{aligned}
&\bar{\mathcal{L}}(\theta_t)\leq\bar{\mathcal{L}}(\theta_{t-1})+\nabla\bar{\mathcal{L}}(\theta_{t-1})^T(\theta_t-\theta_{t-1})\\
&+\frac{1}{2}\nabla^2\bar{\mathcal{L}}(\theta_{t-1})\|\theta_t-\theta_{t-1}\|^2\\
&\leq\bar{\mathcal{L}}(\theta_{t-1})+\nabla\bar{\mathcal{L}}(\theta_{t-1})^T(\theta_t-\theta_{t-1})+\frac{1}{2}\bar{\beta}\|\theta_t-\theta_{t-1}\|^2\\
&=\bar{\mathcal{L}}(\theta_{t-1})+\nabla\bar{\mathcal{L}}(\theta_{t-1})^T\nabla\bar{\mathcal{L}}(\theta_{t-1})+\frac{1}{2}\bar{\beta}\|\eta\nabla\bar{\mathcal{L}}(\theta_{t-1})\|^2\\
&=\bar{\mathcal{L}}(\theta_{t-1})-\left(1-\frac{\bar{\beta}\eta}{2}\right)\eta\|\nabla\bar{\mathcal{L}}(\theta_{t-1})\|^2\\
&\leq\bar{\mathcal{L}}(\theta_{t-1})-\frac{\eta}{2}\|\nabla\bar{\mathcal{L}}(\theta_{t-1})\|^2.
\end{aligned}$$

The last derivation follow from the fact that $\eta\leq\frac{1}{\bar{\beta}}$ holds, and thus we have $\frac{1}{2}\bar{\beta}\eta-1\leq-\frac{1}{2}$. Now, if θ_* is the parameter at the convergence point, we have

$$\begin{aligned}
&\bar{\mathcal{L}}(\theta_t)\leq\bar{\mathcal{L}}(\theta_*)+\nabla\bar{\mathcal{L}}(\theta_t)(\theta_t-\theta_*)-\frac{\eta}{2}\|\nabla\bar{\mathcal{L}}(\theta_t)\|^2\\
&\leq\bar{\mathcal{L}}(\theta_t)-\bar{\mathcal{L}}(\theta_*)\leq\frac{1}{2\eta}\left[2\eta\nabla\bar{\mathcal{L}}(\theta_t)(\theta_t-\theta_*)-\eta^2\|\nabla\bar{\mathcal{L}}(\theta_t)\|^2\right]\\
&\leq\frac{1}{2\eta}\left[\|\theta_t-\theta_*\|^2-\|\theta_t-\theta_*-\eta\nabla\bar{\mathcal{L}}(\theta_t)\|^2\right]\\
&=\frac{1}{2\eta}\left(\|\theta_t-\theta_*\|^2-\|\theta_{t+1}-\theta_*\|^2\right).
\end{aligned}$$

Now, summing over all the loss functions $i=1,\cdots,t$, yields

$$\begin{aligned}
&\sum_{i=1}^{t}\left[\bar{\mathcal{L}}(\theta_i)-\bar{\mathcal{L}}(\theta_*)\right]\leq\frac{1}{2\eta}\sum_{i=1}^{t}\left(\|\theta_{i-1}-\theta_*\|^2-\|\theta_i-\theta_*\|^2\right)\\
&\frac{1}{2\eta}\left(\|\theta_0-\theta_*\|^2-\|\theta_t-\theta_*\|^2\right)\\
&\leq\frac{1}{2\eta}\|\theta_0-\theta_*\|^2. \qquad (20)
\end{aligned}$$

On the other hand, $\bar{\mathcal{L}}(\theta_t)$ is a decreasing function, and thus we have

$$\bar{\mathcal{L}}(\theta_t)-\bar{\mathcal{L}}(\theta_*)\leq\frac{1}{t}\sum_{i=1}^{t}\left[\bar{\mathcal{L}}(\theta_i)-\bar{\mathcal{L}}(\theta_*)\right]. \qquad (21)$$

Substituting the outcome of (21) to (20), we can conclude the statement of Theorem 1.

Data availability

The data that has been used is confidential.

References

[1] W. Hong, Z.H. Jiang, C. Yu, D. Hou, H. Wang, C. Guo, Y. Hu, The role of millimeter-wave technologies in 5G/6G wireless communications, IEEE J. Microw. 1 (1) (2021) 101–122.

[2] M.T. Mezaal, N.B.M. Aripin, N.S. Othman, A.H. Sallomi, The effect of urban environment on large-scale path loss model's main parameters for mmwave 5G mobile network in Iraq, Open Eng. 14 (1) (2024) 20220601.

[3] H. Zarini, M.R. Mili, M. Rasti, S. Andreev, P.H. Nardelli, M. Bennis, Intelligent analog beam selection and beamspace channel tracking in THz massive MIMO with lens antenna array, IEEE Trans. Cogn. Commun. Netw. 9 (3) (2023) 629–646.

[4] Z. Zhu, R. Yang, J. Zhang, S. Xu, C. Li, Y. Huang, L. Yang, Sparse Bayesian learning-based adaptive codebook for near-field channel estimation, in: Proc. IEEE ICC, IEEE, 2024, pp. 2366–2371.

[5] A. Azari, A. Skrivervik, H. Aliakbarian, Design of a novel wide-angle rotman lens beamformer for 5G mmwave applications, Sci. Rep. 14 (1) (2024) 1245.

[6] A. Khlifi, R. Bouallegue, Performance analysis of LS and LMMSE channel estimation techniques for LTE downlink systems, Int. J. Wirel. Mob. Networks (IJWMN) 3 (5) (2011) 53–62, http://dx.doi.org/10.5121/ijwmn.2011.3511.

[7] S. Arif, M.A. Khan, S.U. Rehman, Wireless channel estimation for low-power IoT devices using real-time data, IEEE Access (2024).

[8] P. Wang, J. Li, X. Liu, P. Wang, Multi-stage training optimization for pilot compression and channel estimation in massive MIMO systems under quasi-sparse channel environment, IEEE Commun. Lett. 26 (12) (2022) 3059–3063.

[9] T.L. Marzetta, Noncooperative cellular wireless with unlimited numbers of base station antennas, IEEE Trans. Wirel. Commun. 9 (11) (2010) 3590–3600, http://dx.doi.org/10.1109/TWC.2010.2053761.

[10] L. Tong, B.M. Sadler, M. Dong, Pilot-assisted wireless transmissions: General model, design criteria, and signal processing, IEEE Signal Process. Mag. 21 (6) (2004) 12–25, http://dx.doi.org/10.1109/MSP.2004.823992.

[11] W. Ma, L. Zhu, R. Zhang, Compressed sensing based channel estimation for movable antenna communications, IEEE Commun. Lett. (2023).

[12] Y. Zhang, Secure wireless communications based on compressive sensing: A survey, IEEE Commun. Surv. Tutor. 21 (2) (2018) 1093–1111.

[13] O.O. Oyerinde, et al., Compressive sensing-based channel estimation for uplink and downlink reconfigurable intelligent surface-aided millimeter wave massive mimo systems, Electronics 13 (15) (2024) 2909.

[14] J.A. Tropp, A.C. Gilbert, Signal recovery from random measurements via orthogonal matching pursuit, IEEE Trans. Inform. Theory 53 (12) (2007) 4655–4666, http://dx.doi.org/10.1109/TIT.2007.912354.

[15] D.L. Donoho, A. Maleki, A. Montanari, Message-passing algorithms for compressed sensing, Proc. Natl. Acad. Sci. 106 (45) (2009) 18914–18919, http://dx.doi.org/10.1073/pnas.0902700106.

[16] C. Ruan, Z. Zhang, H. Jiang, H. Zhang, J. Dang, L. Wu, Simplified learned approximate message passing network for beamspace channel estimation in mmwave massive MIMO systems, IEEE Trans. Wirel. Commun. 23 (5) (2024) 5142–5156.

[17] Z. Hu, Y. Chen, C. Han, PRINCE: A pruned AMP integrated deep CNN method for efficient channel estimation of millimeter-wave and terahertz ultra-massive MIMO systems, IEEE Trans. Wirel. Commun. 22 (11) (2023) 8066–8079.

[18] M. Jian, Y. Zhao, A modified off-grid SBL channel estimation and transmission strategy for RIS-assisted wireless communication systems, in: 2020 International Wireless Communications and Mobile Computing, IWCMC, IEEE, 2020, pp. 1848–1853, http://dx.doi.org/10.1109/IWCMC48107.2020.9148350.

[19] Y. Zhang, M. El-Hajjar, L.L. Yang, Multi-layer sparse Bayesian learning for mmwave channel estimation, IEEE Trans. Veh. Technol. 73 (3) (2023) 3485–3498, http://dx.doi.org/10.1109/TVT.2023.3330415.

[20] B. Xiao, P. Huang, Z. Lin, J. Zeng, T. Lv, Channel estimation using variational Bayesian learning for multi-user mmwave MIMO systems, IET Commun. 15 (4) (2021) 566–579, http://dx.doi.org/10.1049/iet-com.2020.0275.

[21] B. Wang, X. Qin, W. Li, Z. Li, L. Zhu, Learned iterative shrinkage and thresholding algorithm for terahertz sparse deconvolution, Opt. Express 30 (11) (2022) 18238–18249, http://dx.doi.org/10.1364/OE.456597.

[22] X. Zhang, X. Dong, M. Shen, A cell-free millimeter-wave channel estimation algorithm based on federated learning, in: Proc. International Conference on Electronic Engineering and Informatics, EEI, 2023, pp. 480–486.

[23] T. Tian, A. Raj, B.M. Xavier, Y. Zhang, F. Wu, K. Yang, A robust ADMM-based optimization algorithm for underwater acoustic channel estimation, in: OCEANS 2023 - Limerick, IEEE, 2023, pp. 1–5, http://dx.doi.org/10.23919/OCEANS2023.10196776.

[24] J. Sun, M. Jia, Q. Guo, X. Gu, Y. Gao, Power distribution based beamspace channel estimation for mmwave massive MIMO system with lens antenna array, IEEE Trans. Wirel. Commun. 21 (12) (2022) 10695–10708.

[25] Z. Li, Q. Xue, C. Dong, K. Niu, H. Wang, Q. Huang, Q. Gao, Y. Fei, J. Zuo, Deep learning beamspace channel estimation for mmwave massive MIMO with switch-based selection network, in: Proc. IEEE WCNC, 2024, pp. 1–6.

[26] J.-M. Kang, C.-J. Chun, I.-M. Kim, Deep-learning-based channel estimation for wireless energy transfer, IEEE Commun. Lett. 22 (11) (2018) 2310–2313.

[27] C.-J. Chun, J.-M. Kang, I.-M. Kim, Deep learning-based channel estimation for massive MIMO systems, IEEE Wirel. Commun. Lett. 8 (4) (2019) 1228–1231.

[28] X.F. Kang, Z.H. Liu, M. Yao, Deep learning for joint pilot design and channel estimation in MIMO-OFDM systems, Sensors 22 (11) (2022) 4188, http://dx.doi.org/10.3390/s22114188.

[29] M. Soltani, V. Pourahmadi, A. Mirzaei, H. Sheikhzadeh, Deep learning-based channel estimation, IEEE Commun. Lett. 23 (4) (2019) 652–655, http://dx.doi.org/10.1109/LCOMM.2019.2895754.

[30] M. Wang, A. Wang, Z. Liu, J. Chai, Deep learning based channel estimation method for mine OFDM system, Sci. Rep. 13 (1) (2023) 17105, http://dx.doi.org/10.1038/s41598-023-43930-z.

[31] H. Huang, Z. Yang, Z. Ding, P. Fan, H. Poor, Deep learning for super-resolution channel estimation and DOA estimation based massive MIMO system, IEEE Trans. Veh. Technol. 67 (9) (2018) 8549–8560.

[32] L. Wan, K. Liu, W. Zhang, Deep learning-aided off-grid channel estimation for millimeter wave cellular systems, IEEE Trans. Wirel. Commun. 21 (5) (2022) 3333–3348.

[33] S. Zheng, S. Wu, H. Jia, C. Jiang, L. Kuang, Hybrid driven learning for joint activity detection and channel estimation in IRS-assisted massive connectivity, IEEE Trans. Wirel. Commun. 23 (9) (2024) 10834–10849.

[34] S.A.U. Haq, A.K. Gizzini, S. Shrey, S.J. Darak, S. Saurabh, M. Chafii, Deep neural network augmented wireless channel estimation for preamble-based OFDM PHY on Zynq system on chip, IEEE Trans. Very Large Scale Integr. (VLSI) Syst. 31 (7) (2023) 1026–1038.

[35] Z. Zhou, Z. Wei, J. Ren, Y. Yin, G. Frölund Pedersen, M. Shen, Two-order deep learning for generalized synthesis of radiation patterns for antenna arrays, IEEE Trans. Artif. Intell. 4 (5) (2023) 1359–1368.

[36] E.C. Vilas Boas, et al., Artificial intelligence for channel estimation in multicarrier systems for B5G/6G communications: a survey, EURASIP J. Wirel. Commun. Netw. 2022 (1) (2022) 116.

[37] J. Chen, F. Meng, N. Ma, X. Xu, P. Zhang, Model-driven deep learning-based sparse channel representation and recovery for wideband mmwave massive MIMO systems, IEEE Trans. Veh. Technol. 72 (12) (2023) 16788–16793.

[38] Y.-S. Liu, S.D. You, Y.-C. Lai, Machine learning-based channel estimation techniques for ATSC 3.0, Information 15 (6) (2024) 350.

[39] K.-H. Liu, W. Zhang, L. Wan, Fast convex method for off-grid millimeter-wave/sub-terahertz channel estimation via exploiting joint sparse structure, in: Proc. IEEE ICC, 2022, pp. 919–925.

[40] W. Sloane, C. Gentile, M. Shafi, J. Senic, P.A. Martin, G.K. Woodward, Measurement-based analysis of millimeter-wave channel sparsity, IEEE Antennas Wirel. Propag. Lett. 22 (4) (2023) 784–788, http://dx.doi.org/10.1109/LAWP.2022.3225246.

[41] X. Wei, C. Hu, L. Dai, Deep learning for beamspace channel estimation in millimeter-wave massive MIMO systems, IEEE Trans. Commun. 69 (1) (2020) 182–193.

[42] W. Ma, C. Qi, Z. Zhang, J. Cheng, Sparse channel estimation and hybrid precoding using deep learning for millimeter wave massive MIMO, IEEE Trans. Commun. 68 (5) (2021) 2838–2849.

[43] M.M. Amiri, D. Gündüz, Federated learning over wireless fading channels, IEEE Trans. Wirel. Commun. 19 (5) (2020) 3546–3557.

[44] M.M. Wadu, S. Samarakoon, M. Bennis, Joint client scheduling and resource allocation under channel uncertainty in federated learning, IEEE Trans. Commun. 69 (9) (2021) 5962–5974.

[45] G. Gad, A. Farrag, A. Aboulfotouh, K. Bedda, Z.M. Fadlullah, M.M. Fouda, Joint self-organizing maps and knowledge-distillation-based communication-efficient federated learning for resource-constrained UAV-IoT systems, IEEE Internet Things J. 11 (9) (2024) 15504–15522.

[46] N. Bansal, S. Mittal, R.S. Bali, N. Kumar, J. Rodrigues, L. Zhao, Federated learning based task orchestration scheme using intelligent vehicular edge networks, in: Proc. IEEE ICC, 2023, pp. 416–421.

[47] V. Ardianto Nugroho, B.M. Lee, A survey of federated learning for mmwave massive MIMO, IEEE Internet Things J. 11 (16) (2024) 27167–27183.

[48] A.M. Elbir, S. Coleri, Federated learning for channel estimation in conventional and RIS-assisted massive MIMO, IEEE Trans. Wirel. Commun. 21 (6) (2021) 4255–4268.

[49] L. Zhao, H. Xu, Z. Wang, X. Chen, A. Zhou, Joint channel estimation and feedback for mm-wave system using federated learning, IEEE Commun. Lett. 26 (8) (2022) 1819–1823.

[50] A.M. Elbir, W. Shi, K.V. Mishra, S. Chatzinotas, Federated multi-task learning for THZ wideband channel and DOA estimation, in: Proc. IEEE ICASSPW, 2023, pp. 1–5.

[51] B. Qiu, X. Chang, X. Li, H. Xiao, Z. Zhang, Federated learning-based channel estimation for RIS-aided communication systems, IEEE Wirel. Commun. Lett. 13 (8) (2024) 2130–2134.

[52] Q. Yi, P. Yang, Z. Liu, Y. Huang, S. Zammit, Model-driven federated learning for channel estimation in millimeter-wave massive MIMO systems, in: Proc. IEEE WCNC, 2024, pp. 1–6.

[53] A. Abdallah, A. Celik, M.M. Mansour, A.M. Eltawil, RIS-aided mmwave MIMO channel estimation using deep learning and compressive sensing, IEEE Trans. Wirel. Commun. 22 (5) (2023) 3503–3521.

[54] G. Lin, W. Shen, Research on convolutional neural network based on improved relu piecewise activation function, Procedia Comput. Sci. 131 (2018) 977–984.
[55] D.P. Kingma, Adam: A method for stochastic optimization, 2014, arXiv preprint arXiv:1412.6980.
[56] C. Zhang, Y. Jing, Y. Huang, L. Yang, Performance analysis for massive MIMO downlink with low complexity approximate zero-forcing precoding, IEEE Trans. Commun. 66 (9) (2018) 3848–3864.
[57] S. Saad, A. Almamori, Energy efficiency optimization of cell-free massive MIMO with zero forcing precoding, Wirel. Pers. Commun. (2024) 1–22.