Received 25 September 2024; accepted 9 October 2024. Date of publication 14 October 2024; date of current version 30 October 2024.

Digital Object Identifier 10.1109/OJCOMS.2024.3479234

Model-Driven Channel Estimation for MIMO Monostatic Backscatter System With Deep Unfolding

YULIN ZHOU [1] (Member, IEEE), XIAOTING LI [2], XIANMIN ZHANG [1] (Member, IEEE), XIAONAN HUI [3] (Member, IEEE), AND YUNFEI CHEN [4] (Senior Member, IEEE)

[1]Ningbo Innovation Center, Zhejiang University, Ningbo 315000, China
[2]Wireless and Computing Power Product Operation Department, Zhongxing Telecommunication Equipment Corporation, Shanghai 200000, China
[3]College of Information Science and Electronic Engineering, Zhejiang University, Hangzhou 310058, China
[4]Department of Engineering, University of Durham, DH1 3LE Durham, U.K.

CORRESPONDING AUTHOR: X. HUI (e-mail: x.hui@zju.edu.cn)

This work was supported in part by the Key Research and Development Program of Zhejiang Province under Grant 2023C01043; in part by the Science and Technology Innovation 2025 Major Project of Ningbo under Grant 2023Z236; and in part by the National Natural Science Foundation of China under Grant 62371419.

ABSTRACT Monostatic backscatter has garnered significant interest due to its distinct benefits in low-cost passive sensing. Observing and sensing with backscatter necessitates determining the phase and amplitude of the backscatter channel to identify the state of the target of interest. In the detection of multiple targets, colliding signals can distort the backscatter channel, complicating channel state recovery. It becomes even more challenging when multiple backscattering devices (BDs) are used. This paper proposes a novel channel estimation scheme to tackle the challenge, which is applied to a monostatic backscatter communication system with multiple reader antennas (RAs) and backscatter devices. Specifically, we propose a backscatter communication model and subsequently develop a de-interfering channel estimation framework that considers the ambient interference in the channel, named model-driven unfolded channel estimation (MUCE). To validate the effectiveness and advantages of the MUCE method, it is compared with the least square (LS) algorithm and convolutional neural network (CNN). The results prove that MUCE requires lower computational costs for the same channel estimation performance and achieves an optimal balance between estimation performance and computational expense.

INDEX TERMS Channel estimation, deep unfolding, interference cancellation, monostatic backscatter.

I. INTRODUCTION

THIS Backscatter communication (BC) has emerged as an appealing technology for various wireless systems and applications. Compared with conventional wireless architecture sensor systems with radio frequency (RF) transceiver components like local oscillators, mixers, and converters, BDs can operate with low power consumption [1]. Numerous BDs have been investigated and deployed in Internet of Things (IoT) applications, such as warehouse management and supply chain monitoring [2], [3]. Their advantages and unique features in the wireless sensing field have been proposed and demonstrated worldwide, with applications encompassing localization, tracking, and motion recognition [4], [5], [6], [7], [8]. The backscatter system is made of the carrier emitter and signal receiver with either full-duplex or bistatic architecture depending on whether the reader transmitting and RAs are shared or not.

The majority of existing backscatter research relies on channel state information (CSI). However, to obtain accurate CSI for backscatter communications symbols is challenging due to limited energy resources available to support the transmission of pilot and the channel variance. A source channel estimation algorithm based on expectation maximization has been proposed, which can determine channel parameters without the need for training symbols [9]. The authors in [10] investigates the impact of phase noise on downlink compressive channel estimation in massive multiple-input multiple-output (MIMO) systems. Furthermore, the authors propose a novel tensor-based approach [11] that addresses

both the channel estimation and target sensing problems. For dedicating signal readers, the authors in [12] have developed a scheme that incorporates the backscatter signal receiver into an orthogonal frequency-division multiplexing access point, leading to detect the backscatter symbols iteratively and perform interference cancellation simultaneously.

In monostatic backscatter communication, the carrier emitter and backscattered signal receiver are co-located and referred to as readers. The downlink carrier activation signal and uplink backscatter signal are reciprocal when sharing the same antenna [13]. Building on this property, the authors in [14] have derived a new LS estimator for forward and backward links between a full-duplex multiple-input multiple-output (MIMO) reader and a single tag, resulting in a linear least mean square (LLMS) error estimation for the corresponding backscatter channel. In addition to backscatter channel estimation under flat fading, the authors in [15] have examined backscatter communication systems under frequency-selective channels and proposed a novel deep learning-based algorithm for tags with varying state parameters. The authors in [16] have introduced a system model for information decoding, which combines RF sources and ambient backscatter equipment cooperatively. Further research in [17] has explored the cognitive ambient backscatter of this type of backscatter system, sharing the same spectrum and RF source as conventional systems, and employed a multi-antenna reader to address co-channel direct link interference in the legacy system. Besides, in [18], the authors have proposed a new receiver structure with a simpler design based on direct averaging of signal samples for detection, in conjunction with time-selective fading channels and non-coherent detectors. Existing research on channel estimation in backscatter communication employs both traditional algorithms and deep learning. For monostatic backscatter, considering environmental interference, two main challenges arise: first, as the number of tags and antennas increases, the problem's dimensionality grows significantly, leading to higher computational costs and reduced channel estimation efficiency. Second, changes in the surrounding environment affect ambient interference estimation, requiring timely adjustments to the interference model to maintain estimation quality. Traditional methods often struggle to effectively address these challenges.

In [19], the authors have examined the channel estimation problem from a signal processing perspective for readers in large-scale antennas array scenario. They perform preliminary channel estimation for different states using the LS algorithm and then combine the angular rotation with a discrete Fourier transform of the estimates to obtain the channel gain and direction of arrival. Most of the research focuses on tags with a single antenna. Considering multi-antenna tags capable of simultaneous energy harvesting and data transmission, the authors in [20] have proposed a undefined detector based on the generalized likelihood ratio test (GLRT) for the multi-antenna channel between tag and reader, circumventing the need to consider CSI, signal power, and noise variance. In general backscatter networks, energy detection occurs at the receiver. The authors in [21] have introduced a matched filter-based OFDM carrier ambient backscatter modulation scheme that forces the OFDM signal forwarded by the BD to be consistently received at the receiver. The studies mentioned above primarily focus on backscatter channel estimation using conventional numerical methods. Nevertheless, more effective deep learning-based methods from conventional wireless communication as supervised learning, reinforcement learning, neural networks, and transfer learning, have also been applied in backscatter communication [22].

The authors in [23] conduct research on orthogonal frequency-division multiplexing systems aided by re configurable intelligent surfaces. Considering the problem that the lack of cyclic prefix (CP) generated by the additional cascading channels of RIS and the nonlinear distortion caused by imperfect hardware affect the channel estimation accuracy, a channel estimation network based on enhanced extreme learning machine was proposed. Based on the traditional linear receiver model, the authors in [24] proposes a joint model and a data-driven receiver scheme. Firstly, least squares estimation and zero-forcing equilibrium are used to extract the initial features for channel estimation and data detection. Then, shallow neural networks called CE-Net and SD-Net were developed to improve channel estimation and data detection. The scheme effectively suppresses symbol mis-identification and achieves similar or better bit error rate (BER) performance without the need for second-order statistics on channels and noise.

In [25], the authors have introduced a tag-assisted transmission framework that transmits two known tag pilots to data transmission. The performance of the proposed constellation learning method is comparable to an optimal detector with perfect CSI. In [26], a machine learning-based detection method for the ambient backscatter communication (AmBC) system has been proposed, transforming the problem of using energy detectors or minimum mean square error (MMSE) detectors for detecting label signals with high bit error rate (BER) into a classification problem. For the same problem, [27] has presented a deep transfer learning detection framework, consisting of offline learning, transfer learning, and online detection. The framework is designed with a signal detector based on the pre-trained deep neural network (DNN) and a few pilots to obtain a deep transfer learning-based likelihood ratio test. Regarding interference handling, the authors in [28] have modelled the interaction between users and intelligent interference sources in an AmBC network as a game. Afterwards, they employ a Q-learning algorithm with a dynamic iterative process to solve the convex optimization problem and obtain the optimal strategy.

In monostatic multi-antenna backscatter networks, [29] introduces a fast and flexible convolutional neural network

(FFDNet) based on the optimal solution of a linear minimum mean square error (LMMSE) estimator for backscatter channels. This approach estimates forward channel coefficients directly from the backscattered signal by using a DNN with a custom loss function. By framing the MMSE channel estimation as an image denoising problem, [30] has proposed two CNN-based methods for denoising and approximating the optimal MMSE channel estimation solution. Similarly, [31] introduces a CNN-based deep residual network (CDRN) that implicitly learns residual noise to recover channel coefficients from noisy pilot-based observations. However, these learning-based channel estimation approaches treat communication systems as closed boxes, relying on extensive data for training and lacking a theoretical basis and interpretability in network design.

A. MOTIVATION

According to our research, most of the current channel estimation work revolves around traditional communication systems. In backscatter systems, accurate channel estimation plays an important role in correctly decoding the state of backscatter devices. The research work in this area is mainly based on traditional neural networks such as least squares or data-driven neural networks, which may have problems of difficulty and high computational complexity. Therefore, we hope to explore a model-driven deep learning method to design a more efficient channel estimation network by combining backscatter communication model and deep learning to solve the decoding problem in the field of backscatter communication. Low-complexity channel estimation models are crucial for practical applications, such as vehicle-to-everything (V2X) communications [32], M-MIMO based cognitive radio system [33], and other wireless communication applications such as LTE (Long-Term Evolution), DVB (Digital Video Broadcasting), DAB (Digital Audio Broadcasting). However, channel estimation models still face to factors such as noise, interference, dynamic environments, and limited computational resources, especially in low-power devices. Simplified models often struggle to accurately capture real-world channel conditions, and scalability becomes a significant issue in large systems. Additional challenges include managing sparse channels, ensure energy efficiency, balance real-time processing needs, minimize pilot overhead, and handle channel state feedback. Furthermore, to adapt diverse environments with limited training data, we need to reduce the complexity of channel estimation, make it effectively with minimal training resources.

B. CONTRIBUTION

In this paper, we present a novel MUCE scheme for multi-antenna, multi-BD monostatic backscatter (MBS) communication networks with ambient interference. A deep unfolding neural network is employed for channel parameter estimation, which circumvents the computationally intensive and complex iterations in traditional LS channel estimation methods. MUCE is the first scheme to apply model-driven deep unfolding to MBS. The main contributions of this learning-based scheme are summarized below:

(1) Network architecture based on deep unfolding: A network hierarchy based on the iterative computational process of gradient descent is designed, by replacing the iterative computation of channel parameters with dedicated neural network layers. In particular, the nonlinear mapping in the iterative computation is substituted with activation functions in deep learning in each layer. We also introduce learnable parameters into the network, which are updated based on error backpropagation to accelerate the training convergence. The channel estimation parameter learning is accomplished through the iterative calculation process connecting multiple network layers. The deep unfolding network scheme offers better interpretability and ease of optimization and modification compared with traditional closed-box neural networks.

(2) Integrating matrix computation into the deep learning hierarchy: We utilize neural network layers to learn variational parameters, avoiding computationally complex iterative operations such as matrix inversion and differentiation, and enhancing computational efficiency. MUCE is streamlined in network design, with learning-based characteristics requiring only a few layers for convergence. This approach necessitates fewer parameters than existing learning-based algorithms and significantly reduces spatial complexity.

(3) Parameterizing the system model to enhance generalization capabilities: as the number of RAs and BDs increases, computational challenges arise. The method we propose requires only one efficient training session and is applicable to repeated channel estimation of the same scale. Furthermore, MUCE is a model-driven approach that achieves performance comparable to data-driven networks trained with many samples, using only a small number of samples.

C. ORGANIZATION

The paper is structured as follows: Section II delves into existing research on deep unfolding. Section III introduces the monostatic multi-antenna multi-BD backscatter communication network system model, along with its associated channel estimation and information decoding protocol. Section IV is composed into three parts: the first part derives the solution to channel estimation based on the LS algorithm; the second part elaborates on our proposed MUCE channel estimation method and its corresponding network structure; and the third part presents a comparison with a CNN-based channel estimation network for channel estimation. Section V showcases numerical results from simulation experiments, and examines the influence of various parameters on MUCE as well as the performance comparison among the three schemes. Finally, Section VI gives the discussion and conclusion.

II. RELATED WORK

With the rapid development of modern wireless communication technology in the Internet of Things, the focus of current research is on making wireless communication systems more intelligent, enhancing their capabilities to process large volumes of wireless data, and improving their accuracy in identifying and adapting to complex environmental changes. Consequently, numerous studies have applied deep learning to various aspects of physical layer communication [34], network resource allocation, error correction coding, intelligent flow control, and perceptual communication. However, many of these studies focus on data-driven networks, which demand substantial computing resources and time for the training process. Additionally, their generalization capability tends to be insufficient, making them unsuitable for wireless communication applications.

As a result, researchers have introduced model-driven deep learning methods that diverge from purely data-driven approaches. These techniques involve creating network topologies that draw on established domain knowledge, engineering theories, and physical mechanisms. This approach significantly reduces the training cost, both in terms of data and time, compared to data-driven methods. In [35], the authors detail the procedure for developing a model-driven deep learning method. Initially, they construct a function family, also known as a model family, which contains numerous unknown parameters tailored to the specific task context. Subsequently, an algorithm is designed to address the model family problem. Finally, the algorithm is integrated into a deep network to facilitate parameter learning.

The depth of such networks is determined by the algorithm's convergence rate estimation, while the parameter space is defined by parameter constraints. All parameters in the algorithm are learnable, allowing for a variety of algorithms to establish and train the deep network topology through backpropagation. This method circumvents the necessity for exact modelling by counterbalancing inaccuracies in the model and pre-established parameters through the deep network's resilient learning abilities. Additionally, model-driven deep learning's low demand for training data reduces the risk of overfitting and enhances training speed. Deep unfolding serves as a potent example of these model-driven neural networks.

Given a model-based optimization problem and an iterative inference algorithm, deep unfolding [36] performs accurate inference within a fixed network size by disentangling model parameters across layers, introducing a specific number of trainable parameters to expand the iterative inference into neural network layers. Numerous signal processing tasks in wireless communication systems, such as detection and decoding, can be formulated as numerical optimization problems solved through iterative algorithms. Most practical communication applications require high throughput and low latency processing, allowing only a minimal number of algorithm iterations and placing high demands on the iterative algorithm's parameter selection. Deep unfolding leverages deep learning's excellent data-driven learning capabilities and model-driven algorithms' training simplification to efficiently solve such problems, avoiding the stability issues faced by traditional algorithms [37]. As a result, it has been widely adopted by scholars worldwide.

Liao et al. propose a deep neural network for MIMO detection, demonstrating robustness to pathological channels and incorrectly specified noise variance [38]. Later, in [39], the authors developed DetNet, a deep unfolding-based signal detection network, by employing an iterative projected gradient descent algorithm. They have confirmed that DetNet achieves high computational accuracy with low complexity when compared to other methods. The authors of [40] devise a new iterative signal processing algorithm for 6G networks, utilizing deep unfolding techniques to fulfil both service requirements and physical layer specifications envisioned by the 6G communication architecture. This approach aims to enable truly efficient hardware-based edge intelligence for future 6G networks. In the context of beamforming, which calculates the maximum weighted sum rate under power constraints, the authors of [41] suggest an iterative method to map a fixed number of weighted minimum mean square error (WMMSE) algorithms onto a trainable neural network layer. This is based on the principle of deep unfolding, which addresses cellular downlink beamforming problems in networks. Reference [42] proposes an algorithm for jointly designing active and passive beamforming matrices based on stochastic successive convex approximation (SSCA) for intelligent reflector (IRS)-assisted MIMO full-duplex systems. By constructing a neural network, the results achieve significantly outperform the closed-box network used as a benchmark. To tackle massive MIMO channel estimation issues, the authors of [43] introduce a deep unfolding-based unsupervised neural network architecture called mpNet. This can adaptively adjust the network depth according to the input signal-to-noise ratio, allowing for online training and estimation. The authors of [44] model the deep unfolding process as a Markov decision process and established a deep deterministic policy gradient (DDPG)-driven deep adaptive unfolding framework. Simulation results demonstrate that the algorithm considerably reduces the number of layers in the algorithm and performs better than traditional methods.

III. THEORIES AND METHODS

A. SYSTEM MODEL

The monostatic backscatter channel estimation (MBCE) scheme, which consists of a full-duplex reader and M pieces of BDs, is depicted in Fig. 1. In this system, $\mathbf{h}_i (i = 1, 2, \ldots, M)$ represents the channel between different BD $_i$ and the reader R, and h_{AI} represents the ambient interference channel. The reader has N RAs, and each BD is equipped with a single antenna. Typically, backscatter devices remain silent until they receive a communication request. The reader (R) initiates the procedure by sending an activation signal to the surrounding BDs of interest, awakening them from their

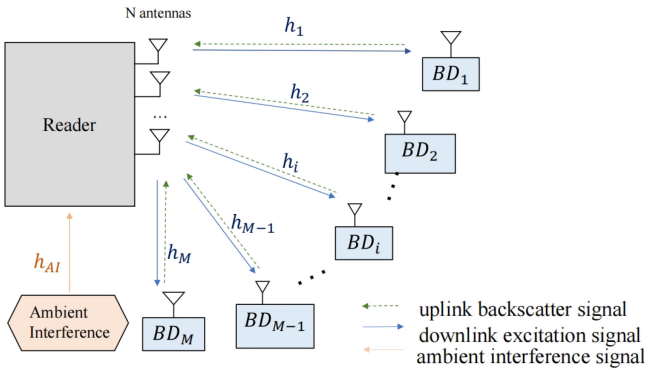


FIGURE 1. Monostatic multi-antenna backscatter communication network with multi-backscatter devices.

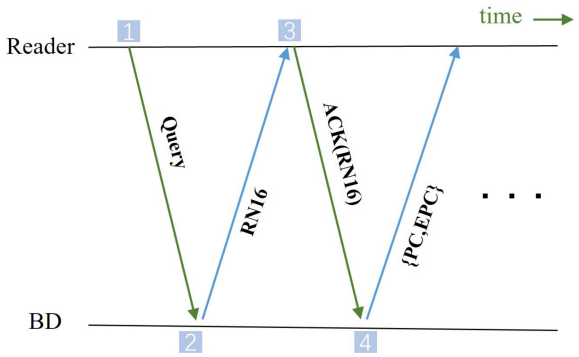


FIGURE 2. EPC global communication cycle.

silent state and preparing them for subsequent backscatter communication. Upon receiving the activation signal, BDs within the reading range respond by sending a 16-bit random number (RN16) to R. When R receives the response, a confirmation signal will be sent back to the BD, prompting the BD to backscatter its respective electronic product code (EPC). The EPC global protocol communication cycle can be seen in Fig. 2. Following this communication establishment process, the R and the BD proceed to conventional channel estimation and backscatter communication.

We use the enhanced multipath channel model presented in [45]. It takes the characteristics of a specific environment and all signal paths between the reader and BDs into account in modelling the passive ultra-high-frequency (UHF) RFID system. According to this model, the backscatter signal received by the reader is not only related to the azimuth and elevation angles of different signal paths, but also to the gain of the BDs. This insight inspires us to consider the effect of the backscatter device's response on the channel when modeling the backscatter channel. In the following section, we will give the BDs response model.

In backscatter communication, the activation of the BDs results in the modulation of its complex baseband signal $x_{\mathrm{BD}} = F - \zeta$ with the reader's carrier signal. Here, F represents a load-independent constant connected to the antenna structure, while $\zeta \in \{\zeta_0, \zeta_1, \zeta_2, \ldots, \zeta_{M-1}\}$ denotes a set of reflection coefficients controlled by the load to achieve desired modulation. For simplicity, let $a_i = |x_{\mathrm{BD_i}}|(i = 1, 2, \ldots, M)$ signify the modulation parameters of the activated M BDs, with a_0 representing the parameters of inactive BDs. In a typical backscatter communication scenario, the values of the two variables are $a_0 \approx 0, a_i \approx 1$.

B. ESTIMATION AND INFORMATION DECODING PROTOCOL

The established research on channel estimation primarily relies on the approximate time-invariant property of the channel within coherence time. For multi-BD MBCE, we propose a sequential estimation strategy analogous to traditional communication preamble design. Specifically, the estimation is conducted sequentially for M BDs, with only BD_i active (denoted by BD_{on}) during the estimation phase τ_{CE_i} of the $i-th$ BD, while other BDs remain silent (denoted by BD_{off}). Furthermore, during communication between R and BD_{on}, the signal received by R includes interference from environmental random noise (collectively referred to as ambient interference), in monostatic systems interference is typically minimal, in addition to the backscatter signal modulated by BD_{on} to downlink excitation signal from R. The presence of such interference severely hinders R's ability to correctly demodulate the backscatter signal. As a result, we design the MUCE channel estimation and information decoding protocol, as illustrated in Fig. 3. Over τ sample durations (a coherence time), the first stage τ_{CE_0} is dedicated to estimating ambient interference. Subsequently, channel estimation between each BD and R occurs sequentially during $\tau_{CE_i}(i = 1, \ldots, M)$, with the remaining time period $\tau - \sum_{i=0}^{M} \tau_{CE_i}$ allocated for signal reception and decoding.

1) AMBIENT INTERFERENCE ESTIMATION

In the multi-antenna monostatic backscatter communication system, R performs channel estimation by transmitting orthogonal pilot signals of length l_0 samples from all N RAs. During this process, all BDs are set to a silent state, with a modulation parameter of a_0. To simplify, we set $l_0 = N$, and the received signal at R during τ_{CE_0} can be described as

$$\mathbf{Y}_{AI} = \left(\mathbf{H}_{AI} + \sum_{k=1}^{M} a_0 \mathbf{h}_k\right)\mathbf{S}_0 + \mathbf{W}_0. \tag{1}$$

where the transmitting pilot signal matrix as $\mathbf{S}_0 \in \mathbb{C}^{N\times N}$, the received signal $\mathbf{Y}_{AI} \in \mathbb{C}^{N\times N}$, $\mathbf{H}_{AI} \in \mathbb{C}^{N\times N}$ denotes ambient interference channel parameters, $\mathbf{h}_k \in \mathbb{C}^{N\times N}$ denotes the backscatter channel parameters of BD_k. Based on channel reciprocity [14], [46], due to characteristic of channel reciprocity, the channel between each BD and R can be represented as $\mathbf{h}_k = \mathbf{h}_{\Delta_k^{UP}}\mathbf{h}^T_{\Delta_k^{UP}}$, with $\mathbf{h}_{\Delta_k^{UP}}$ being the uplink channel parameters from BD_k to R, the proposed method performs channel estimation for ambient interference and

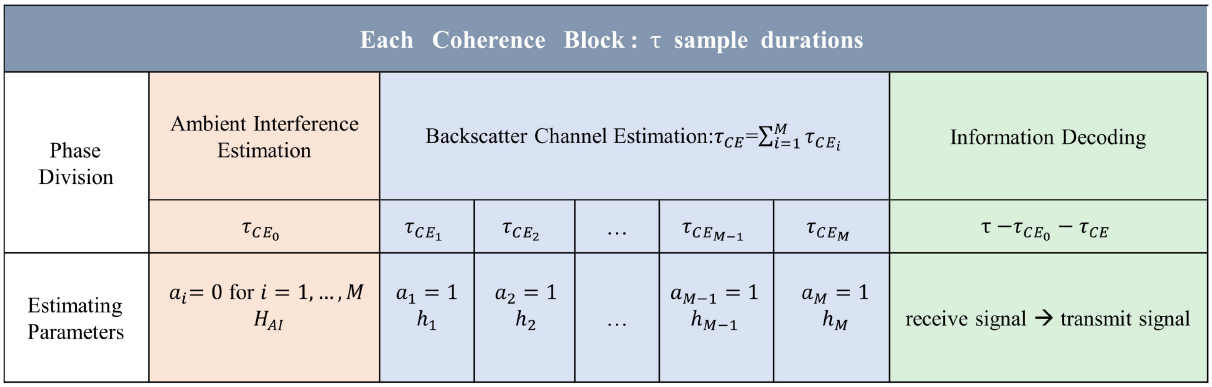


FIGURE 3. The MUCE channel estimation and information decoding protocol.

backscatter channels, followed by signal decoding using the estimated channels, hence, the channel estimation for $\mathbf{h}_{\Delta_k^{UP}}$ does not take into account.

The parameter β_k represents the average channel power gain, which accounts for both fading gain and propagation loss. For Rayleigh channel, this leads to $\mathbf{h}_{\Delta_k^{UP}} \sim \mathcal{CN}(\mathbf{0}_{N\times 1}, \beta_k \mathbf{I}_N)$, $(\forall k = 1, 2, \ldots, M)$, where $\mathbf{I}_N$ represents the identity matrix of $N \times N$. The additive Gaussian white noise (AWGN) is represented by $\mathbf{W}_0 \in \mathbb{C}^{N\times N}$, with $\mathbf{W}_0 \sim \mathcal{CN}(0, \sigma_{W_0}^2 I)$.

Using the LS algorithm, the LS estimate of the interference channel can be expressed as:

$$\hat{\mathbf{H}}_{AI} = \mathbf{Y}_{AI}\mathbf{S}_0^{+} = \mathbf{H}_{AI} + \sum_{k=1}^{M} a_0 \mathbf{h}_k + \frac{\mathbf{W}_0 \mathbf{S}_0^H}{p} = \mathbf{H}_{AI} + \tilde{\mathbf{H}}_{AI}. \tag{2}$$

where $\mathbf{S}_0^{+}$ denotes the pseudo-inverse matrix of the pilot signal, defined as $\mathbf{S}_0^{+} = \mathbf{S}_0^H(\mathbf{S}_0\mathbf{S}_0^H)^{-1}$, and $\tilde{\mathbf{H}}_{AI}$ represents the estimation error of the interfering channel. The power of the transmitting pilot, denoted by p, satisfies $\mathbf{S}_0\mathbf{S}_0^H = p\mathbf{I}_N$.

2) BACKSCATTER CHANNEL ESTIMATION

The second stage involves modeling and estimating the backscatter channel for M BD$_s$. During the estimation phase, R sends the pilot signal $\mathbf{S}_0$ to each BD sequentially. The received signal from the reader in $\tau = \sum_{k=1}^{M} \tau_{CE_k}$ can be represented as:

$$\mathbf{Y}_{CE} = (\mathbf{A} \otimes \mathbf{S}_0)\mathbf{H} + (\mathbf{1}_{M\times 1} \otimes \mathbf{H}_{AI})\mathbf{S}_0 + \mathbf{W}_1, \tag{3}$$

where $\mathbf{Y}_{CE} \in \mathbb{C}^{MN\times N}$, $\mathbf{H} \in \mathbb{C}^{MN\times N}$ is the backscattered channel to be estimated, and $\mathbf{W}_1 \in \mathbb{C}^{MN\times N}$ is the additive Gaussian white noise with $\mathbf{W}_1 \sim \mathcal{CN}(0, \sigma_{W_1}^2 I)$. The operator $\otimes$ denotes the Kronecker product of two matrices, $\mathbf{A}$ is the $M \times M$ combined modulation matrix of all M BD$_s$ and the element in column m of row i in $\mathbf{A}$ is defined as:

$$[\mathbf{A}]_{im} = \begin{cases} a_{i+1} \approx 1, i = m \\ a_0 \approx 0, i \neq m, \end{cases} \quad \forall i, m \in \{0, 1, 2, \ldots, M-1\} \tag{4}$$

where $a_i (\forall i \in \{0, 1, 2, \ldots, M-1\})$ is defined the modulation parameter.

$$\mathbf{H} = \begin{bmatrix} \mathbf{h}_1 & \mathbf{h}_2 & \cdots & \mathbf{h}_M \end{bmatrix}^T, \tag{5}$$

with $\mathbf{h}_i \in \mathbb{C}^{N\times N}$ $(\forall i \in \{0, 1, 2, \ldots, M-1\})$ being the channel parameter between R and BD$_i$.

Based on the estimation of the interference channel $\hat{\mathbf{H}}_{AI}$, the received signal after removing the environmental interference can be expressed as:

$$\mathbf{Y} = \mathbf{Y}_{CE} - \hat{\mathbf{Y}}_{AI} = \mathbf{Y}_{CE} - \left(1_{M\times 1} \otimes \hat{\mathbf{H}}_{AI}\right)\mathbf{S}_0 = (\mathbf{A} \otimes \mathbf{S}_0)\mathbf{H} + \mathbf{W}. \tag{6}$$

where $\mathbf{W} = 1_{M\times 1} \otimes (\sum_{k=1}^{M} a_0 \mathbf{h}_k \mathbf{S}_0 + \mathbf{W}_0) + \mathbf{W}_1$. Therefore, the channel estimation for multi-BD multi-antenna backscatter channels based on the LS algorithm can be modeled as:

$$\mathrm{P}_1 : \underset{\mathbf{h}_1, \mathbf{h}_2, \ldots, \mathbf{h}_m}{\arg\min}\ E(\mathbf{H}) = \|\mathbf{Y} - (\mathbf{A} \otimes \mathbf{S}_0)\mathbf{H}\|^2, \tag{7}$$

We are considering a conventional monostatic multi-antenna backscatter communication system. In order to achieve full-duplex function, each of all N antennas in the reader can transmit a carrier signal to a BD and also receive the backscatter signal from the BD. But as stated in [47], we don't need to consider the self-interference caused by full-duplex. This is because the reader includes a decoupler consisting of an automatic gain control circuit and a conventional phase-locked loop, which can effectively suppress the self-interference carrier by carefully adjusting the underlying phase shifter and attenuator. Consequently, our model remains general.

In the backscatter communication system under discussion, all BDs are considered semi-passive devices equipped with an energy storage module. This module enables low-power onboard operation without the need to await sufficient RF signal acquisition before backscatter, thereby eliminating the need to account for reader access delays to the BD. With the completion of the mathematical modelling for the MBCE, we will present the novel channel estimation algorithm, derived from this model, in the following section.

IV. THE MODEL-DRIVEN UNFOLDED CHANNEL ESTIMATION

Based on the channel estimation scheme MUCE proposed in Section III for a multi-antenna multi-BD mono-static backscatter system considering interference, this section addresses several challenges. As the number of BDs and RAs increases, the dimension of the problem not only raises computational costs but also impacts the efficiency of channel estimation. Furthermore, changes in the surrounding environment will affect the estimation of ambient interference. Therefore, it becomes necessary to adjust the interference estimation model in a timely manner to prevent any adverse effects on the quality of channel estimation. Therefore, we propose a model-driven deep learning with better generalization ability and lower computational cost. In this section, we will introduce the network structure of MUCE and the algorithm model on which the network construction is based.

Firstly, (6) can be rewritten as:

$$\mathbf{Y} = \mathbf{S}_0'\mathbf{H} + \mathbf{W}. \tag{8}$$

where, the receiving signal is represented by $\mathbf{Y} = \begin{bmatrix}\mathbf{y}_1 & \mathbf{y}_2 & \cdots & \mathbf{y}_M\end{bmatrix}^T$, with $\mathbf{Y} \in \mathbb{C}^{MN\times N}$ and $\mathbf{y}_k(\forall k = 1,2,\ldots,M) \in \mathbb{C}^{N\times N}$, the noise $\mathbf{W} = \begin{bmatrix}\mathbf{w}_1 & \mathbf{w}_2 & \cdots & \mathbf{w}_M\end{bmatrix}^T$, with $\mathbf{w}_k(\forall k = 1,2,\ldots,M) \in \mathbb{C}^{N\times N}$ and $\mathbf{W} \in \mathbb{C}^{MN\times N}$.

The joint pilot matrix $\mathbf{S}_0'$ can be extended as:

$$\mathbf{S}_0' = \mathbf{A}\otimes\mathbf{S}_0 = \begin{bmatrix} a_1\mathbf{S}_0 & \mathbf{0}_{N\times N} & \cdots & \mathbf{0}_{N\times N} & \mathbf{0}_{N\times N} \\ \mathbf{0}_{N\times N} & a_2\mathbf{S}_0 & \cdots & \mathbf{0}_{N\times N} & \mathbf{0}_{N\times N} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \mathbf{0}_{N\times N} & \mathbf{0}_{N\times N} & \cdots & a_{M-1}\mathbf{S}_0 & \mathbf{0}_{N\times N} \\ \mathbf{0}_{N\times N} & \mathbf{0}_{N\times N} & \cdots & \mathbf{0}_{N\times N} & a_M\mathbf{S}_0 \end{bmatrix} \in \mathbb{C}^{MN\times MN} \tag{9}$$

To solve (7), we can utilize gradient descent. Given that $\mathbf{h}_k(\forall k = 1,2,\ldots,M)$ is uncorrelated with each other, the complex multivariate gradient descent is not needed. Instead, we can transform it into a univariate gradient descent through matrix vectorization. In the following, we will first perform an equivalent vectorization operation on (8). Then, we will derive an iterative gradient descent algorithm based on the vectorized expression.

A. EQUIVALENT VECTORISATION

For ease of representation during vectorization, we propose the following definition: for a matrix $\mathbf{B}$, $\mathbf{B}(i;)$ represents the i_{th} row, while $\mathbf{B}(;j)$ denotes the j_{th} column. Consequently, we can vectorize (8) as:

$$\mathbf{y}_{vec} = \mathbf{S}\mathbf{h}_{vec} + \mathbf{w}_{vec}, \tag{10}$$

In (10), $\mathbf{S}$ represents the new pilot matrix with modulation parameters of BDs, which is obtained as:

$$\mathbf{S} = \begin{bmatrix} \mathbf{S}^{1,1} & \mathbf{0}_{N\times N^2} & \cdots & \mathbf{0}_{N\times N^2} & \mathbf{0}_{N\times N^2} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \mathbf{S}^{1,N} & \mathbf{0}_{N\times N^2} & \cdots & \mathbf{0}_{N\times N^2} & \mathbf{0}_{N\times N^2} \\ \mathbf{0}_{N\times N^2} & \mathbf{S}^{2,1} & \cdots & \mathbf{0}_{N\times N^2} & \mathbf{0}_{N\times N^2} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \mathbf{0}_{N\times N^2} & \mathbf{S}^{2,N} & \cdots & \mathbf{0}_{N\times N^2} & \mathbf{0}_{N\times N^2} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \mathbf{0}_{N\times N^2} & \mathbf{0}_{N\times N^2} & \cdots & \mathbf{0}_{N\times N^2} & \mathbf{S}^{M,1} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \mathbf{0}_{N\times N^2} & \mathbf{0}_{N\times N^2} & \cdots & \mathbf{0}_{N\times N^2} & \mathbf{S}^{M,N} \end{bmatrix} \in \mathbb{C}^{MN^2\times MN^2}, \tag{11}$$

where:

$$\mathbf{S}^{i,j} = a_i\mathbf{S}_0(j;) = \begin{bmatrix} a_i\mathbf{S}_0(j;) & \mathbf{0}_{1\times N} & \cdots & \mathbf{0}_{1\times N} \\ \mathbf{0}_{1\times N} & a_i\mathbf{S}_0(j;) & \cdots & \mathbf{0}_{1\times N} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0}_{1\times N} & \mathbf{0}_{1\times N} & \cdots & a_i\mathbf{S}_0(j;) \end{bmatrix} \in \mathbb{C}^{N\times N^2}, \tag{12}$$

Moreover, $\mathbf{y}_{vec} = vec\{\mathbf{Y}\} = [\mathbf{y}^1 \quad \mathbf{y}^2 \quad \cdots \quad \mathbf{y}^M]^T$. is an $MN^2\times 1$ column vector, with $\mathbf{y}^i(\forall i \in 1,2,\ldots,M)$ being an $N^2\times 1$ column vector. The latter is obtained by splicing $\mathbf{y}_i$ horizontally by row and then transposing it:

$$\mathbf{y}^i = \begin{bmatrix}\mathbf{y}_i(1;), & \mathbf{y}_i(2;), & \ldots, & \mathbf{y}_i(N;)\end{bmatrix}^T, \tag{13}$$

Similarly, $\mathbf{w}_{vec} = vec\{\mathbf{W}\} = [\mathbf{w}^1\ \mathbf{w}^2 \quad \cdots \quad \mathbf{w}^M]^T$ is an $MN^2\times 1$ column vector, with $\mathbf{w}^i(\forall i \in 1,2,\ldots,M)$ being an $N^2\times 1$ column vector. This vector is obtained by splicing $\mathbf{w}_i$ horizontally by row and then transposing it:

$$\mathbf{w}^i = \begin{bmatrix}\mathbf{w}_i(1;), & \mathbf{w}_i(2;), & \ldots, & \mathbf{w}_i(N;)\end{bmatrix}^T, \tag{14}$$

Similarly, the vectorization of the channel parameters to be estimated is $\mathbf{h}_{vec} = vec\{\mathbf{H}\} = [\mathbf{h}^1 \quad \mathbf{h}^2 \quad \cdots \quad \mathbf{h}^M]^T$, which is a column vector of $MN^2\times 1$. Furthermore, $\mathbf{h}^i(\forall i \in 1,2,\ldots,M)$ is a column vector of $N^2\times 1$, and it is obtained by concatenating the columns of each channel parameter matrix vertically as:

$$\mathbf{h}^j = \begin{bmatrix}\mathbf{h}_j(;1) & \mathbf{h}_j(;2) & \cdots & \mathbf{h}_j(;N)\end{bmatrix}^T. \tag{15}$$

B. COMPLEX-TO-REAL

In this part, we design the network hierarchy and overall architecture based on the model given above. In the previous part, we completed the vectorization of (8). It is important to note that the transmit signal, receive signal, and channel parameters mentioned above are all in the complex domain. To facilitate processing with gradient descent, we first

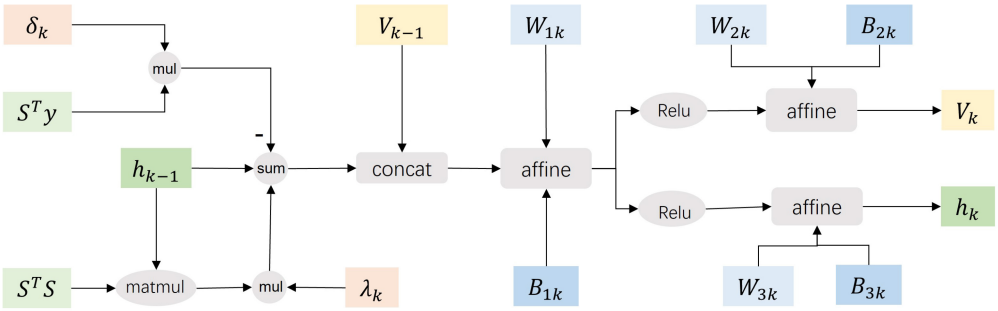


FIGURE 4. The detailed structure of each layer in MUCE.

transform them into the real domain. Specifically, in the neural network, the real part and the imaginary part of the complex signal are taken, and then the real part and the imaginary part are spliced by columns/rows as [48]

$$\mathbf{y}' = \begin{bmatrix} \Re(\mathbf{y}_{vec}) \\ \Im(\mathbf{y}_{vec}) \end{bmatrix} \in \mathbb{R}^{2MN^2 \times 1}, \tag{16a}$$

$$\mathbf{h}' = \begin{bmatrix} \Re(\mathbf{h}_{vec}) \\ \Im(\mathbf{h}_{vec}) \end{bmatrix} \in \mathbb{R}^{2MN^2 \times 1}, \tag{16b}$$

$$\mathbf{S}' = \begin{bmatrix} \Re(\mathbf{S}) & -\Im(\mathbf{S}) \\ \Im(\mathbf{S}) & \Re(\mathbf{S}) \end{bmatrix} \in \mathbb{R}^{2MN^2 \times 2MN^2}, \tag{16c}$$

where $\Re(\cdot)$ and $\Im(\cdot)$ denote the real and imaginary parts of complex numbers, while $\mathbf{y}'$, $\mathbf{h}'$ and $\mathbf{S}'$ are the results of realization of $\mathbf{y}_{vec}$, $\mathbf{h}_{vec}$ and $\mathbf{S}$ of (10), respectively. Since our labels have undergone the same equivalent real-field conversion, the loss can be calculated directly using the separated real and imaginary components, without the need to revert to complex numbers. The final result is also stored in the real-number form, as shown in the formula above.

With the result of the above equivalent, the real number transformation is as the input to the network. Thus, (7) can be reformulated as follows:

$$\mathrm{P}_2: \ \arg\min_{\mathbf{h}_1, \mathbf{h}_2, \ldots, \mathbf{h}_m} E(\mathbf{h}') = \left\| \mathbf{y}' - \mathbf{S}'\mathbf{h}' \right\|^2. \tag{17}$$

C. PROPOSED UNFOLDING ARCHITECTURE

As stated in Section II, given a model-based optimization problem, deep unfolding can perform inference across layers and solve for relevant parameters. Therefore, we design an iterative process based on projected gradient descent [49] for the problem in (17) as:

$$\begin{aligned} \hat{\mathbf{h}}'_{\mathrm{k}} &= \prod \left[\hat{\mathbf{h}}'_{k-1} - \beta_k \frac{\partial \left\| \hat{\mathbf{y}}' - \mathbf{y}' \right\|^2}{\partial \mathbf{h}'} \Big|_{\mathbf{h}' = \hat{\mathbf{h}}'_{k-1}} \right] \\ &= \prod \left[\hat{\mathbf{h}}'_{\mathbf{k}-1} - \beta_k \mathbf{S}'^{\mathrm{T}} \mathbf{y}' + \beta_k \mathbf{S}'^{\mathrm{T}} \mathbf{S}' \hat{\mathbf{h}}'_{k-1} \right]. \end{aligned} \tag{18}$$

where $\hat{\mathbf{y}}' = \mathbf{S}'\hat{\mathbf{h}}'_{k-1}$, and $\hat{\mathbf{h}}'_{\mathrm{k}}$ is the estimated value of the k_{th} iteration. The output of the last iteration is the channel parameter sought, and the initial iteration value is set to 0. $\mathbf{S}'$ and $\mathbf{y}'$ are transmitted pilot signal matrix and the received signal vector in (16), respectively. The nonlinear projection operator is represented as $\prod[\cdot]$, and β_k is the step factor.

In Section II, we examine the model-driven deep unfolding technique extensively employed in wireless communication and investigate its applications within the field. As previously mentioned, deep unfolding enables the construction of neural networks that solve required parameters, and leverage a priori knowledge of wireless communication systems. This approach eliminates the need for repetitive iterative operations found in traditional parameter estimation methods and also addresses the "closed box" issue in deep learning.

Consequently, we introduce MUCE, which integrates the deep unfolding and the LS parameter solution using gradient descent. Initially, we establish a single-layer network, as illustrated in Fig. 4, grounded in the iterative computation process detailed in (18). We then employ an activation function within the neural network to replace non-linear mapping to the output. Additionally, we incorporate learnable parameters into the network and adjust these parameters based on error gradient backpropagation.

The forward propagation error is calculated for each layer of the network in the figure using the following equations:

$$\mathbf{z}_k = \mathrm{ReLU}\left(W_{1k} \begin{bmatrix} \hat{\mathbf{h}}'_{k-1} - \delta_k \mathbf{S}'^T \mathbf{y} + \lambda_k \mathbf{S}'^T \mathbf{S}' \hat{\mathbf{h}}'_{k-1} \\ \mathbf{V}_{k-1} \end{bmatrix} + \mathbf{B}_{1k} \right). \tag{19}$$

where $\mathbf{V}_k = f(\mathbf{W}_{2k}\mathbf{z}_k + \mathbf{B}_{2k})$, $\mathbf{h}_k = g(\mathbf{W}_{3k}\mathbf{z}_k + \mathbf{B}_{3k})$, $\mathbf{V}_0 = \mathbf{0}$ and $\mathrm{ReLU}(x) = \max\{0, x\}$. $f(\bullet)$ and $g(\bullet)$ represent different nonlinear activation functions, respectively. $\mathbf{V}$ are the auxiliary training variables, which are incorporated to increase the width of the network for improving training performance. Furthermore, both $\hat{\mathbf{h}}'_0$ and $\hat{\mathbf{V}}'_0$ are initialized to $\mathbf{0}$. The inputs for the current k_{th} layer include $\mathbf{S}'^T\mathbf{y}'$, $\mathbf{S}'^T\mathbf{S}'$, and the output $\hat{\mathbf{h}}'_{k-1}$ from the previous layer. The trainable parameter introduced to the network is $\boldsymbol{\theta} = \{\mathbf{W}_{1k}, \mathbf{W}_{2k}, \mathbf{W}_{3k}, \mathbf{B}_{1k}, \mathbf{B}_{2k}, \mathbf{B}_{3k}, \delta_k, \lambda_k\}_{k=1}^{L}$.

The size of auxiliary training variable $\mathbf{V}$ and $\mathbf{B}$ is $2MN^2 \times 1$, and variable $\mathbf{W}$ is $2MN^2 \times 4MN^2$. The computational complexity is $\mathcal{O}(M^3N^6)$, where $\mathcal{O}(\cdot)$ represents the asymptotic upper bound of computational complexity. The width of the deep neural network determines the amount of feature information extracted from each layer. The more feature information the easier it is to train the network.

The loss function employs a logarithmically weighted approach, taking into account the output of each layer. The

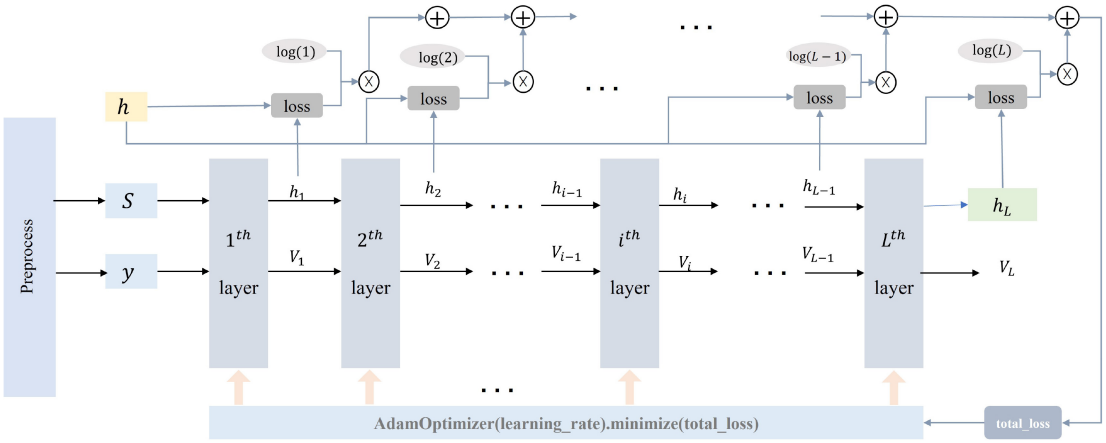


FIGURE 5. The architecture of the MUCE network for backscatter channel estimation.

closer the layer to the output, the higher the proportion of the loss function given as:

$$\text{Loss}\left(\mathbf{h}', \hat{\mathbf{h}}'(\mathbf{S}', \mathbf{y}'; \boldsymbol{\theta})\right) = \sum_{k=1}^{L} \log(l) \left\| \mathbf{h}' - \hat{\mathbf{h}}'_k \right\|^2. \quad (20)$$

We simulate the iterative calculation process by connecting the single-layer network depicted in Fig. 4, for obtaining the channel parameters estimation after iterating through L layers ultimately. Fig. 5 displays the comprehensive network structure of MUCE, which employs the Adam optimizer for error backpropagation and parameter tuning based on the loss function. The layer named "Preprocess" stands for the operations described in Section IV.A 1) and 2). In addition, to ensure the conciseness of the schematic, the superscript of the corresponding variables in (19) and (20) have been removed, and the data used in the actual network training is still the result in (16). The design of this network reduces computational complexity and enhances channel estimation efficiency by eliminating the need for complex matrix inversion operations in the LS algorithm. Consequently, the network can adapt to changing channels, allowing it to estimate channel parameters based on current transmit and receive signals even when the channel state varies.

The channel is measured using the normalized mean squared error (NMSE), as demonstrated in (21):

$$\text{NMSE} = \frac{1}{\text{M}} \sum_{\text{m}=1}^{\text{M}} \frac{\left\| \hat{\mathbf{h}}_{\text{m}} - \mathbf{h}_{\text{m}} \right\|^2}{\left\| \mathbf{h}_{\text{m}} \right\|^2}. \quad (21)$$

Here, $\mathbf{h}_m$ and $\hat{\mathbf{h}}_m$ represent the true channel parameters and the estimated channel parameters by MUCE between BD_m and R, respectively.

V. NUMERICAL RESULTS AND DISCUSSION

In this section, the effectiveness of the proposed MUCE scheme is assessed. Subsequently, the impact of various network architectures on channel estimation performance is evaluated. Finally, the LS algorithm, MUCE, and Backscatter Channel Estimation Convolutional Neural Network (BCECNN) are deployed for channel estimation on an identical dataset for comparison. The hardware used for the simulations is a desktop equipped with an Intel i7-10700K CPU, an NVIDIA GeForce RTX 3080 GPU and 64GB RAM. And we implement the proposed framework in Python 3.6.13 with Tensorflow 2.3.0.

A. SIMULATION SETUP

In the simulation, we set the transmit power of the reader at $p = 30$ dBm, while the variances of $\mathbf{W}_1$ and $\mathbf{W}_2$ are both assigned as $\sigma^2_{W_0} = -20$ dBm and $\sigma^2_{W_1} = -20$ dBm, respectively. We model all links as being subject to flat quasi-static Rayleigh block fading. Consequently, the channel parameter β_i is given by $(\frac{3\times10^8}{4\pi f})^2 d_i^{-\varrho} (\forall i \in 1, 2, \ldots, M)$. Unless otherwise specified, the transmit frequency and path loss exponent of the system default as $f = 915$ MHz and $\varrho = 3$, respectively. Furthermore, we assume that M BDs are randomly distributed within a circular area of radius $R = 100$m centered on the reader. The coordinates of each BD_i are denoted as (x_i, y_i), and the distance between BD_i and the reader, d_i, is calculated as $\sqrt{x_i^2 + y_i^2}$. Lastly, the modulation parameters $a_i (\forall i \in 1, 2, \ldots, M)$ for backscatter device BD_i at activation are set randomly, following a uniform distribution $U(0.9, 1)$, while the silent modulation parameter a_0 is set to 0.01.

B. IMPACT OF KEY PARAMETERS ON MUCE

In this section, we assess the performance of MUCE by varying network parameters, such as learning rate and number of layers, as well as system model settings, such as the number of RAs and BDs. To streamline our presentation, we use *lr* for the learning rate, *iter* for total training iterations, and L for the number of network layers. The results presented below are obtained from a training set containing 6000 samples and a validation set of 2000 samples, which are randomly generated from the established system model. Unless otherwise stated, these simulations were conducted

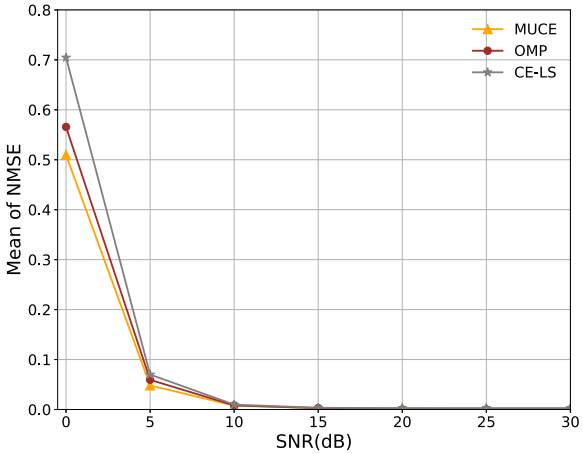


FIGURE 6. Mean of NMSE of three methods with different *SNR*.

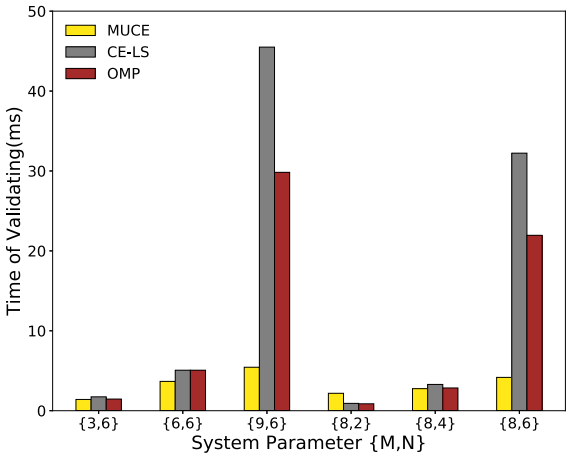


FIGURE 7. The validation time of three methods with different *M*, *N*.

on a desktop featuring an Intel i7-10700K CPU, an NVIDIA GeForce RTX 3080 GPU and 64 GB RAM.

To compare the average performance with other traditional methods, the mean NMSE for different SNRs is shown in Fig. 6, with parameters $M = 12, N = 6, L = 3$, and $iter = 100$. As shown in Fig. 6, the proposed MUCE method performs better than the LS and OMP methods. To compare the robustness of validation time for different groupings of M, N, Fig. 7 shows that when either M or N increases, unlike the LS and OMP methods, the MUCE method can maintain stable validation time, demonstrating better robustness than traditional methods.

The cumulative distribution function (CDF) of channel estimation NMSE for different learning rates is shown in Fig. 8, with parameters $M = 3, N = 6, L = 5$, and $iter = 100$. As shown in Fig. 8, when the learning rate is 1e−06, it is too small, causing the model to get stuck in a local optimum. When the learning rate is 1e−04, it is too large, leading the model to oscillate around the optimal solution. At a learning rate of 1e−05, the model can converge quickly and steadily to the optimal solution of the loss function.

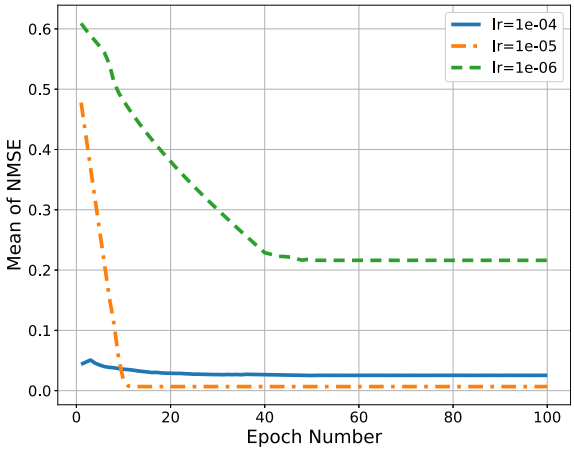


FIGURE 8. NMSE with different *lr*. ($M = 3$, $N = 6$, $L = 5$, $iter = 100$).

TABLE 1. The performance of different *lr*.

	$lr = 1e-4$	$lr = 1e-5$	$lr = 1e-6$
mean of NMSE	0.032	0.007	0.176
max of NMSE	0.642	0.013	1.883
min of NMSE	0.0056	0.0018	0.0825
training time(s)	5911.5	5974.7	5908.3
validation time(ms)	1.407	1.45	1.411

As shown in Fig. 8, the proportion of samples with NMSE less than 0.5 increases from 60% to 80% when lr decreases from 0.001 to 0.00001 and the samples with NMSE less than 0.5 also exceed 95%. However, as lr continues to decrease, the estimated performance worsens. To facilitate a more intuitive comparison of performance, we summarize the mean, maximum, and minimum NMSE values for each learning rate, as well as the network training and validation times, in Table 1, with the minimum value for each metric highlighted in a red box. As observed in the table, the smallest difference between the maximum and mean estimation NMSE values occurs at lr =1e−05, where the minimum is also notably smaller than that at lr =1e−06. In addition, the learning rate affects the speed of convergence of the network, which in turn affects the training time of the network. The penultimate row in Table 1 reveals that the training time is shortest when the learning rate is lr =1e−6. Thus, the optimal learning rate for estimation is set to be lr =1e−6 [50], [51], [52].

The empirical CDF of the channel estimation NMSE for different numbers of layers is depicted in Fig. 9 with parameters set at $M = 12$, $N = 6$, $lr = 0.0001$, and $iter = 100$. As shown in Fig. 9, when the NMSE increases the estimated performance is increasing first and then tend to be stable, and the estimated performance does not improve as L continues to increase. The proportion declines when

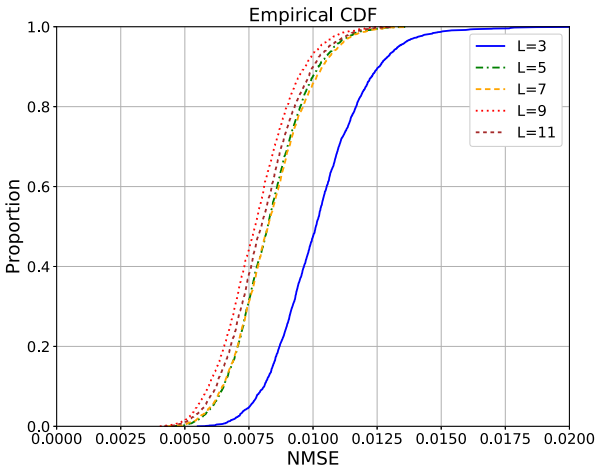


FIGURE 9. NMSE with different L. ($M = 12$, $N = 6$, $lr = 0.0001$, $iter = 100$).

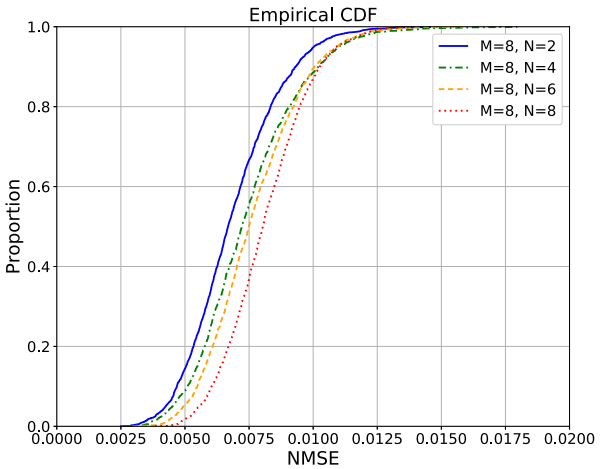


FIGURE 10. NMSE with different N. ($M = 8$, $L = 5$, $lr = 0.0001$, $iter = 100$).

TABLE 2. The performance of different L.

	$L=3$	$L=5$	$L=7$	$L=9$	$L=11$
mean of NMSE	0.01023	0.00830	0.00833	0.00778	0.00805
max of NMSE	0.02032	0.01371	0.01369	0.01300	0.01311
min of NMSE	0.00546	0.00451	0.00434	0.00402	0.00419
training time(s)	2615.50	3755.78	5182.09	6420.50	7991.05
validation time(ms)	4.65	5.62	6.19	7.44	8.36

TABLE 3. The performance of different N.

	$N=2$	$N=4$	$N=6$	$N=8$
mean of NMSE	0.00687	0.00745	0.00767	0.00812
max of NMSE	0.01459	0.01797	0.01583	0.01392
min of NMSE	0.00246	0.00267	0.00296	0.00383
training time(s)	5676.42	5751.41	6424.57	9640.40
validation time(ms)	2.18	2.76	4.17	10.05

L increases from 3 to 11, while it remains approximately the same for $L = 5$, 7, and 11. An analysis of several cases provided in Table 2 reveals that the difference between maximum and minimum values is more prominent for smaller numbers of layers, indicating that the estimation performance is less stable. Additionally, both the training and validation time for the network significantly increase with a greater number of layers. By conducting a comprehensive comparison of the results across varying parameters, the optimal choice for achieving better performance and reduced time costs at $M = 12$ and $N = 6$, as the inherent uncertainty of deep learning networks, coupled with the risk of over fitting when layers are excessively numerous and under training when they are insufficient, contributes to the non-monotonic trend observed in the estimated performance as the number of layers is varied. For our method, the optimal number of layers is determined to be $L = 9$.

The variation of the CDF of the NMSE with respect to the number of BDs is depicted in Fig. 10, while Table 3 presents several essential metrics. Model parameters are set as follows: $M = 8, L = 5, lr = 0.0001$, and $iter = 100$. We observe that when M remains constant, an increase in N leads to a decline in the proportion. Besides the mean NMSE displayed in Table 3 that demonstrates the aforementioned changes, the training and validation times listed in the final two rows also reveal the substantial rise in computational cost correlated with the expansion of the system model.

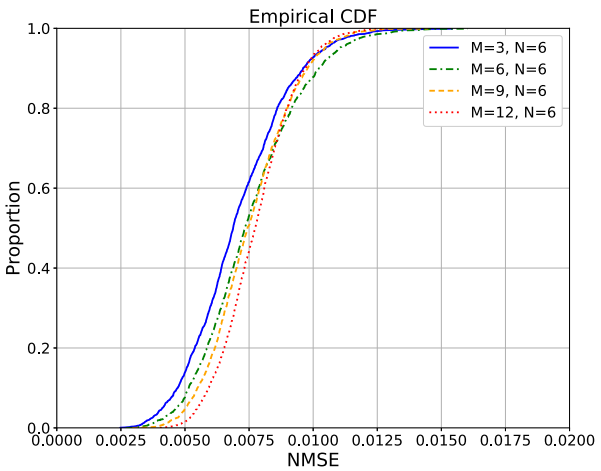


FIGURE 11. NMSE with different M. ($N = 6$, $L = 5$, $lr = 0.0001$, $iter = 100$).

The channel estimation error increases with the number of BDs M as illustrated in Fig. 11, which mirrors the effect caused by the increase in N in Figure 10. By analyzing the variation in the four curves displayed in the figure, along with the first three rows of data in Table 4, it's obvious to see that the estimation performance deteriorates progressively as M grows from 3 to 12 while keeping parameters constant as $N = 6, L = 5, lr = 0.0001, iter = 100$. This is an expected

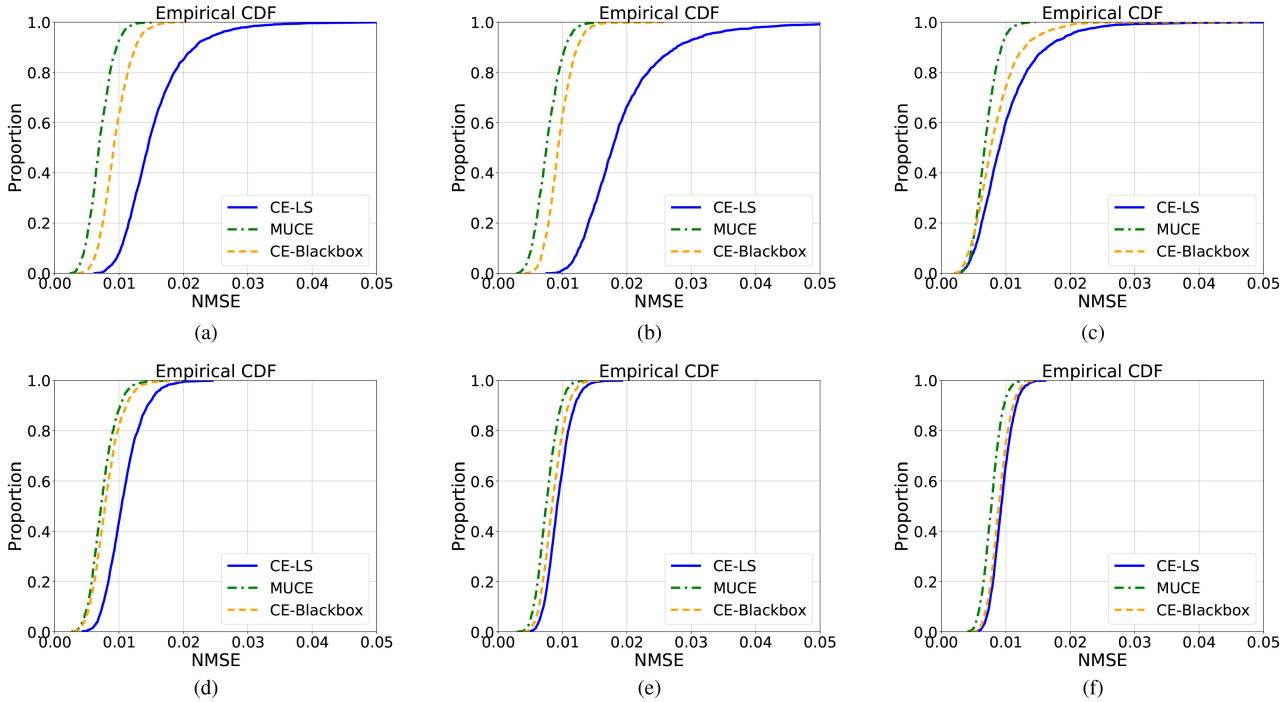


FIGURE 12. The NMSE Empirical CDF of three methods. (a) $M = 3$, $N = 6$. (b) $M = 6$, $N = 6$. (c) $M = 8$, $N = 2$. (d) $M = 8$, $N = 4$. (e) $M = 9$, $N = 6$. (f) $M = 12$, $N = 6$.

TABLE 4. The performance of different M.

	$M = 3$	$M = 6$	$M = 9$	$M = 12$
mean of NMSE	0.00706	0.00755	0.00757	0.00778
max of NMSE	0.01543	0.01603	0.01449	0.01300
min of NMSE	0.00247	0.00271	0.00294	0.00402
training time(s)	4976.43	5841.10	6168.37	6420.50
validation time(ms)	1.41	3.67	5.44	7.44

TABLE 5. The mean values of NMSE with different $\{M, N\}$.

$\{M, N\}$	$\{3, 6\}$	$\{6, 6\}$	$\{8, 2\}$	$\{8, 4\}$	$\{9, 6\}$	$\{12, 6\}$
CE-LS	0.01540	0.01941	0.01013	0.01071	0.00936	0.00938
MUCE	0.00706	0.00755	0.00687	0.00745	0.00757	0.00778
CE-Bb	0.00944	0.00957	0.00834	0.00807	0.00853	0.00898

outcome, as a larger M value results in higher channel dimensionality and a significantly greater number of parameters to estimate. Consequently, the network training time and validation time also experience substantial increases.

C. COMPARISON WITH BCECNN

In the previous section, we assessed the estimation performance of MUCE under various parameter settings to determine the optimal L and lr values for achieving the best performance in MBCE with differing M and N. In this section, we conduct channel estimation while keeping both M and N constant, and analyze the results of the three methods proposed in Section IV. For ease of reference, we will use MUCE, and BCECNN to denote each of the two methods in subsequent content and graphs.

MUCE and BCECNN are learning-based methods, where different network settings significantly impact channel estimation performance, necessitating the tuning of network parameters for improved results. On one hand, as demonstrated in the previous section, the number of layers influences the network's training time. On the other hand, it is evident that iterative training rounds also affect training time, with more rounds requiring more time for training. To ensure a fair comparison, we restrict *iter* to 100 and L to 6, while separately adjusting other parameters for MUCE and BCECNN. We then compare the estimation results of both methods.

After the validation process, we selected six distinct combinations of M and N parameters for comparison and performance analysis. Fig. 12 displays the CDF of channel estimation for the three methods across each of the six parameter sets in separate subplots. As depicted in Fig. 12, the overall distribution of NMSE for the MUCE method surpasses the performance of BCECNN for all six settings.

Table 5 illustrates the differences between the maximum and minimum NMSE values for different M, N combinations among the three methods. The table highlights the maximum difference under each of the six parameter sets

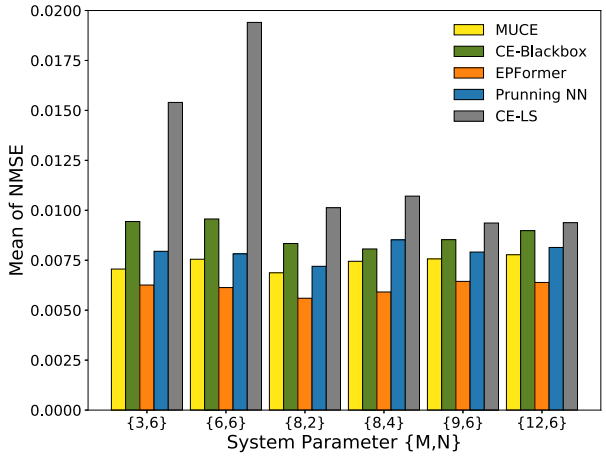


FIGURE 13. Mean of NMSE of five methods with different M, N.

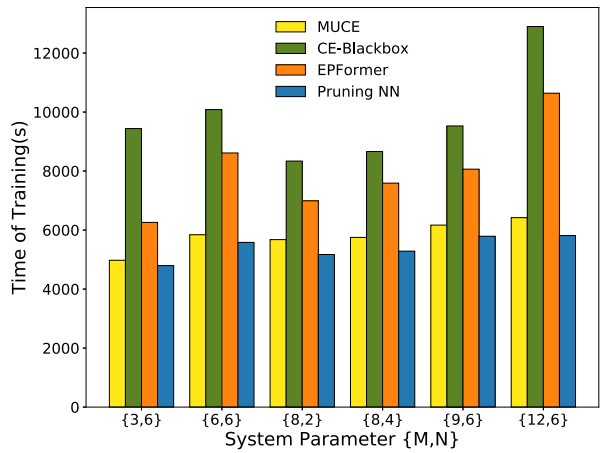


FIGURE 14. The training time of four methods with different M, N.

with red boxes. The results indicate that CE-LS has the most significant performance gap in five out of the six sets, while BCECNN has the largest performance gap in the remaining set. Furthermore, the performance gap of the MUCE method is considerably smaller compared to the other two methods, demonstrating its superior performance in estimating stability.

To compare the average performance with other advanced deep learning methods, we present the mean NMSE of each method in Fig. 13. The figure demonstrates that our MUCE method achieves comparable channel estimation performance to EPFormer [50], while outperforming BCECNN, PruningCNN [51], and LS. In addition to performance, our purpose is to reduce computational time, which is an essential consideration for channel estimation. Fig. 14 displays the validation time for each method using a bar chart on the left y-axis, while the network training time is illustrated with a line chart on the right y-axis. BCECNN and EPFormer consistently take longer than MUCE, while PruningCNN has similar training time but worse performance. With focusing on the training time for BCECNN and MUCE, the line graph in Fig. 14 demonstrates that BCECNN generally demands more time, with the gap widening as M and N increase. From this perspective, MUCE holds a substantial advantage in computational time compared to the two alternative methods. MUCE is a learning-based approach that involves the computation of multi-layer networks when performing parameter estimation. For cases with a large number of BDs and RAs, there has been a significant computational efficiency improvement over conventional methods such as L. For smaller system settings such as $M = 3, N = 6$, traditional methods involve less computation and less time. This is also what we mentioned in the contribution, and our method is important to solve the channel estimation challenges caused by the increase in the number of RAs and BDs.

By synthesizing the preceding validation and analysis results, the MUCE method emerges as an effective backscatter channel estimation technique that balances high performance with low computational overhead. MUCE outperformed traditional least squares (LS) and CNN-based approaches in terms of computational efficiency and estimation accuracy, while also demonstrating strong generalization capabilities. Specifically, MUCE reduces computational costs by up to 83% compared to LS methods, with only minimal accuracy loss.

VI. CONCLUSION

In this paper, we propose a novel channel estimation scheme for monostatic backscatter communication systems that incorporates ambient interference cancellation. Our approach extends the gradient descent-based parameter iterative algorithm to a hierarchical neural network structure, by updating parameters and solving problems based on channel estimation errors. The results demonstrate that our method, which is referred to as MUCE, successfully blends the benefits of the traditional LS algorithm and closed-box neural networks, for achieving an optimal balance between computational cost and performance. To set the number of BDs $M = 12$ and the number of RAs $N = 6$, our novel channel estimation scheme reduces computational time costs by approximately 83% compared to the conventional LS algorithm, while delivering marginally improved estimation performance. Additionally, the MUCE significantly outperforms the closed-box networks in terms of network training efficiency. When $M = 9$ and $N = 6$, the training time required by the proposed method is only 63.8% of that needed by the closed-box approach, without a decline in estimation performance. Furthermore, the MUCE is more explainable and optimizable than deep learning-based closed-box networks, which tend to be more data-driven than model-driven in their network design. In future work, we will take channel capacity, precoding, and other techniques into account to optimize our approach.

APPENDIX

A. LS CHANNEL ESTIMATION ALGORITHM

The Appendix to this section describes the LS algorithm we used for comparison.

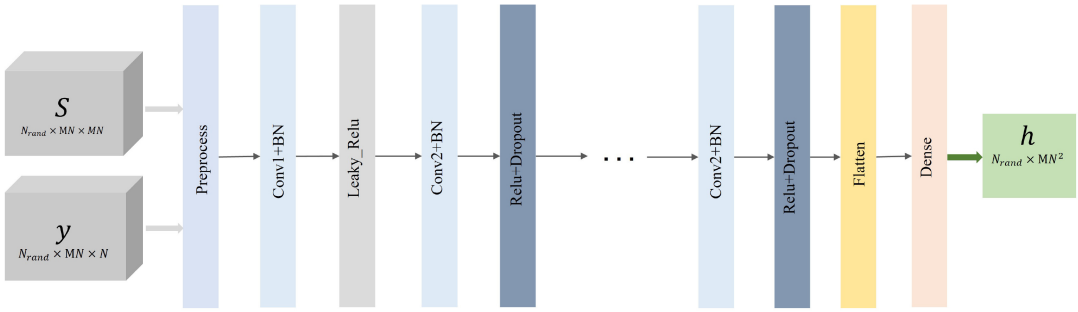


FIGURE 15. The architecture of the BCECNN network, where the layer named "Preprocess" is the same as that of Fig. 5.

For channel estimation, the essence of channel estimation is to determine the channel parameters. Let the receiving observation vector be denoted by $\mathbf{y}$. The channel estimation problem can then be represented in matrix form as

$$\mathbf{y} = \mathbf{S}\mathbf{h} + \mathbf{w}. \tag{22}$$

where $\mathbf{w}$ represents the noise observation vector, $\mathbf{S}$ is the pilot signal transmitted by the reader, and $\mathbf{h} = [h(0), h(1), \ldots, h(L-1)]$ is the channel parameter to be determined. The LS algorithm is widely used for parameter estimation, and the primary goal of the LS algorithm is to minimize the estimation error. Let $\hat{\mathbf{h}}$ denotes the estimated channel parameters, then the entire channel estimation problem can then be modeled as follows:

$$\hat{\mathbf{h}} = \underset{\hat{\mathbf{h}}}{argmin} \left\| \mathbf{y} - \mathbf{S}\hat{\mathbf{h}} \right\|^2, \tag{23}$$

According to [53], the solution of the LS is obtained as

$$\hat{\mathbf{h}}_{\mathbf{LS}} = \left(\mathbf{S}^{\mathbf{H}}\mathbf{S}\right)^{-1}\mathbf{S}^{\mathbf{H}}\mathbf{y}. \tag{24}$$

This leads to the LS solution to the problem described in (17) being as

$$\hat{\mathbf{h}}'_{\mathbf{LS}} = \left(\mathbf{S}'^{\mathbf{H}}\mathbf{S}'\right)^{-1}\mathbf{S}'^{\mathbf{H}}\mathbf{y}'. \tag{25}$$

The LS requires relatively high-complexity computations, such as matrix inversion. The computational complexity based on the LS algorithm is $\mathcal{O}(M^3N^6) + \mathcal{O}(M^4N^8)$.

B. BCECNN FOR CONTRAST

The semi-closed solution based on the LS algorithm and the MUCE based on deep unfolding for channel estimation has been presented in the previous part. Since MUCE employs a deep learning approach, we developed a CNN-based benchmark for channel estimation called BCECNN. This network takes transmitted signal $\mathbf{S}$ and received signal $\mathbf{y}$ as inputs. These inputs are first converted from complex to real domain through a signal processing layer, and then tensor-dimensional transformations are performed to meet the input requirements of the network. Subsequently, multiple batch normalization (BN), convolution (Conv), and activation function layers are incorporated into the complete network structure, as depicted in Fig. 15. Overfitting is mitigated using a dropout layer. Ultimately, the estimated channel parameters, denoted as $\mathbf{h}$, are derived from the flatten and dense layers.

REFERENCES

[1] X. Lu, D. Niyato, H. Jiang, D. I. Kim, Y. Xiao, and Z. Han, "Ambient backscatter assisted wireless powered communications," *IEEE Wireless Commun.*, vol. 25, no. 2, pp. 170–177, Apr. 2018.

[2] P. Wang, Z. Yan, and K. Zeng, "BCAuth: Physical layer enhanced authentication and attack tracing for backscatter communications," *IEEE Trans. Inf. Forensics Security*, vol. 17, pp. 2818–2834, 2022.

[3] C. Girerd, Q. Zhang, A. Gupta, M. Dunna, D. Bharadia, and T. K. Morimoto, "Towards a wireless force sensor based on wave backscattering for medical applications," *IEEE Sensors J.*, vol. 21, no. 7, pp. 8903–8915, Apr. 2021.

[4] M. Merenda, G. Cimino, R. Carotenuto, F. G. D. Corte, and D. Iero, "Edge machine learning techniques applied to RFID for device-free hand gesture recognition," *IEEE J. Radio Freq. Identif.*, vol. 6, pp. 564–572, Jun. 2022.

[5] H. Emenike et al., "Characterizing everyday objects using human touch: Thermal dissipation as a sensing modality," in *Proc. IEEE Int. Conf. Pervasive Comput. Commun. (PerCom)*, 2021, pp. 1–8.

[6] S. Qi et al., "DE-sword: Incentivized verifiable tag path query in RFID-enabled supply chain systems," *IEEE Trans. Dependable Secure Comput.*, vol. 19, no. 3, pp. 2122–2137, May/Jun. 2022.

[7] Z. Chen, M. I. AlHajri, M. Wu, N. T. Ali, and R. M. Shubair, "A novel real-time deep learning approach for indoor localization based on RF environment identification," *IEEE Sens. Lett.*, vol. 4, no. 6, pp. 1–4, Jun. 2020.

[8] W. Jiang et al., "Towards environment independent device free human activity recognition," in *Proc. 24th Annu. Int. Conf. Mobile Comput. Netw.*, 2018, pp. 289–304.

[9] S. Ma, G. Wang, R. Fan, and C. Tellambura, "Blind channel estimation for ambient backscatter communication systems," *IEEE Commun. Lett.*, vol. 22, no. 6, pp. 1296–1299, Jun. 2018.

[10] R. Zhang, B. Shim, and H. Zhao, "Downlink compressive channel estimation with phase noise in massive MIMO systems," *IEEE Trans. Commun.*, vol. 68, no. 9, pp. 5534–5548, Sep. 2020.

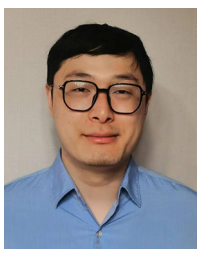
[11] R. Zhang et al., "Integrated sensing and communication with massive MIMO: A unified tensor approach for channel and target parameter estimation," *IEEE Trans. Wireless Commun.*, vol. 23, no. 8, pp. 8571–8587, Aug. 2024.

[12] D. Darsena, G. Gelli, and F. Verde, "Joint channel estimation, interference cancellation, and data detection for ambient backscatter communications," in *Proc. IEEE 19th Int. Workshop Signal Process. Adv. Wireless Commun. (SPAWC)*, 2024, pp. 1–5.

[13] C. He, S. Chen, H. Luan, X. Chen, and Z. J. Wang, "Monostatic MIMO backscatter communications," *IEEE J. Sel. Areas Commun.*, vol. 38, no. 8, pp. 1896–1909, Aug. 2020.

[14] D. Mishra and E. G. Larsson, "Optimal channel estimation for reciprocity-based backscattering with a full-duplex MIMO reader," *IEEE Trans. Signal Process.*, vol. 67, no. 6, pp. 1662–1677, Mar. 2019.

[15] C. Y. Jung, J. M. Kang, and D. I. Kim, "Deep learning based channel estimation for full-duplex backscatter communication systems," in *Proc. Int. Conf. Artif. Intell. Inf. Commun. (ICAIIC)*, 2023, pp. 347–352.
[16] G. Yang, Q. Zhang, and Y.-C. Liang, "Cooperative ambient backscatter communications for green Internet-of-Things," *IEEE Internet Things J.*, vol. 5, no. 2, pp. 1116–1130, Apr. 2018.
[17] H. Guo, Q. Zhang, S. Xiao, and Y.-C. Liang, "Exploiting multiple antennas for cognitive ambient backscatter communication," *IEEE Internet Things J.*, vol. 6, no. 1, pp. 765–775, Feb. 2019.
[18] J. K. Devineni and H. S. Dhillon, "Non-coherent detection and bit error rate for an ambient backscatter link in time-selective fading," *IEEE Trans. Commun.*, vol. 69, no. 1, pp. 602–618, Jan. 2021.
[19] W. Zhao, G. Wang, S. Atapattu, R. He, and Y.-C. Liang, "Channel estimation for ambient backscatter communication systems with massive-antenna reader," *IEEE Trans. Veh. Technol.*, vol. 68, no. 8, pp. 8254–8258, Aug. 2019.
[20] C. Chen, G. Wang, P. D. Diamantoulakis, R. He, G. K. Karagiannidis, and C. Tellambura, "Signal detection and optimal antenna selection for ambient backscatter communications with multi-antenna tags," *IEEE Trans. Commun.*, vol. 68, no. 1, pp. 466–479, Jan. 2020.
[21] J. Choi, "Matched-filter-based backscatter communication for IoT devices over ambient OFDM carrier," *IEEE Internet Things J.*, vol. 6, no. 6, pp. 10229–10239, Dec. 2019.
[22] Y. Azimi, S. Yousefi, H. Kalbkhani, and T. Kunz, "Applications of machine learning in resource management for RAN-slicing in 5G and beyond networks: A survey," *IEEE Access*, vol. 10, pp. 106581–106612, 2022.
[23] C. Qing, L. Wang, L. Dong, and J. Wang, "Enhanced ELM based channel estimation for RIS-assisted OFDM systems with insufficient CP and imperfect hardware," *IEEE Commun. Lett.*, vol. 26, no. 1, pp. 153–157, Jan. 2022.
[24] C. Qing, L. Dong, L. Wang, J. Wang, and C. Huang, "Joint model and data-driven receiver design for data-dependent superimposed training scheme with imperfect hardware," *IEEE Trans. Wireless Commun.*, vol. 21, no. 6, pp. 3779–3791, Jun. 2022.
[25] Q. Zhang, H. Guo, Y.-C. Liang, and X. Yuan, "Constellation learning-based signal detection for ambient backscatter communication systems," *IEEE J. Sel. Areas Commun.*, vol. 37, no. 2, pp. 452–463, Feb. 2019.
[26] Y. Hu, P. Wang, Z. Lin, M. Ding, and Y.-C. Liang, "Machine learning based signal detection for ambient backscatter communications," in *Proc. IEEE Int. Conf. Commun. (ICC)*, 2019, pp. 1–6.
[27] C. Liu, Z. Wei, D. W. K. Ng, J. Yuan, and Y.-C. Liang, "Deep transfer learning for signal detection in ambient backscatter communications," *IEEE Trans. Wireless Commun.*, vol. 20, no. 3, pp. 1624–1638, Mar. 2021.
[28] A. Rahmati and H. Dai, "Reinforcement learning for interference avoidance game in RF-powered backscatter communications," in *Proc. IEEE Int. Conf. Commun. (ICC)*, 2019, pp. 1–6.
[29] M. Yerzhanova and Y. H. Kim, "Channel Estimation via model and learning for monostatic multiantenna backscatter communication," *IEEE Access*, vol. 9, pp. 165341–165350, 2021.
[30] N. K. Kundu and M. R. McKay, "Channel estimation for reconfigurable intelligent surface aided MISO communications: From LMMSE to deep learning solutions," *IEEE Open J. Commun. Soc.*, vol. 2, pp. 471–487, 2021.
[31] C. Liu, X. Liu, D. W. K. Ng, and J. Yuan, "Redefining wireless communication for 6G: Signal processing meets deep learning with deep unfolding," in *Proc. IEEE Int. Conf. Commun. (ICC)*, 2021, pp. 1–7.
[32] L. Cazzella et al., "Deep learning of transferable MIMO channel modes for 6G V2X communications," *IEEE Trans. Antennas Propag.*, vol. 70, no. 6, pp. 4127–4139, Jun. 2022.
[33] J. Yang, X. Gong, B. Ai, and W. Chen, "A priori based deep unfolding method for mmWave channel estimation in MIMO radar aided V2X communications," in *Proc. IEEE Int. Conf. Commun. (ICC)*, 2023, pp. 2946–2951.
[34] 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.
[35] 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, Feb. 2020.
[36] J. R. Hershey, J. L. Roux, and F. Weninger, "Deep unfolding: Model-based inspiration of novel deep architectures," 2014, *arXiv:1409.2574*.
[37] Q. Hu, S. Shi, Y. Cai, and G. Yu, "DDPG-driven deep-unfolding with adaptive depth for channel estimation with sparse Bayesian learning," *IEEE Trans. Signal Process.*, vol. 70, pp. 4665–4680, Sep. 2022.
[38] J. Liao, J. Zhao, F. Gao, and G. Y. Li, "A model-driven deep learning method for massive MIMO detection," *IEEE Commun. Lett.*, vol. 24, no. 8, pp. 1724–1728, Aug. 2020.
[39] N. Samuel, T. Diskin, and A. Wiesel, "Learning to detect," *IEEE Trans. Signal Process.*, vol. 67, no. 10, pp. 2554–2564, May 2019.
[40] A. Jagannath, J. Jagannath, and T. Melodia, "Redefining wireless communication for 6G: Signal processing meets deep learning with deep unfolding," *IEEE Trans. Artif. Intell.*, vol. 2, no. 6, pp. 528–536, Dec. 2021.
[41] L. Pellaco, M. Bengtsson, and J. Jaldén, "Matrix-inverse-free deep unfolding of the weighted MMSE beamforming algorithm," *IEEE Open J. Commun. Soc.*, vol. 3, pp. 65–81, 2022.
[42] Y. Liu, Q. Hu, Y. Cai, G. Yu, and G. Y. Li, "Deep-unfolding beamforming for intelligent reflecting surface assisted full-duplex systems," *IEEE Trans. Wireless Commun.*, vol. 21, no. 7, pp. 4784–4800, Jul. 2022.
[43] T. Yassine and L. Le Magoarou, "mpNet: Variable depth unfolded neural network for massive MIMO channel estimation," *IEEE Trans. Wireless Commun.*, vol. 21, no. 7, pp. 5703–5714, Jul. 2022.
[44] Q. Hu, S. Shi, Y. Cai, and G. Yu, "Sparse Bayesian learning for channel estimation: A DDPG-driven deep-unfolding approach with adaptive depth," in *Proc. IEEE Int. Conf. Commun. (ICC)*, 2022, pp. 1282–1287.
[45] A. Motroni et al., "An RFID tracking system for agricultural safety," in *Proc. IEEE Int. Conf. RFID Technol. Appl. (RFID-TA)*, 2021, pp. 28–31.
[46] G. Yang, C. K. Ho, and Y. L. Guan, "Multi-antenna wireless energy transfer for backscatter communication systems," *IEEE J. Sel. Areas Commun.*, vol. 33, no. 12, pp. 2974–2987, Dec. 2015.
[47] A. J. S. Boaventura and N. B. Carvalho, "The design of a high-performance multisine RFID reader," *IEEE Trans. Microw. Theory Tech.*, vol. 65, no. 9, pp. 3389–3400, Sep. 2017.
[48] F. Zhang, *Matrix Theory: Basic Results and Techniques*. New York, NY, USA: Springer, 2011.
[49] Y. Chen and M. J. Wainwright, "Fast low-rank estimation by projected gradient descent: General statistical and algorithmic guarantees," 2015, *arXiv:1509.03025*.
[50] J. Guo, G. Liu, Q. Wu, and P. Fan, "Parallel attention-based transformer for channel estimation in RIS-aided 6G wireless communications," *IEEE Trans. Veh. Technol.*, early access, Jul. 9, 2024, doi: 10.1109/TVT.2024.3425433.
[51] M. B. Mashhadi and D. Gündüz, "Pruning the pilots: Deep learning-based pilot design and channel estimation for MIMO-OFDM systems," *IEEE Trans. Wireless Commun.*, vol. 20, no. 10, pp. 6315–6328, Oct. 2021.
[52] T. Hu, Y. Huang, Q. Zhu, and Q. Wu, "Channel estimation enhancement with generative adversarial networks," *IEEE Trans. Cogn. Commun. Netw.*, vol. 7, no. 1, pp. 145–156, Mar. 2021.
[53] S. M. Kay, *Fundamentals of Statistical Signal Processing: Estimation Theory*. Hoboken, NJ, USA: Prentice-Hall, 1993.

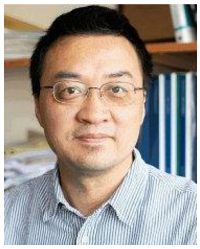


YULIN ZHOU (Member, IEEE) received the M.E. and Ph.D. degrees in electronics engineering from the University of Warwick, Coventry, U.K., in 2016 and 2019, respectively. He is currently working as a Lecturer with the Ningbo Innovation Center, Zhejiang University, China. His research interests include wireless communications, channel estimation, wireless relaying, UAV communications, and energy harvesting.

XIAOTING LI received the B.S. and M.E. degrees in electronics engineering from Zhejiang University, China, in 2021 and 2024, respectively. She is currently working with Zhongxing Telecommunication Equipment Corporation, China. Her research interests include wireless communications, performance analysis, channel estimation, and UAV communications.

XIANMIN ZHANG (Member, IEEE) received the B.S. and Ph.D. degrees in physical electronics and optoelectronics from Zhejiang University, Hangzhou, China, in 1987 and 1992, respectively. He was appointed as an Associate Professor of Information and Electronic Engineering with Zhejiang University in 1994, and a Full Professor in 1999. He was a Research Fellow with the University of Tokyo, Tokyo, Japan, and Hokkaido University, Sapporo, Japan, from November 1996 to September 1997 and from October 1997 to September 1998, respectively. In 2007, he spent two months with the Research Laboratory of Electronics, Massachusetts Institute of Technology, Cambridge, MA, USA, as a Visiting Research Fellow. He was the Dean of the Department of Information Science and Electronic Engineering, Zhejiang University from September 2005 to November 2017, the Dean of the School of Microelectronics, Zhejiang University from May 2015 to September 2018, the Vice Dean of the Polytechnic Institute, Zhejiang University from July 2016 to September 2018, the President of the Zhejiang University Ningbo Institute of Technology, Ningbo, China, from July 2018 to April 2020, and the Dean of Ningbo Campus, Zhejiang University from September 2018 to November 2020. His research interests include microwave photonics and electromagnetic wave theory and applications.

XIAONAN HUI (Member, IEEE) received the B.S. degree in electrical engineering from Northeastern University, China, in 2012, the M.S. degree in electrical engineering from Zhejiang University, China, in 2015, and the Ph.D. degree (Outstanding Thesis Award) in electrical engineering from Cornell University, Ithaca, NY, USA, in 2021. In 2021, he joined the College of Information Science and Electronic Engineering, Zhejiang University as an Assistant Professor. His research interests include wireless sensing, RFID, IoT, and wireless communication systems.



YUNFEI CHEN (Senior Member, IEEE) received the B.E. and M.E. degrees in electronics engineering from Shanghai Jiao Tong University, Shanghai, China, in 1998 and 2001, respectively, and the Ph.D. degree from the University of Alberta in 2006. He is currently working as a Professor with the Department of Engineering, University of Durham, U.K. His research interests include wireless communications, performance analysis, joint radar communications designs, cognitive radios, wireless relaying, and energy harvesting.