IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received 9 January 2024, accepted 21 January 2024, date of publication 1 February 2024, date of current version 8 March 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3360862

RESEARCH ARTICLE

One-Bit Massive MIMO Precoding Using Unsupervised Deep Learning

MOHSEN HOSSEINZADEH[1], HASSAN AGHAEINIA[1], AND MOHAMMAD KAZEMI[2], (Member, IEEE)
[1]Department of Electrical Engineering, Amirkabir University of Technology, Tehran 1591634311, Iran
[2]Department of Electrical and Electronics Engineering, Bilkent University, 06800 Ankara, Turkey

Corresponding author: Hassan Aghaeinia (aghaeini@aut.ac.ir)

ABSTRACT The recently emerged symbol-level precoding (SLP) technique is a promising solution in multi-user wireless communication systems due to its ability to transform harmful multi-user interference (MUI) into useful signals, thereby improving system performance. Conventional symbol-level precoding designs have a significant computational complexity that makes their practical implementation difficult and imposes excessive computational complexity on the system. To deal with this problem, we suggest a new deep learning (DL) based approach that utilizes low-complexity designs of symbol-level precoding. This paper focuses on DL-based one-bit precoding approaches for downlink massive multiple-input multiple-output (MIMO) systems, where one-bit digital-to-analog converters (DACs) are used to reduce cost and power. Unlike previous works, the optimized one-bit precoder for multiuser massive MIMO system (HDL-O1PmMIMO) for a wide range of signal-to-noise-ratio (SNR) has a low computational complexity, making it suitable for real precoding scenarios. In this paper, we first design an unsupervised DL-based precoder (UDL-O1PmMIMO) to address the low SNR scenarios, using which we then design a hybrid DL-based precoder (HDL-O1PmMIMO) to address both low and high SNR scenarios. The method suggested in this article utilizes a novel residual DL network structure, which helps overcome the problem of training very deep networks. Additionally, a novel customized cost function, specifically for one-bit precoding in massive MIMO systems, is introduced to optimize the performance of the system in handling interference. The results of an experiment conducted on a general test set using Python and MATLAB show that the proposed approach outperforms existing methods in three aspects: it has a lower bit error rate, it takes less time to generate the precoded vector, and it is more resistant to imperfect channel estimation.

INDEX TERMS Massive MIMO, one-bit DAC, precoding, unsupervised deep learning.

I. INTRODUCTION

Massive multiple-input multiple-output (MIMO) as a wireless communication technology that uses a large number of antenna elements to improve spectral and energy efficiency is a core technology that is likely to be utilized in all future wireless technologies. This technology is a key enabler of 5G's and 6G's extremely fast data rates and promises to raise their potential to a new level. Along with all the advantages of this technology such as improved coverage at cell edge and improved throughput, there are a series of challenges, cost and power consumption for instance, that many researches have been conducted to overcome and unlock its full potential.

Precoding is a technique used in massive MIMO systems to optimize the transmission of signals from multiple antennas to multiple users. Precoding combines the input signals in a predefined way and delivers them in the right proportion to the multiple antenna elements. one-bit precoding is a technique used in massive MIMO systems to alleviate multi-user interference (MUI). It is a relatively new problem in the massive MIMO downlink.

The adverse effects due to employing a large number of antennas in the base station (BS) in massive MIMO

The associate editor coordinating the review of this manuscript and approving it for publication was Walid Al-Hussaibi.

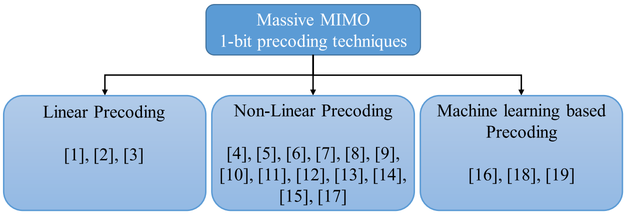


FIGURE 1. One-bit precoding techniques.

systems which brings about an increase in hardware cost and power consumption need to be contemplated to exploit the complementary features of massive MIMO systems. Since data converters especially DACs in the downlink transmission are a significant factor of high power consumption in massive MIMO systems and their resolution in bits causes power consumption to increase exponentially, one-bit precoding is a promising technique for 5G and next-generation wireless cellular networks and improving spectral efficiency and power consumption in massive MIMO systems. This technique can be implemented using simple hardware but due to a reduction in signal quality and high complexity, optimization problems need further research to address its limitations. One-bit precoding involves using one-bit DACs to transmit signals from the BS to multiple users. The one-bit precoding problem in massive MIMO systems is NP-hard and is challenging in frequency-selective and flat-fading channels.

Deep learning (DL) is a powerful technique that has shown great potential in solving optimization problems in wireless communications. Deep learning techniques can be used for a variety of tasks in wireless communications, such as active sensing, channel modeling, distributed source coding, signal detection, channel estimation, compression sensing, encoding and decoding, security and privacy, Internet of Thing (IoT) resource management with massive number of nodes and channel estimation in massive MIMO systems [21], [22], [23], [24], [25], [26]. This paper attempts to address the issue of one-bit precoding in massive MIMO systems using DL methods.

A. RELATED WORK

As demonstrated in Fig.1 there are three main sets of algorithms and designs for one-bit precoding in massive MIMO systems, including linear, nonlinear precoding algorithms and methods based on machine learning concepts. Linear precoding algorithms use linear processing techniques to optimize the transmission of signals from the BS to multiple users in massive MIMO systems [1], [2], [3]. These methods stem from conventional high-resolution precoding methods in massive MIMO systems, in which the quantization process is directly performed on the final linear precoded signal. In these methods, an error floor is encountered due to the final coarse quantization process on the final linear precoded signal. To address this issue, researchers have proposed various techniques such as subset-codebook precoding [4], minimum probability-of-error perturbation precoding [5], and Bussgang analysis of one-bit quantized precoding [6]. These techniques aim to improve the performance of linear precoding methods in conjunction with one-bit quantization by perturbing the linearly precoded signal or optimizing the precoding matrix.

On the other hand, several symbol-level nonlinear one-bit precoding algorithms have been proposed, actively researched, and developed to improve the performance and reliability of massive MIMO systems. These algorithms attempt to optimize the transmission of signals from the BS to multiple users in a massive MIMO system using nonlinear processing techniques. In these methods solving optimization problems can be a challenging task, especially when the problem is of high complexity. Researchers have developed various scenarios to characterize the optimization problems to find the best solution and transmit precoded vectors, among a set of feasible solutions. Some research works in this field that have provided sophisticated results are proposed in [7], [8], [9], [10], [11], [12], [13], [14], and [15], which try to directly map the received symbol vector to a one-bit transmit signal vector. In nonlinear one-bit precoding, the precoding process is formulated as a non-convex optimization problem with high complexity of designing, which can be challenging to solve and can be mentioned as a drawback.

The one-bit precoding with the assumption of constructive interference exploitation is a wireless communication technique that enhances the received symbol by directing it towards the constellation points and making them more concentrated. This enables us to design precoding algorithms that can effectively utilize the interference in the system to improve performance. Several works discuss the topic of one-bit precoding with constructive interference exploitation in the context of massive MIMO systems [13], [19], and [20].

Machine learning (ML) has become an increasingly important tool in communication systems. ML techniques can be used to improve various aspects of communication systems, such as improving the quality of transmitted signals by optimizing the baseband processing algorithms, for instance. The use of DL in one-bit precoding has the potential to improve the performance of communication systems by optimizing the precoding algorithm and reducing its complexity. There has been a significant amount of research done in recent years on the use of DL in one-bit symbol-level precoding (SLP) for communication systems.

The proposed NNO-C2PO one-bit precoding algorithm in [16], improves the performance of the original C2PO algorithm in [17] which iteratively solves a high complexity biconvex optimization problem using projected gradient descent by using DL tools to automatically tune the algorithm parameters while in [17], per-iteration parameters are introduced. One drawback of this approach is that the neural network is just used for specifying the parameters of each iteration of the algorithm in [17] to tackle the performance degradation caused by using equal parameters for all iterations so its computational complexity remains high, similar to the previous method. Additionally, because of perfect channel state information, the NNO-C2PO algorithm is not robust to channel estimation error.

The method in [18] uses an autoencoder to design the received constellation and transmit the precoded vector simultaneously, which can be more efficient and effective than hand-crafted methods. This method tries to assess risks and prevent hazards caused by a gradient of zero in the binary layer. In contrast, the method uses a received constellation, which may not be feasible for all systems and the assumption of sending the same symbol to all users is not a realistic assumption in communication systems. Moreover, end-to-end DL models require a significant amount of data for training, which can be difficult to obtain in some cases.

The author of [19] states that the method employed can help mitigate the effects of quantization and interference simultaneously, leading to improved performance. The author uses a DL-based one-bit precoder with supervision for massive MIMO systems. The method introduced by the author involves simplifying the problem to a convex optimization form, solving it using the CVX library, and training a DL network with the generated dataset to optimize the one-bit precoding process. This approach combines the power of convex optimization and DL to achieve efficient and effective precoding in massive MIMO systems. The precoding design is learned from a rich labeled dataset, which can improve the accuracy and effectiveness of the model and the DL-based precoding design is resistant to imperfect channel state information (CSI), which can improve the robustness and reliability of the system. On the flip side, addressing these problems can be challenging and require a significant amount of time and effort due to the iterative nature of the algorithms involved in generating the labeled dataset, and consequently the exponential increase in complexity as the number of users and antennas grows.

B. OUR WORK

The primary challenge of one-bit SLP design as an NP-hard problem in massive MIMO system is posed by the constraint incurred by the one-bit DACs, which enforce the transmit signals on each antenna to be selected from the set $\{\pm 1 \pm j\}$. In this article, we show that ML, specifically DL, can be a practical solution for one-bit precoding. This paper takes advantage of an innovative DL model that utilizes both supervised and unsupervised DL structures adaptively to reach high-resolution zero-force (ZF) algorithm performance as a theoretical lower limit of bit error rate (BER). The initial step involves the utilization of a unique architecture of an unsupervised residual DL network (UDL-O1PmMIMO). This network is accompanied by a novel customized cost function that is tailored to address the interference exploitation problem. Next, a hybrid network (HDL-O1PmMIMO) is introduced, which operates adaptively with the SNR value. This hybrid network combines the capabilities of unsupervised DL networks proposed in this article and supervised DL networks in [19]. A proposed approach involves a brief exchange of information between the BS and users to assist in reconstructing the main transmit symbol. Table 1 provides a summary of how this article's accomplishments compare with those of other works in the literature. The main achievements of the paper will be discussed in the next section.

C. CONTRIBUTIONS AND OUTLINE

The present article discusses a new proposed method that has achieved satisfactory results compared to previous research in this field. The paper makes the following significant contributions:

- The proposed SLP method aims to approach practical scenarios by simultaneously employing two DL methods, with and without supervision. These methods are used to select the precoding scenario based on the channel conditions. The advantage of this approach is that it simplifies the problem and provides high performance.
- Due to the employment of an unsupervised DL network the proposed solution does not require labeled data, which is time-consuming to manufacture since it requires human experts to identify, categorize, and annotate the data, and it is more closely aligned with actual scenarios.
- The proposed DL network has been optimized for one-bit precoding in massive MIMO systems. This innovative approach utilizes a novel customized cost function and achieves the same error-rate performance with lower complexity compared to the traditional hand-crafted method in [13], which is a near-optimal but NP-hard problem with high and time-consuming computational complexity.
- Employing a novel CI-based customized cost function in our proposed transmit beamforming design enables end-users to detect data bits using simple decision boundaries, which is achieved by pushing the received symbols away from the decision boundary of the constellation in our design.
- A numerical outcome obtained through extensive simulation demonstrates that the proposed method is resilient to uncertainty in CSI.

The following sections will outline the organization of the paper: Section II outlines the system model used in the study, including the assumptions and simulation model. Section III of the article presents the proposed symbol-level one-bit massive MIMO precoding based on the DL method. The simulation results are discussed in section IV, and finally, the paper is wrapped up in section V.

II. SYSTEM MODEL

In this section, we provide an overview of the considered system model. We summarize the key components of the model and describe how it is designed to work. In this paper, as depicted in the illustration of the system model provided in Fig.2, we examine a multiuser massive MIMO system in the downlink direction and the BS with a large

TABLE 1. Pros and cons of some DL-based one-bit precoding in massive MIMO systems.

Research paper	Network	General received modulation	Imperfect CSI robustness	Pros	Cons
[16]	Unsupervised	✓	✗	The process of obtaining the target parameters is more time efficient.	High computational complexity. The proposed method is not resilient to imperfect CSI.
[18]	Unsupervised	✗	✓	A new method is employed to address the problem of zero gradients in binary layers of DL networks. Simultaneously designing the received constellation and transmitting precoded vector using a single DL network.	The constellation diagram of the received signal is not applicable in practical situations. The incorrect and unrealistic assumption of transmitting identical symbols to all users.
[19]	Supervised	✓	✓	It achieves a performance close to the ZF BER bound. Robust to channel estimation errors.	Creating a dataset with labels is a time-consuming process with significant effort and resources, which increases exponentially with the number of users.
UDL O1PmMIMO	Unsupervised	✓	✓	Robust to channel estimation errors. Higher efficiency in low SNR intervals compared to alternative approaches. It achieves comparable performance without the need for a labeled dataset.	Lower performance compared to other methods when the SNR is high.
HDL O1PmMIMO	Hybrid	✓	✓	Robust to channel estimation errors. It achieves comparable performance with much less labeled data than other approaches. It achieves the highest performance and is the closest to the theoretical lower bound of BER compared to other methods across a wide range of SNRs.	The need for a block to estimate SNR whose error affects the overall system performance.

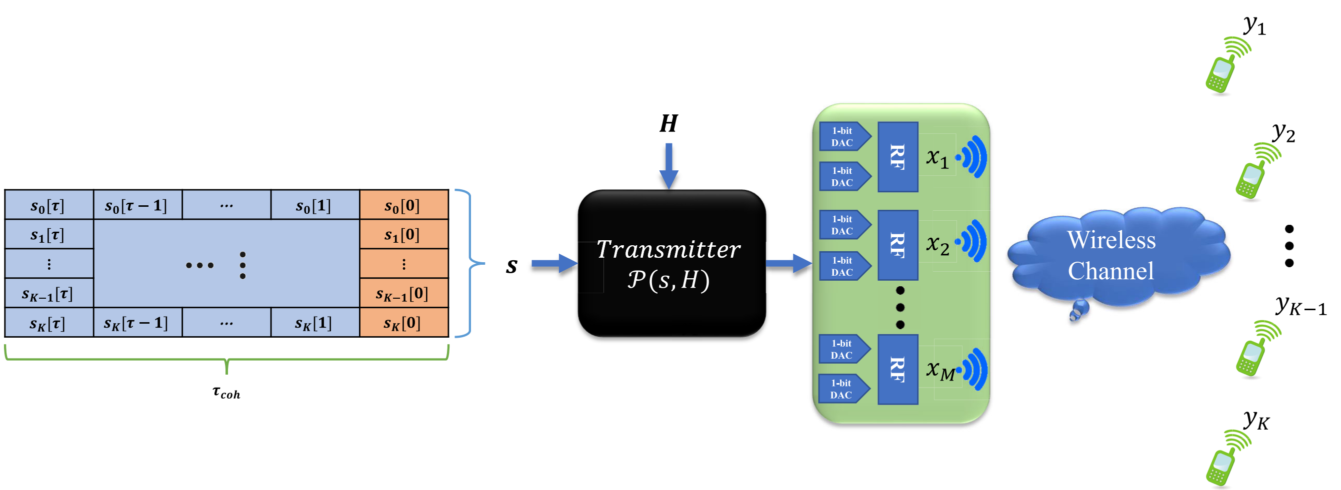


FIGURE 2. One-bit massive MIMO precoding system model.

number of antennas, M, can transmit data to K single-antenna users at the same time using the same frequency band. The assumption is made that the BS possesses complete knowledge of CSI. The data symbol of multiple users can be encoded and stored in a vector $\boldsymbol{s} \in \mathbb{C}^{K\times 1}$ whose elements are derived from an 8PSK constellation with unit-norm. We represent the channel gains between the BS and the k-th user as $\boldsymbol{h}_k \in \mathbb{C}^{M\times 1}$, which corresponds to the k-th column of the instantaneous CSI matrix, $\boldsymbol{H} = [\boldsymbol{h}_1, \ldots, \boldsymbol{h}_k]^H \in \mathbb{C}^{K\times M}$, representing the channel gains between the BS and all users. The vector $\boldsymbol{h}_k$ represents a flat-fading Rayleigh channel, where each element is drawn from a standard complex Gaussian distribution $\mathcal{CN}(0, 1)$. The precoding can be represented as a non-linear mapping between the M-dimensional precoded vector $\boldsymbol{x}$ that is transmitted and the transmitted symbols $\boldsymbol{s}$ and accurately known CSI $\boldsymbol{H}$ as

$$\boldsymbol{x} = \mathcal{P}(\boldsymbol{s}, \boldsymbol{H}), \tag{1}$$

where $\mathcal{P} : \mathbb{C}^{K\times 1} \times \mathbb{C}^{M\times K} \rightarrow \mathbb{C}^M$ is the DL-based precoder [13].

A mathematical expression for the received signal at the k-th user is derived as

$$y_k = \sqrt{P}\boldsymbol{h}_k^H \boldsymbol{x} + z_k, \tag{2}$$

where $\boldsymbol{x} = [x_1, \ldots, x_M]^T$ is a normalized vector such that $\|\boldsymbol{x}\|_2^2 = 1$ with its entries picked from the set $\left\{\pm\frac{1}{\sqrt{2M}} \pm \frac{1}{\sqrt{2M}}j\right\}$, and $z_k \sim \mathcal{CN}\left(0, 2\sigma^2\right)$ is the circularly symmetric additive white Gaussian noise (AWGN).

In the proposed structure outlined in this paper, it is suggested that the transmitter sends a pilot signal during the first time slot of coherence intervals $\boldsymbol{\tau}_{coh}$. This pilot signal consists of pilot symbols that are known in advance by both the transmitter and the receiver. The data transmission should be scheduled in the remaining $\boldsymbol{\tau} - \boldsymbol{\tau}_p$ time slots. In the simulations performed in this article, $\boldsymbol{\tau}_p$ has been considered equal to one.

III. ONE-BIT MASSIVE MIMO PRECODING BASED ON DL NETWORK

In this section, the first step involves designing and formulating a novel one-bit precoding problem based on an unsupervised DL network with a corresponding loss function. Secondly, with the objective of approaching optimal performance while considering practical scenario considerations, low-complexity and high-speed factors, for instance, a hybrid supervised and unsupervised learning approach is constructed. This approach leverages the strengths of both methods to address the symbol-level one-bit precoding problem in a massive MIMO system as a complex problem. Lastly, the approach to training DL networks is outlined.

A. PROBLEM FORMULATION

The one-bit precoder vector formulated the solution of the optimization problem as a non-linear function of the user symbols $\boldsymbol{s}$ and the matrix of channel coefficients $\boldsymbol{H}$ as (1). Luckily, given the fact that universal features are available, deep neural networks excel at addressing this challenge of mapping a nonlinear function due to the inherent nonlinearity of their activation functions. In this paper, we aim to address the complexity and performance challenges associated with the design problem of one-bit precoders in massive MIMO systems. To achieve this, we propose an efficient unsupervised DL approach. The key idea behind our approach is to develop a DL network that can effectively map the channel matrix $\boldsymbol{H}$ and the symbol vector $\boldsymbol{s}$ to the one-bit precoded vector $\boldsymbol{x}$. By leveraging the power of DL, we can learn complex mappings and extract meaningful representations from the input data. By employing unsupervised learning techniques, our DL network can learn from unlabeled data, which makes it suitable for scenarios where labeled training data may be scarce or expensive to obtain. This further enhances the practicality and applicability of our proposed approach. In subsequent, the proposed theory of constructive interference in precoding will be utilized to create an efficient, novel cost function for the designed DL network.

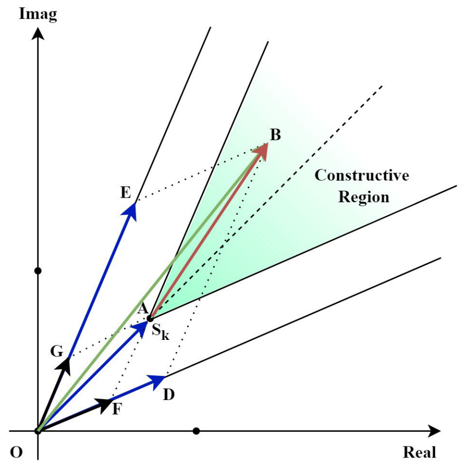


FIGURE 3. CI condition for PSK signaling. $\overrightarrow{OA}$: The ideal transmitted symbol. $\overrightarrow{OG}$ and $\overrightarrow{OF}$ are two decomposed vectors of $\overrightarrow{OA}$ which are parallel to the detection boundaries. $\overrightarrow{OB}$: The received symbol. $\overrightarrow{AB}$: The channel effect. $\overrightarrow{OD}$ and $\overrightarrow{OE}$ are two decomposed vectors of $\overrightarrow{OB}$ which are parallel to the detection boundaries [13].

The main objective is to formulate the problem in a way that can be effectively addressed by the DL network.

As illustrated in Fig.3, $\overrightarrow{OA} = s_k$ is the transmitted symbol and $\overrightarrow{OB} = y_k$ is the received symbol, these two vectors are shown in different colors, blue and green respectively, to distinguish between them. The vector $\overrightarrow{AB}$ in red color represents the channel effects. It is evident that by the law of vector addition and decomposition the relation between the received and transmitted symbol can be expressed mathematically as $\overrightarrow{OB} = \overrightarrow{OA} + \overrightarrow{AB} = \boldsymbol{h}_k^T\boldsymbol{x} = \alpha_k^{\mathcal{A}}s_k^{\mathcal{A}} + \alpha_k^{\mathcal{B}}s_k^{\mathcal{B}}$, where $\alpha_k^{\mathcal{A}}$ and $\alpha_k^{\mathcal{B}}$ are real variables and $s_k^{\mathcal{A}} = \overrightarrow{OG}$, $s_k^{\mathcal{B}} = \overrightarrow{OF}$ are the projection of transmitted symbol on constellation decision boundaries.

It is preferable and the reliability of the transmission can be improved by ensuring that the received symbol is in the decision area of the transmitted symbol as far away as possible from the point where a decision on a received symbol is made. This goal will be achieved by maximizing all the projection of transmitted symbols on constellation decision boundaries. Considering this accurate interpretation, the formulation of the following constructed optimization problem will be well-posed and meaningful.

$$\begin{aligned}
\boldsymbol{x} = \arg\max_{\boldsymbol{x}} \min_{k,u} \alpha_k^U \\
\boldsymbol{h}_k^T\boldsymbol{x} = \alpha_k^{\mathcal{A}}s_k^{\mathcal{A}} + \alpha_k^{\mathcal{B}}s_k^{\mathcal{B}}, \\
&\forall k \in \{1, 2, \ldots, K\}, \\
&\forall u \in \{\mathcal{A}, \mathcal{B}\}, \\
&\forall x_n \in \left\{\pm\frac{1}{\sqrt{2M}} \pm \frac{1}{\sqrt{2M}}j\right\}.
\end{aligned} \tag{3}$$

Due to discrete one-bit constraint which is non-convex, the problem (3) is non-convex and hard to solve. It is possible to convert this problem into a convex one by relaxing the constraints as follows:

$$\begin{aligned}
&\tilde{\boldsymbol{x}} = \arg\max_{\tilde{\boldsymbol{x}}} \min_{k,u} \alpha_k^U \\
&\boldsymbol{h}_k^T \tilde{\boldsymbol{x}} = \alpha_k^{\mathcal{A}} s_k^{\mathcal{A}} + \alpha_k^{\mathcal{B}} s_k^{\mathcal{B}}, \\
&\quad \forall k \in \{1, 2, \ldots, K\}, \\
&\quad \forall u \in \{\mathcal{A}, \mathcal{B}\}, \\
&\quad \forall \tilde{x}_n \in \left\{ |\boldsymbol{Re}(\tilde{x}_n)| \leq \frac{1}{\sqrt{2M}}, |\boldsymbol{Im}(\tilde{x}_n)| \leq \frac{1}{\sqrt{2M}} \right\}.
\end{aligned} \tag{4}$$

The problem (4) can be simplified by following certain algebraic transformations and simplification techniques.

$$\begin{aligned}
\boldsymbol{h}_k^T \boldsymbol{x} &= \alpha_k^{\mathcal{A}} s_k^{\mathcal{A}} + \alpha_k^{\mathcal{B}} s_k^{\mathcal{B}}, \\
\boldsymbol{Re}(\boldsymbol{h}_k^T \boldsymbol{x}) &= \alpha_k^{\mathcal{A}} \boldsymbol{Re}(s_k^{\mathcal{A}}) + \alpha_k^{\mathcal{B}} \boldsymbol{Re}(s_k^{\mathcal{B}}), \\
\boldsymbol{Im}(\boldsymbol{h}_k^T \boldsymbol{x}) &= \alpha_k^{\mathcal{A}} \boldsymbol{Im}(s_k^{\mathcal{A}}) + \alpha_k^{\mathcal{B}} \boldsymbol{Im}(s_k^{\mathcal{B}}),
\end{aligned} \tag{5}$$

so,

$$\begin{aligned}
\alpha_k^{\mathcal{A}} &= \frac{\boldsymbol{Im}\left(\boldsymbol{h}_k^T \boldsymbol{x}\right) \boldsymbol{Re}\left(s_k^{\mathcal{B}}\right) - \boldsymbol{Re}\left(\boldsymbol{h}_k^T \boldsymbol{x}\right) \boldsymbol{Im}\left(s_k^{\mathcal{B}}\right)}{\boldsymbol{Im}\left(s_k^{\mathcal{A}}\right) \boldsymbol{Re}\left(s_k^{\mathcal{B}}\right) - \boldsymbol{Re}\left(s_k^{\mathcal{A}}\right) \boldsymbol{Im}\left(s_k^{\mathcal{B}}\right)} \\
\alpha_k^{\mathcal{B}} &= \frac{\boldsymbol{Im}\left(\boldsymbol{h}_k^T \boldsymbol{x}\right) \boldsymbol{Re}\left(s_k^{\mathcal{A}}\right) - \boldsymbol{Re}\left(\boldsymbol{h}_k^T \boldsymbol{x}\right) \boldsymbol{Im}\left(s_k^{\mathcal{A}}\right)}{\boldsymbol{Im}\left(s_k^{\mathcal{A}}\right) \boldsymbol{Re}\left(s_k^{\mathcal{B}}\right) - \boldsymbol{Re}\left(s_k^{\mathcal{A}}\right) \boldsymbol{Im}\left(s_k^{\mathcal{B}}\right)}
\end{aligned} \tag{6}$$

Therefore, the final simplified version of the optimization problem will be as follows:

$$\begin{aligned}
\tilde{\boldsymbol{x}}_E &= \arg\max_{\tilde{\boldsymbol{x}}_E} \min \left[\alpha_1^{\mathcal{A}}, \alpha_2^{\mathcal{A}}, \ldots, \alpha_K^{\mathcal{A}}, \alpha_1^{\mathcal{B}}, \alpha_2^{\mathcal{B}}, \ldots, \alpha_K^{\mathcal{B}} \right] \\
\left| \tilde{\boldsymbol{x}}_{E_m} \right| &\leq \frac{1}{\sqrt{2M}}, \quad \forall m \in \{1, 2, \ldots, 2M\}, \\
\tilde{\boldsymbol{x}}_E &= \left[\boldsymbol{Re}(\tilde{\boldsymbol{x}})^T, \boldsymbol{Im}(\tilde{\boldsymbol{x}})^T \right]^T,
\end{aligned} \tag{7}$$

The use of the mean squared error (MSE) cost function in DL scenarios may lead to a decrease in performance due to its irrelevance to our main optimization problem. The MSE cost function is used to minimize the difference between predicted and actual values, but it may not be the best choice for our DL scenario. In the mentioned optimization problem, the goal is to place the received symbol in the correct decision interval at the receiver side, which is a much simpler condition. Besides this, the MSE cost function is not recommended for non-convex optimization problems and expects real-valued inputs in range $(-\infty, \infty)$, while the target model output probabilities are in range $(-\frac{1}{\sqrt{2M}}, \frac{1}{\sqrt{2M}})$ and close to the interval boundary through the sigmoid function. Therefore, there is a need for a new cost function that can better gauge the performance of our specific DL models. This simplified version of the main optimization problem allows for the creation of a new cost function that is suitable for addressing the constructive interference problem. The cost function that has been suggested can be expressed in the following manner:

$$\mathcal{L}_{\text{loss}} = -\omega + \frac{1}{\lambda} \sum_{u \in \{\mathcal{A}, \mathcal{B}\}} \sum_{k=1}^{K} \left(\omega - \alpha_k^{\mathcal{U}} \right) \tag{8}$$

where λ is a regularization factor, $\alpha_k^{\mathcal{A}}$ and $\alpha_k^{\mathcal{B}}$ are determined in (6) and

$$\omega = \sum_{u \in \{\mathcal{A}, \mathcal{B}\}} \sum_{k=1}^{K} \alpha_k^u \tag{9}$$

The proposed loss function is a variant of the objective function in the optimization problem (7) and is used to parameterize the model's parameters in proposed DL-based one-bit precoding algorithms. In the optimization problem (7) the goal is to maximize the smallest objective value, subject to the constraints. This objective is achieved by minimizing the proposed cost function, which results from maximizing ω in the first term of (8) and minimizing its second term with a specific regularization factor. Put simply, the further the received symbol in the receiver is from the decision boundary according to Fig.3 is located in the constructive region, the lower the value of this cost function, and conversely, the closer they are to the decision boundary, the higher the value of this cost function.

In this article, it is assumed that the receiver has the capability to adjust the scale of the received signal y_k with the scalar $\beta \in \mathbb{C}$ to estimate the transmitted symbol s_k. The optimal beta value can be computed based on the assumption that the value of the precoded vector $\boldsymbol{x}$ is fixed, using the following approach.

$$\hat{\beta} = \arg\min_{\beta \in \mathbb{C}} \|\mathbf{s} - \beta \mathbf{H} \mathbf{x}_p\|_2^2 \tag{10}$$

Solving this optimization gives us:

$$\hat{\beta} = \|\mathbf{s}\|_2^2 / \left(\mathbf{s}^H \mathbf{H} \mathbf{x}_p \right) \tag{11}$$

where $\mathbf{x}_p$ is in the first time slot of $\boldsymbol{\tau}_{coh}$.

In the introduced system model, a series of symbols are used as pilots. These symbols serve the purpose of beta estimation at the receiver side. In this context, the initial time slot ($t = 1$) is dedicated to transmitting a pilot signal which helps estimate $\hat{\beta}$ and the subsequent ($\tau_{coh} - 1$) time slots are then used for transmitting the data. The value of data that each user has in the initial time slot is the same and is equal to $s_k[0] = \sqrt{E_s}$

B. UNSUPERVISED DL-BASED ONE-BIT PRECODING

The structure of the proposed unsupervised DL network is depicted in Fig.4. The DL network takes the matrix of channel coefficients $\boldsymbol{H}$ and data vector $\boldsymbol{s}$ as inputs and produces the precoded vector of $\tilde{\boldsymbol{x}}_E$ as output. The vector $\tilde{\boldsymbol{x}}_E$ is a concatenation of the real and imaginary parts of vector $\boldsymbol{x}$, $\tilde{\boldsymbol{x}}_E = [x_1, x_2, \ldots, x_{2M}]^T = \left[\boldsymbol{x}_R^T, \boldsymbol{x}_I^T\right]^T$. Since the proposed DL networks do not support complex values, to use complex value data, the channel coefficient matrix which is complex is split into their real and imaginary parts before passing the data to the subsequent layers in the network.

The parameters of the proposed unsupervised DL network, which is utilized for one-bit precoding in massive MIMO systems, are presented in Table 2. The input data is first

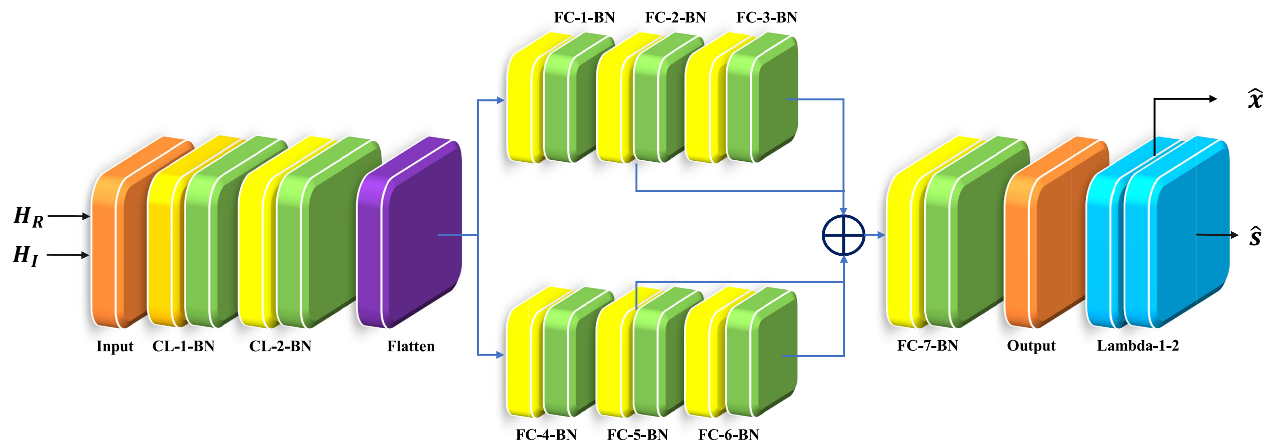


FIGURE 4. The proposed DL network.

TABLE 2. Parameters of the proposed unsupervised DL network.

Type of layer	Parameter
CL-1	(256×K×1)
CL-2	(256 × 1 × 1)
FC-1	2048
FC-2	2048
FC-3	8192
FC-4	2048
FC-5	2048
FC-6	8192
FC-7	2048
Output activation layer	Sigmoid

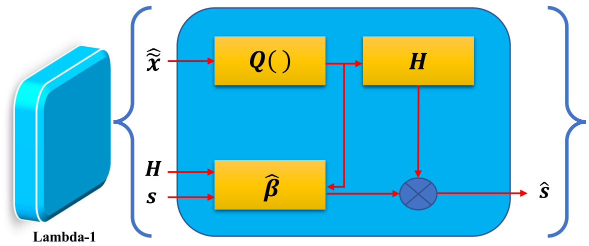


FIGURE 5. Lambda layer structure. $Q()$ is a quantization block.

processed with two convolutional layers, and then the output is flattened into a 1-dimensional array to be converted into a vector. In order to ensure that the size of the output feature map remains the same as the size of the input data a padding technique called SAME is employed. The convolution filter moves across the input data with a stride of K in the horizontal direction and 1 in the vertical direction. By doing so, the network aims to capture the relationships and interactions between the channels in the input data. The second convolutional layer uses a stride value of (1×1) to extract more features from the input data. Then, the network incorporates a parallel structure of fully connected layers, the fully connected layers are designed using the residual method. The purpose of this structure is to obtain additional information. The problem of gradient vanishing can occur in a deep neural network (DNN) when the gradients become very small as they propagate through the layers, making it difficult for the network to learn effectively. Residual network addresses this problem by introducing shortcut connections or additional connections in the network. By using the ReLU activation function after each convolutional layer and all fully connected layers except the last layer, the structure benefits from the non-linear behavior of ReLU, making it easier to train and potentially achieving better performance. The activation function used in the given structure is $\sigma_{\text{LeakyRelu}}(x) = \max(\alpha x, x)$ where $\alpha = 0.1$. Finally, after each activation function, batch normalization is implemented to enhance the training speed and mitigate overfitting. The network produces an output vector with dimension $2M \times 1$, which includes the real and imaginary parts of the precoded one-bit vector. At the end of the process, two lambda layers are added to perform scaling and quantization operations. A lambda layer is created to examine the value of β in the network training process. This layer is illustrated in Fig.5 and is designed based on (11).

This article introduces a new network for one-bit precoding in massive MIMO systems that has two key features that differentiate it from previous DL-based one-bit precoder methods. The first feature is the use of residual blocks to address the issue of vanishing gradients in deep neural networks, which is a general problem in one-bit DL-based precoding in massive MIMO systems. The second feature is the use of a lambda layer to consider the MLE in the training process with the aim of creating a simple receiver.

On the receiver side, a pilot-based estimation of the beta value is considered. The value of the symbol received at the receiver can be determined by sending a one-bit precoded vector, which is the output of the trained network.

$$y_k[t] = \beta^{-1} s_k[t] + e_k[t] + n_k[t] \tag{12}$$

The value of $e_k[t]$ represents the interference between users and the quantization error. It is assumed that the sum of $e_k[t]$ and $n_k[t]$ has a normal distribution and is independent of

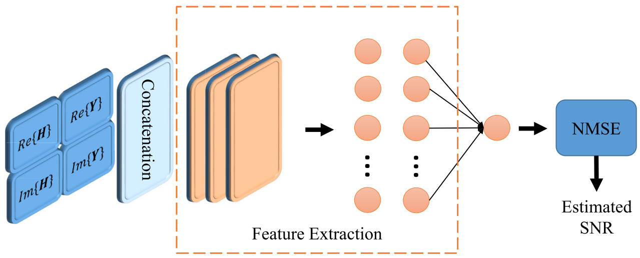


FIGURE 6. DL-based SNR estimation network.

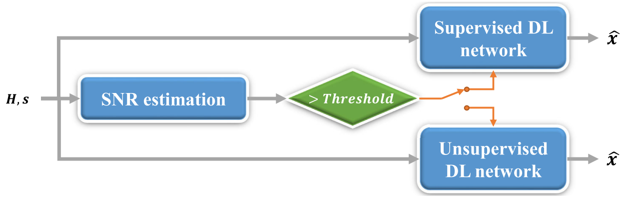


FIGURE 7. Hybrid network structure (O1P-mMIMO).

$s_k[t]$. As a consequence, each user can estimate the ML for β by employing maximum likelihood estimation as follows:

$$\hat{\beta}_k^{\mathrm{MLE}} = \Re\left\{\sqrt{E_s}/y_k[0]\right\}, \quad \forall t \in \{0, 1, \ldots, \tau_{coh} - 1\} \tag{13}$$

The method being described involves assigning time slots to a pilot signal, and the amount of time slots assigned affects the performance and information transfer rate. Specifically, assigning more time slots to the pilot signal results in better performance, but a proportionally lower information transfer rate.

C. DL-BASED ONE-BIT PRECODING ALGORITHM

As shown in Fig.4, nine hidden layer networks are considered. Channel matrix $\boldsymbol{H}$ and symbol vector $\boldsymbol{s}$ are considered as inputs and the precoded vector $\boldsymbol{x}$ as network output. In this section, we will provide a detailed explanation of the mathematical framework underpinning the proposed DL network.

The CSI matrix's real and imaginary parts can be treated as two distinct inputs $\boldsymbol{H_R}$ and $\boldsymbol{H_I}$. After being normalized and passing through the concatenation layer, the intended input will be prepared to enter the DL network. This process results in a new feature map with dimensions $(2 \times M) \times K \times 1$. The output of the i-th convolutional layer can be calculated as:

$$\mathbf{Z}^{[i]} = f\left(\mathbf{W}^{[i]} * \mathbf{O}^{[i-1]} + \mathbf{B}^{[i]}\right) \tag{14}$$

where,

- $\mathbf{W}^{[i]}$ represents the weights of the i-th layer,
- $\mathbf{O}^{[i-1]}$ is the output of the previous layer,
- $\mathbf{B}^{[i]}$ is the bias,
- f is the RELU activation function, and
- $*$ denotes the convolution operation.

In order to stabilize the learning process and reduce internal covariate shift, batch normalization with the following formula is used:

$$\mathbf{O}^{[i]} = \gamma^{[i]}\mathbf{Z}^{[i]} + \beta^{[i]} \tag{15}$$

where $\mathbf{O}^{[i]}$ is the normalized output for the i-th layer and $\gamma^{[i]}$ and $\beta^{[i]}$ are learnable parameters that are determined during the training process. The output of the second convolution layer is processed using a flattening function, which converts the data into a 1-dimensional array. This 1-dimensional array is then fed into two separate fully connected paths with a residual branch.

In a similar manner, the process for 6 fully connected layers can be expressed as follows. The output of each layer can be computed using the following equations:

$$\mathbf{o}^{[i]} = BN\left(f\left(\mathbf{w}^{[i]}\mathbf{a}^{[i-1]} + \mathbf{b}^{[i]}\right)\right) \tag{16}$$

where the term "BN" is used to represent the batch normalization process for brevity and,

- $\mathbf{w}^{[i]}$ represents the weights vector of the i-th layer,
- $\mathbf{o}^{[i-1]}$ is the output of the previous layer,
- $\mathbf{b}^{[i]}$ is the bias vector,
- f is the RELU activation function, and

By using the residual strategy, the input of the seventh fully connected layer is obtained by adding the output of layers 2, 3, 5, and 6 after they have been resized. The output is then passed through the softmax activation layer, which is utilized as the activation function in the output layer. The equation for the softmax activation layer is given by:

$$\sigma(\mathbf{o}_i^{[7]}) = \frac{e^{\mathbf{o}_i^{[7]}}}{\sum_{j=1}^{2M} e^{\mathbf{o}_j^{[7]}}} \tag{17}$$

Finally, the result of this activation function is processed using the specific lambda layer, the structure and equation of which are provided in Fig.5 and (11). By using an adequate number of channel samples to train the neural network, it is feasible to optimize the one-bit precoded output. The algorithm takes the following steps:

Step 1: The parameters of our unsupervised neural network are initialized using the He initialization technique, which is particularly suitable for networks using ReLU activation functions. The He initialization method involves sampling the weights from a normal distribution with a mean of 0 and a standard deviation of $\sqrt{2/n}$, where n represents the number of inputs to the neuron. This technique is designed to mitigate the issue of vanishing or exploding gradients that are often encountered during the training of neural networks. The learning rate decay approach is set up with an initial learning rate of 0.001 and a decay factor of 0.1 applied every 20 iterations.

Step 2: Getting a sample from the CSI dataset, transmit symbol, and preparing it for the network input.

Step 3: Determining the output of convolutional layers based on (14) and (15).

Step 4: Computing the output of the seventh FC layer by considering two separate branches of FC layers and the residual connections by using (16) and (17).

Step 5: Calculating the proposed DL network output using two lambda layers in Fig.5 which is designed based on (11).

Step 6: Determining the value of $\mathcal{L}_{\text{loss}}$ using (6), (8) and (9) as a designed novel CI-based cost function.

Step 7: Going back to **Step 2** if $\left|\mathcal{L}_{\text{loss}}^{i+1} - \mathcal{L}_{\text{loss}}^{i}\right| > \varepsilon$, ε, which represents the maximum allowable error, is assigned a small value. If this condition is met, proceed to **Step 8**.

Step 8: The proposed deep learning network has been effectively trained to generate a one-bit precoding vector corresponding to both pairs of channel coefficients matrix and transmitted symbols.

It is worth mentioning that, the one-bit precoding problem in massive MIMO systems, initially formulated as an unsupervised learning problem, can also be represented as a reinforcement learning network. In this representation, the **step 2** corresponds to observations of the environment can be considered as a **state**, the **step 4** as an **action**, and the $-\mathcal{L}_{\text{loss}}$ as a crucial part of learning process in **step 6** can be interpreted as a **reward**. Creating a new network using reinforcement learning for one-bit precoding in massive MIMO systems is left for future work.

D. HYBRID DL-BASED ONE-BIT PRECODING

In the previous section, a precoding method based on unsupervised DL is proposed. In this section, a hybrid network is introduced to achieve the minimum possible BER, which combines a supervised DL network which is designed in [19] for high SNR and a current proposed unsupervised DL network for low SNR. The diagram in Fig.7 illustrates the structure of the proposed hybrid network as an optimized one-bit precoder for a massive MIMO system which is called O1P-mMIMO. As illustrated, at any given moment, only one of the precoding paths is active and the decision to activate a particular path is made by thresholding the SNR. In this structure, the need for an SNR estimator block before each operation is highlighted. By incorporating this block, The suggested framework ensures that the proposed hybrid structure leads to improved one-bit precoding in massive MIMO systems. To design this network, the first step involves determining a threshold level, in order to select the precoding algorithm. Then the threshold level is used in conjunction with an SNR estimation algorithm. The selection of the threshold level is a crucial step that involves comparing the BER versus SNR of two different methods, namely, the supervised one in [19] and the proposed unsupervised one in this paper. The intersection point is then chosen as the threshold level. The following section discusses the process of estimating SNR in BS using a supervised deep-learning network.

E. DL-BASED SNR ESTIMATION

As mentioned earlier, to determine whether to use a supervised or unsupervised DL network in the one-bit precoder based on the proposed hybrid DL network, an SNR estimation block is required in the BS. The SNR can be estimated using the uplink channel estimation pilots employed in massive MIMO systems.

TABLE 3. Parameters of the proposed DL-based SNR estimation network.

Layer types	Parameter
Input	$(2\times M)\times(PL+K)\times 1$
Conv1 + ReLU	$(5 \times 5 \times 128)$
Max Pooling	$(3 \times 3 \times 128)$
Conv2 + ReLU	$(3 \times 3 \times 256)$
Conv3 + ReLU	$(3 \times 3 \times 256)$
FC4 + ReLU & Dropout	4096
FC5 + ReLU & Dropout	2048

When the users transmit an uplink pilot $\mathbf{P} \in \mathbb{C}^{PL\times K}$, where the PL is the pilot length and K is the number of users, the received signal at the BS with high-resolution analog to digital converters (ADC) can be expressed as follow:

$$\mathbf{Y} = \mathbf{H}\mathbf{P}^T + \mathbf{N}, \tag{18}$$

where $\boldsymbol{H} = [\boldsymbol{h}_1, \ldots, \boldsymbol{h}_k] \in \mathbb{C}^{M\times K}$ is the CSI between the BS with M antennas and K users, $\mathbf{N}$ is a matrix of independent and identically distributed (i.i.d.) elements drawn from $\mathcal{N}_{\mathbb{C}}\left(0, \sigma^2\right)$ and $\mathbf{Y} \in \mathbb{C}^{M\times PL}$ is the matrix of received pilot sequences.

DL is a powerful tool in regression tasks, so we propose the DL network in Fig.6 to estimate the SNR at BS as a regression task. The proposed DL-based SNR estimation network is composed of five layers, the first three of which are convolutional layers, some of them followed by max-pooling layers, and the last two are fully connected layers. The output of all layers except the last one will be passed through the ReLU activation function before being passed to the next layer. The linear activation function is used in the last layer to produce the final output of the neural network. This is because the linear activation function does not limit or put restrictions on the output value, which can be any number. The architecture of the proposed network is presented in Table 3.

In order to train the proposed network, a large number of known pilot symbols are created and transmitted over the channel. The pilot symbols are assumed to have a power of one, and the power of noise is randomly set, resulting in a large number of signal observations with random SNRs that follow a uniform distribution. In this section, the errors in the estimation are measured using NMSE as the loss function.

IV. SIMULATION RESULTS

This section begins by introducing the settings that will be used in the upcoming simulations and then, we will present the simulation results of the one-bit precoder based on unsupervised, and hybrid DL networks in massive MIMO systems. These results aim to demonstrate the efficiency of the proposed method.

A. SETUP

To ensure a fair comparison between the simulation results and the results of the proposed method in [19], the assumed

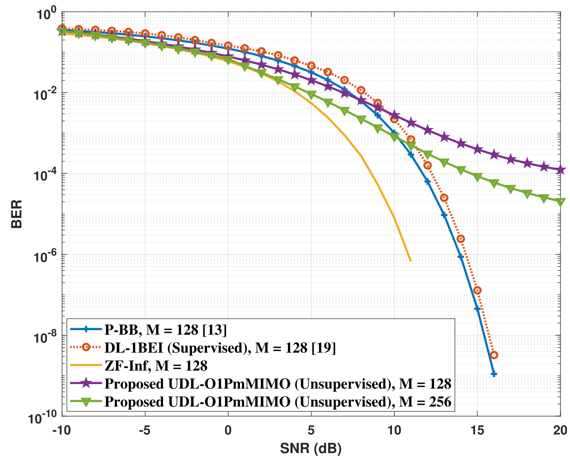


FIGURE 8. BER vs. transmit SNR (Perfect CSI).

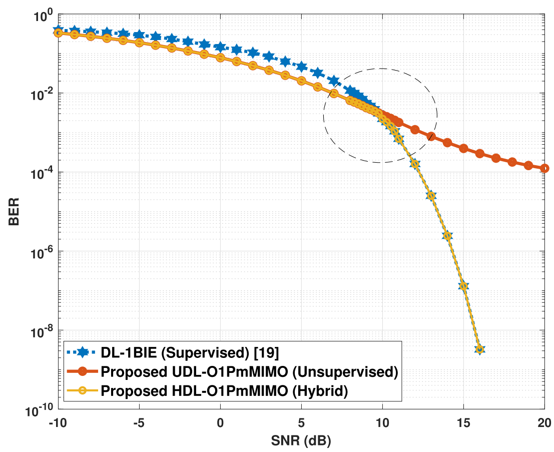


FIGURE 9. BER vs. transmit SNR (Perfect CSI), with perfect SNR estimation.

modulation in this section is 8PSK modulation. It should be noted that the proposed method has the potential to be applied to QPSK modulation as well. Since QPSK modulation, like 8PSK modulation, is a phase modulation, rewriting the equations written in section III-A is mathematically straightforward. However, for brevity and fair comparison in this article, only 8PSK modulation is considered. In the given scenario, the BS is equipped with a massive MIMO system that serves 10 users. The system has either 128 or 256 antennas.

To provide an efficient implementation, the DL network was implemented using the TensorFlow library. The simulations were performed on a personal computer with the following specifications:

- Graphics Processing Unit (GPU): NVIDIA GeForce RTX 3090.
- Central Processing Unit (CPU): Intel Core i9-9980XE processor.
- Memory: 64 GB.

The simulations were conducted in two parts. In the first part, it is assumed that the CSI in the BS is fully known. In the second part, it is presumed that CSI is available to the BS, but with some error. The training data set used in this context contains a channel matrix $\boldsymbol{H}$ with dimensions $K \times M$. Each element of the matrix has real and imaginary parts that follow a standard complex Gaussian distribution $\mathcal{CN}(0, 1)$. Additionally, the data set includes a vector of symbols $\boldsymbol{s}$ with dimensions $K \times 1$. The elements of the vector are randomly selected from the ideal symbols of 8PSK modulation, with a uniform distribution.

To train the proposed network, 10 million pairs of channel matrix and data vectors were used, while 100,000 pairs were used for testing. the network that has been proposed is trained using the ADAM optimizer. The training starts with a learning rate of 0.001, which is gradually decreased by a factor of 0.1 every 20 iterations. The minibatch size used in the training is 1024. The regularization factor is set to 0.2.

B. DISCUSSION

The simulations involved three types of networks: a supervised network proposed in [19], an unsupervised network, and a hybrid network proposed in this context. The comparison of BER in terms of signal-to-noise ratio (SNR) for four precoders in massive MIMO systems is shown in Fig.8. The purpose of this comparison is to evaluate the performance of different precoding techniques in terms of their ability to minimize bit errors in the presence of noise and fading channels. The first precoder, partial branch and bound (P-BB) uses constructive interference and is solved analytically with the CVX toolbox, and the second precoder, DL-1BIE is based on constructive interference and is solved using a deep neural network that is trained by supervision. This approach is proposed in [19] that introduces a novel neural network architecture for one-bit precoding in massive MIMO systems, the third precoder is the ZF precoder, which employs high-precision DACs and is used as the lower limit of the BER in massive MIMO systems and finally, the proposed precoder in this context (UDL-O1PmMIMO) for two different antenna configurations ($M = 128$, $M = 256$). In this simulation, it is observed that the one-bit precoder based on an unsupervised learning network performs better than the precoder based on a DL network with supervision in low SNRs. Additionally, the performance of the proposed one-bit precoder is further improved by increasing the number of antennas in the BS.

The proposed hybrid network (HDL-O1PmMIMO) structure, shown in Fig.7, that combines the capabilities of both supervised and unsupervised networks is based on an SNR estimation structure and setting an optimal threshold level. This hybrid learning architecture can be used to improve the performance of the proposed previous DL models. By leveraging the strengths of both supervised and unsupervised learning, this approach can produce more accurate results and handle a wide range of SNRs. Fig.9 and Fig.10 display the outcomes of the comprehensive simulations conducted. As depicted in Fig.9, the proposed hybrid network has demonstrated acceptable performance in both high and low SNR conditions and requires less labeled training data compared to the supervised network proposed in [19]. The desired outcome for the proposed system is

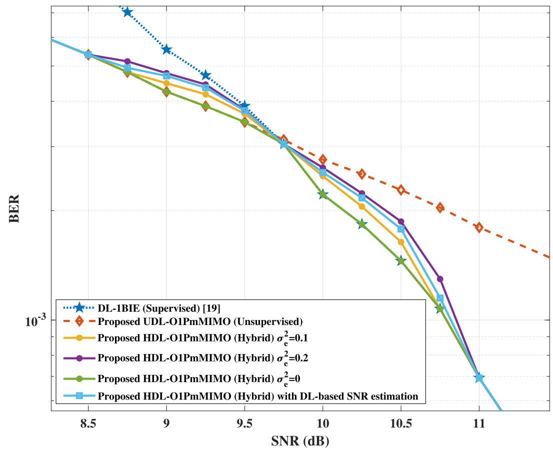


FIGURE 10. BER vs. transmit SNR (Perfect CSI), with imperfect SNR estimation.

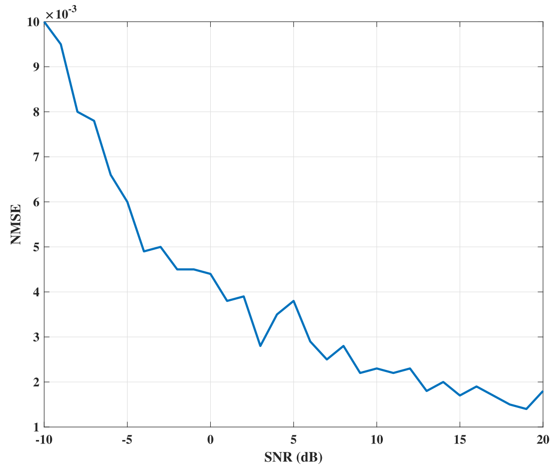


FIGURE 11. NMSEs of the DL-based SNR estimation block.

to achieve the lowest possible BER when compared to the supervised and unsupervised systems that are also proposed. In real-world scenarios, it is more practical to consider errors due to noise when estimating the SNR. In the current simulations, the error is considered as an AWGN term as follows:

$$\boldsymbol{SNR_E} = \boldsymbol{SNR} + \boldsymbol{SNR_e} \tag{19}$$

The SNR estimation error, denoted by $\boldsymbol{SNR_e}$, is a value that is drawn from a zero-mean complex Gaussian distribution $\mathcal{CN}\left(0, \sigma_e^2\right)$. The value of σ_e^2 is taken as $\{0, 0.1, 0.2\}$. The simulation results illustrate how the mentioned error affects the BER of the proposed system, and these results are presented in Fig.9 and Fig.10. Fig.10 is a zoomed-in view of a significant part of Fig.9, which is focused on a specific threshold of SNR ($\boldsymbol{SNR_{th}} = 9.7$ dB) where the impact of the error is more pronounced around it.

To make the hybrid network entirely based on DL, we have proposed a DL-based SNR estimation network to estimate the SNR in BS. We then evaluated the performance of our proposed HDL-O1PmMIMO precoder with the DL-based SNR estimation block, and the results are shown in a light blue solid line in Fig.10. The performance of the DL-based SNR

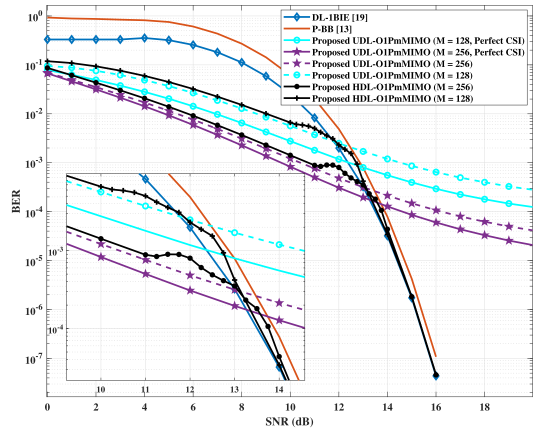


FIGURE 12. BER vs. transmit SNR (Imperfect CSI).

estimation block can be found in Fig.11. The neural network illustrated in Fig.6 is trained using many samples from a set of pre-known channels and received samples in BS labeled with corresponding SNRs. The network's performance is acceptable for SNRs around the chosen threshold level.

The last simulation focused on examining how channel estimation error affects the performance of the proposed system. As the simulation results in Fig.12 show a proposed HDL-O1PmMIMO, DL-based one-bit precoder for a massive MIMO system outperforms the two supervised DL network based one-bit precoders in [19] and the analytical method in [13]. The simulation results show that the proposed one-bit massive MIMO precoder's performance improves as the number of BS antennas increases, and it becomes more resilient to channel estimation errors. Despite having a similar error variance, the one-bit precoder with 256 antennas in the BS has less performance loss than the one with 128 antennas. It is essential to highlight that, the simulation result in Fig.12 is based on the system model depicted in Fig.7 and utilizes the DL-based SNR estimation block as illustrated in Fig.6.

The results of the simulation for addressing performance degradation due to imperfect CSI are illustrated in Fig.13. In this simulation, the inaccurate CSI is represented as an additional component to the ideal CSI as follows:

$$\widehat{\boldsymbol{H}} = \boldsymbol{H} + \boldsymbol{H_e}, \tag{20}$$

where the inaccuracies in estimating the CSI, denoted as $\boldsymbol{H_e}$, are modeled as random errors following a zero-mean complex Gaussian distribution with a variance of $\sigma_e^2 = \{0.05, 0.1, 0.2\}$. In this scenario, the dataset is augmented for each pair of the user symbols $\boldsymbol{s}$ and the matrix of channel coefficients $\boldsymbol{H}$ by incorporating different channel error modes. subsequently, the proposed unsupervised DL network is fine-tuned using an expanded and more varied dataset. This process allows for a more comprehensive analysis of the system's performance under various channel conditions. The one-bit precoding performance in a massive MIMO system based on a new fine-tuned unsupervised DL network is illustrated in green color in Fig.13. The results show that the proposed network's one-bit precoding process

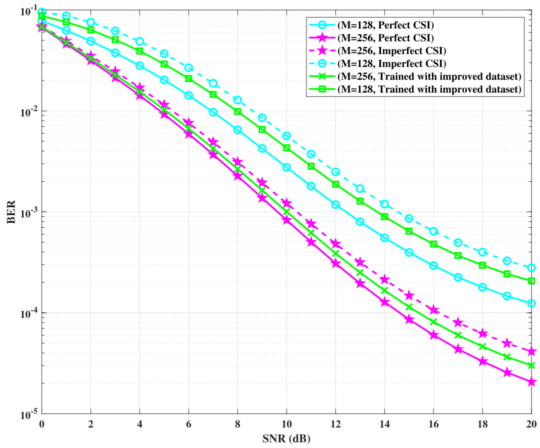


FIGURE 13. BER vs. transmit SNR (Trained with improved dataset).

TABLE 4. Running time in second for M = 128 and K = 10.

Environment	CPU		GPU	
Algorithm	DL-1BIE	P-BB	DL-1BIE	HDL-O1PmMIMO
Matlab	2.4	14.2	0.45	*
Python	1.9	*	0.31	0.22

is enhanced with a richer dataset, and this improvement is more significant with a higher number of antennas.

As the last step of the validation process, The one-bit precoding run time of the proposed method is compared with the processing time of previous methods. The results of this comparison are presented in Table 4. As observed, the processing speed of the proposed method is faster than previous methods under the same simulation conditions and hardware. One of the main reasons is the use of residual blocks in the proposed network, which allows for a larger dropout value compared to the method in [19]. The proposed method not only reduces the number of learnable parameters and makes the final model simpler, but also significantly increases the learning speed of the network in addition to speeding up the one-bit precoding process. A common way to compare the speeds of two DL networks is to calculate the number of addition and multiplication operations. It is possible to use the concepts of FLOPs (Floating Point Operations) as a metric to provide insights into the computational efficiency of DL models. To calculate the FLOPs for the proposed DL network and network in [19] the items listed below must be calculated for all layers of the network:

- For Convolutions: FLOPs = 2 $\times$ Number of Kernels $\times$ Kernel Shape $\times$ Output Shape.
- For Fully Connected Layers: FLOPs = 2 $\times$ Input Size $\times$ Output Size.

Considering the dropout value and the applied pooling strategy, the FLOP value for the current proposed network is approximately 3.6 MFLOPs operations and for the network in [19] is around 5.1 MFLOPs operations. Based on the FLOPs value, which compares the number of multiplications and additions in a DL network, it can be inferred that the proposed unsupervised DL network in Fig.4 is faster in one-bit precoded vector generation than [19], when used in a similar processing system with specifications outlined in subsection IV-A.

V. CONCLUSION

This study has shed new light on the issue of symbol-level one-bit precoding in massive MIMO systems to minimize errors in detecting the sent symbols and maintain its performance under imperfect CSI conditions. In this article, two novel structures of deep learning networks are proposed to address the problem of one-bit precoding in massive MIMO systems as an NP-hard problem with huge complexity. The point of strength of the proposed method over existing methods in literature is utilizing the residual structure in DL networks and offering a novel cost function that aligns with the objective problem. Overall, our proposed hybrid DL approach offers a promising solution to the design problem of one-bit precoders in massive MIMO systems. It combines the benefits of reduced complexity and satisfactory performance making it a valuable contribution to the field.

REFERENCES

[1] J. Guerreiro, R. Dinis, and P. Montezuma, "Use of 1-bit digital-to-analogue converters in massive MIMO systems," *Electron. Lett.*, vol. 52, no. 9, pp. 778–779, 2016.

[2] A. Mezghani, R. Ghiat, and J. A. Nossek, "Transmit processing with low resolution D/A-converters," in *Proc. 16th IEEE Int. Conf. Electron., Circuits Syst.*, Yasmine Hammamet, Tunisia, Dec. 2009, pp. 1–11.

[3] O. B. Usman, H. Jedda, A. Mezghani, and J. A. Nossek, "MMSE precoder for massive MIMO using 1-bit quantization," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Shanghai, China, Mar. 2016, pp. 3381–3385.

[4] O. Tirkkonen and C. Studer, "Subset-codebook precoding for 1-bit massive multiuser MIMO," in *Proc. 51st Annu. Conf. Inf. Sci. Syst. (CISS)*, Mar. 2017, pp. 1–5.

[5] A. Swindlehurst, A. Saxena, A. Mezghani, and I. Fijalkow, "Minimum probability-of-error perturbation precoding for the one-bit massive MIMO downlink," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Mar. 2017, pp. 6483–6487.

[6] A. K. Saxena, I. Fijalkow, and A. L. Swindlehurst, "Analysis of one-bit quantized precoding for the multiuser massive MIMO downlink," *IEEE Trans. Signal Process.*, vol. 65, no. 17, pp. 4624–4634, Sep. 2017.

[7] S. Jacobsson, G. Durisi, M. Coldrey, T. Goldstein, and C. Studer, "Quantized precoding for massive MU-MIMO," *IEEE Trans. Commun.*, vol. 65, no. 11, pp. 4670–4684, Nov. 2017.

[8] M. Shao, Q. Li, W.-K. Ma, and A. M. So, "A framework for one-bit and constant-envelope precoding over multiuser massive MISO channels," *IEEE Trans. Signal Process.*, vol. 67, no. 20, pp. 5309–5324, Oct. 2019.

[9] F. Sohrabi, Y.-F. Liu, and W. Yu, "One-bit precoding and constellation range design for massive MIMO with QAM signaling," *IEEE J. Sel. Topics Signal Process.*, vol. 12, no. 3, pp. 557–570, Jun. 2018.

[10] A. Li, C. Masouros, F. Liu, and A. L. Swindlehurst, "Massive MIMO 1-bit DAC transmission: A low-complexity symbol scaling approach," *IEEE Trans. Wireless Commun.*, vol. 17, no. 11, pp. 7559–7575, Nov. 2018.

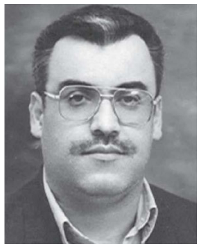
[11] L. T. N. Landau and R. C. de Lamare, "Branch-and-bound precoding for multiuser MIMO systems with 1-bit quantization," *IEEE Wireless Commun. Lett.*, vol. 6, no. 6, pp. 770–773, Dec. 2017.

[12] S. Jacobsson, W. Xu, G. Durisi, and C. Studer, "Mse-optimal 1-bit precoding for multiuser MIMO via branch and bound," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Apr. 2018, pp. 3589–3593.

[13] A. Li, F. Liu, C. Masouros, Y. Li, and B. Vucetic, "Interference exploitation 1-bit massive MIMO precoding: A partial branch-and-bound solution with near-optimal performance," *IEEE Trans. Wireless Commun.*, vol. 19, no. 5, pp. 3474–3489, May 2020.

[14] X. Yuan and J. Zheng, "Nonlinear one-bit precoding for massive MIMO downlink systems with L_0-norm constraint," *IEEE Wireless Commun. Lett.*, vol. 9, no. 9, pp. 1514–1518, Sep. 2020.
[15] L. Chu, F. Wen, L. Li, and R. Qiu, "Efficient nonlinear precoding for massive MIMO downlink systems with 1-Bit DACs," *IEEE Trans. Wireless Commun.*, vol. 18, no. 9, pp. 4213–4224, Sep. 2019.
[16] A. Balatsoukas-Stimming, O. Castañeda, S. Jacobsson, G. Durisi, and C. Studer, "Neural-network optimized 1-bit precoding for massive MU-MIMO," in *Proc. IEEE 20th Int. Workshop Signal Process. Adv. Wireless Commun. (SPAWC)*, Jul. 2019, pp. 1–5.
[17] O. Castañeda, S. Jacobsson, G. Durisi, M. Coldrey, T. Goldstein, and C. Studer, "1-bit massive MU-MIMO precoding in VLSI," *IEEE J. Emerg. Sel. Topics Circuits Syst.*, vol. 7, no. 4, pp. 508–522, Dec. 2017.
[18] F. Sohrabi and W. Yu, "One-bit precoding constellation design via autoencoder-based deep learning," in *Proc. 53rd Asilomar Conf. Signals, Syst., Comput.*, Nov. 2019, pp. 754–758.
[19] M. Hossienzadeh, H. Aghaeinia, and M. Kazemi, "Deep learning based interference exploitation in 1-bit massive MIMO precoding," *IEEE Access*, vol. 11, pp. 17096–17103, 2023.
[20] A. Li, C. Masouros, A. L. Swindlehurst, and W. Yu, "1-bit massive MIMO transmission: Embracing interference with symbol-level precoding," *IEEE Commun. Mag.*, vol. 59, no. 5, pp. 121–127, May 2021.
[21] Y. Zhang, M. Alrabeiah, and A. Alkhateeb, "Deep learning for massive MIMO with 1-bit ADCs: When more antennas need fewer pilots," *IEEE Wireless Commun. Lett.*, vol. 9, no. 8, pp. 1273–1277, Aug. 2020.
[22] A. Salh, R. Ngah, G. A. Hussain, L. Audah, M. Alhartomi, Q. Abdullah, R. Alsulami, S. Alzahrani, and A. Alzahmi, "Intelligent resource management using multiagent double deep Q-networks to guarantee strict reliability and low latency in IoT network," *IEEE Open J. Commun. Soc.*, vol. 3, pp. 2245–2257, 2022.
[23] A. Salh, L. Audah, K. S. Kim, S. H. Alsamhi, M. A. Alhartomi, Q. Abdullah, F. A. Almalki, and H. Algethami, "Refiner GAN algorithmically enabled deep-RL for guaranteed traffic packets in real-time URLLC B5G communication systems," *IEEE Access*, vol. 10, pp. 50662–50676, 2022.
[24] L. Dai, R. Jiao, F. Adachi, H. V. Poor, and L. Hanzo, "Deep learning for wireless communications: An emerging interdisciplinary paradigm," *IEEE Wireless Commun.*, vol. 27, no. 4, pp. 133–139, Aug. 2020.
[25] H. Wu, X. Li, and Y. Deng, "Deep learning-driven wireless communication for edge-cloud computing: Opportunities and challenges," *J. Cloud Comput.*, vol. 9, no. 1, pp. 1–14, Dec. 2020.
[26] W. Yu, F. Sohrabi, and T. Jiang, "Role of deep learning in wireless communications," *IEEE BITS Inf. Theory Mag.*, vol. 2, no. 2, pp. 56–72, Nov. 2022.

MOHSEN HOSSEINZADEH received the B.S. degree from Urmia University and the M.S. degree from the Electrical Engineering Department, Amirkabir University of Technology, Tehran, Iran, where he is currently pursuing the Ph.D. degree with the Electrical Engineering Department. His research interests include wireless communications, signal processing, and machine learning, with a particular focus on m-MIMO systems.



HASSAN AGHAEINIA received the B.Sc. and first M.Sc. degrees from the Amirkabir University of Technology (Tehran Polytechnic), Tehran, Iran, in 1987 and 1989, respectively, and the second M.Sc. and Ph.D. degrees from Valenciennes University (UVHC), Valenciennes, France, in 1992 and 1996, respectively, all in electronic engineering. Since 1996, he has been a Faculty Member with the Amirkabir University of Technology, where he is currently an Associate Professor with the Communication Engineering Group. His research interests include advanced communication systems and digital image processing.

MOHAMMAD KAZEMI (Member, IEEE) received the B.S. and M.S. degrees from the K. N. Toosi University of Technology, Tehran, Iran, in 2007 and 2010, respectively, and the Ph.D. degree from the Amirkabir University of Technology, Tehran, in 2017, all in electrical engineering (EE). He was a Visiting Student with Bilkent University, Turkey, in 2015. He was a Researcher in MMWCL with the Amirkabir University of Technology, from 2017 to 2019. He is currently a Postdoctoral Fellow with the EE Department, Bilkent University. His research interests include wireless communications and signal processing, with a particular focus on massive MIMO and massive random access systems. He is the Editorial Assistant to the Editor-in-Chief of IEEE Transactions on Communications.