IEEE *Access*
Multidisciplinary : Rapid Review : Open Access Journal

Deep Learning based Autoencoder for m-user Wireless Interference Channel Physical Layer Design

DEHAO. WU[1], MAZIAR. NEKOVEE[1,2], AND YUE. WANG .[3], (Senior Member, IEEE)
[1]Centre for Advanced Communications, Mobile Technology and IoT, University of Sussex, Brighton, United Kingdom (e-mail: dehao.wu@sussex.ac.uk)
[2]Quantrom Technologies Ltd, United Kingdom
[3]Network Standards and Research, Communications Research Group, Samsung R&D Institute, United Kingdom

Corresponding author: Dehao Wu (e-mail: dehao.wu@sussex.ac.uk).

ABSTRACT Deep learning (DL) based autoencoder (AE) has been proposed recently as a promising, and potentially disruptive approach to design the physical layer of beyond-5G communication systems. Compared to a traditional communication system with a multiple-block structure, the DL based AE approach provides a new paradigm to physical layer design with a pure data-driven and end-to-end learning based solution. In this paper, we address the dynamic interference in a multi-user Gaussian interference channel. We show that standard constellation are not optimal for this context, in particular, for a high interference condition. We propose a novel adaptive DL based AE to overcome this problem. With our approach, dynamic interference can be learned and predicted, which updates the learning processing for the decoder. Compared to other machine learning approaches, our method does not rely on a fixed training function, but is adaptive and applicable to practical systems. In comparison with the conventional system using n-psk or n-QAM modulation schemes with zero force (ZF) and minimum mean square error (MMSE) equalizer, the proposed adaptive deep learning (ADL) based AE demonstrates a significant achievable BER in the presence of interference, especially in strong and very strong interference scenarios. The proposed approach has laid the foundation of enabling adaptable constellation for 5G and beyond communication systems, where dynamic and heterogeneous network conditions are envisaged.

INDEX TERMS Deep learning, Autoencoder, 5G physical layer, and Interference channel.

I. INTRODUCTION

ARTIFICIAL intelligence (AI) is becoming increasingly present in all aspects of our lives, and it has the capability to manage more complex, data-intensive tasks. In the area of communications, there is an increasing awareness that communication networks and services are becoming more intelligent with the novel advancements and unprecedented levels of computational capacity. AI is soon to move and work among the networks, processing locally or in the cloud. Machine learning (ML) and deep learning (DL), which are the most prominent AI approaches today, are being extensively employed for designing and managing complex communication systems and services. It has been demonstrated to significantly improvement of the system performance, as well as the quality of the services [1] [2]. Therefore, design, development and use of AI systems has attracted great attention, not only in industry, but also in the research

community [3]. Many studies of the AI technologies have been carried out in communication systems in recent years, including unknown channel estimation and detection through DL [4], super-resolution channel estimation for a massive multiple-input multiple-output (MIMO) system, novel DL based algorithm for decoding [6], joint channel encoding and source encoding [8] and DL for joint channel estimation and detection in Orthogonal Frequency Division Multiplexing (OFDM) systems [7].

In a conventional mathematical derived communication system model, multiple functional blocks structure is used to build a link. Each block is optimized individually to improve performance. However, the DL based autoencoder design provide a holistic solution which is a pure data-driven and end-to-end learning-based optimization approach. Many studies have been carried out recently. For instance, end-to-end learning-based autoencoder (AE) is studied in [9]. By

interpreting the system as an AE, the system is reconstructed, and the transmitter and receiver components are optimized jointly in a single process. In [10], authors proposed a novel learning algorithm that iterates between supervised training of the receiver and reinforcement learning (RL)-based training of the transmitter and achieved an end-to-end system without a channel model. In [11], authors designed a system that learns to transmit real numbers over an unknown channel without a preexisting feedback link.

Some DL-based experimental work has been carried out and implemented, as shown in [17] and [18]. Those studies above have significant insights into the performance enhancement, and it shows that the DL based AE is the most promising approach for interference-free channels. Studies in [19] provided an overview of physical layer DL and the state of the art for 5G and beyond system. It shows the potential of DL approaches to address problems in the physical layer, such as the dynamic interference channel [12], modulation recognition [20], radio fingerprinting [21] and medium access control [22], especially when casting in the context of real-time hardware-based implementation. On the other hand, 5G network requires a higher bandwidth in order to achieve greater data rate. It will be largely characterized by small cell deployments. The implementation of small size networks delivers various advantages such as high data rate and low signal delay. However, it also suffers from various interference [12]. For instance, ultra-dense small-cell networks (USNs) have been established as one of the vital networking architectures in the 5G [14]. However, intensive deployment of cells results in a complex interference problem. Other interference challenges include the high mobility induced multi-user interference [15] and problems in multi-user MIMO interference channel [16] are studied too.

In this work, our DL based AE demonstrates excellent robustness to various interference levels, where the conventional design either does not consider the interference channel or uses simplified models of interference channel which usually yields sub-optimal performance. We believe that the proposed AE approach is one of the most promising approaches to address the problem of dynamic interference channel in the physical layer, especially for the 5G-and-beyond networks. 5G small cells is an important example of such challenges, where the proposed approach could be useful.

In contrast, studies in [23] indicated that AE can be vulnerable to adversarial and jamming attacks comparing to the conventional schemes. The work in [24] points out that such disadvantages can be mitigated through adversarial training. The studies in [25] [26] show that it is also not clear that what is the behaviour of AE under a multi-user interference channel. And how to enhance the performance of a multi-user system which is often impaired. The study in [9] proposed a solution for the interference of a two-user link when AE is applied. However, only two users are considered, and offline training is used. Furthermore, there is no adaptive training for different levels of interference under

multiple user's scenarios. Some other studies on AE are also presented, such as the MIMO channel learning [28], channel estimation in an OFDM system [29], and learning to optimize for interference management [30]. However, those studies are based on offline learning, and, theretofore, are unable to cope with scenarios where interference is dynamic and may vary in real-time.

In this paper, we address the dynamic interference in a multi-user Gaussian interference channel. We show that standard constellation are not optimal for this context, in particular, for a strong or very strong interference condition. We propose a novel adaptive DL based AE to overcome this problem. We study the constraint of a conventional offline trained system and demonstrate the improvement of our proposed approach. Additionally, an in-depth analysis of the symbol constellation is studied, and we also apply the ZF and MMSE equalizers for the interference channel and analyze the performance when it compares with the proposed ADL based AE approach. In our ADL based AE algorithm, interference strength is predicted through a DL learning process. With the real-time online learning of the knowledge of the interference level, we show that the proposed AE works more robustly in an interference channel for all interference levels. In particular, the improvement is more notable for the strong and very strong interference scenarios. Preliminary discussions and results were presented in our earlier work [31].

The main contributions of this paper are summarized below:

1) For a conventional communication channel, it is a challenge to overcome the dynamic interference caused by multiple users with a predetermined mathematical model. In this work, we propose an ADL algorithm for a m-user interference Gaussian channel, to enhance the robustness of the link by estimating the uncertain interference via learning. Comparing with studies using other ML based AE methods, our system does not rely on a fixed training model. Instead, we utilize the pilot signals, and estimate the dynamic interference strengths via an adaptive training loop at the receiver side. We optimize the ML training processing by turning the estimation function. By substituting the estimated interference 'status' back into the ML training, we update the decoder and obtain a more robust communication link.

2) In the proposed model, the interferences are classified as different levels from weak to very strong, based on a coupling parameter α. With our algorithm, α can be estimated through a training process and a predetermined 'reward' function. We further characterize the tolerance and robustness of the system, according to different interference levels. We demonstrate the constraints and propose an algorithm. We compare results against the system using a conventional DL-assist AE. We demonstrate that our proposed ADL-assist algorithm has a significant capability to overcome the

effect induced by different levels of interference-to-noise ratio (INR). In particular, the enhancement is more notable for the strong and very strong interference scenarios.

3) We study the learned constellation, and analyze its difference with comparing to a conventional system using n-psk or n-QAM modulation schemes, as well as with conventional ZF and MMSE equalizer. We reveal the possible fact that how the constellation effected by the interference. We study the compressed techniques such as one hot vector and inverse one hot vector based encoder and decoder, and discuss the promising methodologies to minimize the computational complexity when vectors get larger.

The rest of the paper is organized as follows. In Section II we introduce the system model and its underlying mathematical description. The algorithm and specifications of DNNs learning architecture are also introduced. We evaluate the performance of the proposed algorithm in Section III, comparing and discussing the results with the system using a conventional AE method. We conclude the paper in Section VI.

II. SYSTEM MODEL
A. SYSTEM OVERVIEW AND DNN BASICS

The system block diagram is shown in Fig.1. An ADL algorithm based AE is proposed for a wireless communication interference channel with m-user. The system is composed of m pair transceivers. Each sender encodes data by using a predetermined codebook, and each receiver decodes data by treating the interference as noise. It has three main blocks: transmitter, channel, and receiver. Compared to a conventional communication system with a number of blocks, this proposed diagram recast the block diagram as an end-to-end optimization task and represent the system as a simplified AE system. For the transmitter side, the transmitted messages s is reconstructed, and $s_i \in \mathbb{M} = \{1, 2, \ldots, M\}$, where $M = 2^k$ is the dimension of M with k being the number of bits per message. The format of messages s is one hot encoded vector $1_s \in \mathbb{R}^M$. A one hot encoding is a representation of categorical variables as binary vectors. One hot encoding is used in ML as a method to quantify categorical data. This method produces a vector with length equal to the number of categories in the data set. More details of one hot encoding can be found in [39]. The message is passed to the transmitter. Transmitter output is a $2n$-dimensianal vector which corresponds to n complex symbols transmitted in n channel uses by considering one half as real part and the other as the imaginary part. The channel is represented by an additive noise layer with a fixed variance $\beta = (2RE_b/N_0)^{-1}$, where E_b/N_0 denotes the energy per bit (E_b) to noise power spectral density (N_0) ratio. R is the communication rate. The receiver is implemented as a feedforward NN with a single or multiple dense layers followed by an output layer with softmax activation whose output $p \in (0, 1)^M$ is a probability vector over all possible messages. The decoded message $\hat{s}_i$

corresponds to the element index of p which has the highest probability. The AE can be trained using stochastic gradient descent or any other suitable optimization approaches on the set of all possible messages using the categorical cross-entropy loss function. The basics of DNN is introduced in [35], where also a method for stochastic optimization is proposed. Similar to [35], the NN layers considered in this work transform an input data l_{in} into an output l_{out} as follows:

$$l_{\text{out}} = f(\boldsymbol{w}l_{\text{in}} + \boldsymbol{b}), \tag{1}$$

where $\mathbf{w}$ and $\mathbf{b}$ are weights and trainable parameters and $f(.)$ is the activation function, which includes both linear and non-linear ones [37]. These functions are listed in Table 1. With the appropriate choice of parameters, multi-layer neural networks can in principle approximate any smooth function, with more hidden units allowing one to achieve better approximations. The weights of the whole layers are optimized jointly.

For a fully connected neural network with J layers, an input vector $\boldsymbol{l}_0$ maps to an output vector $\boldsymbol{l}_J$ via J iterations:

$$\boldsymbol{l}_j = f_j(\boldsymbol{l}_{j-1}; \nu_j), j = 1, ..., J \tag{2}$$

where $f_j(\boldsymbol{l}_{j-1}; \nu_j) : \mathbb{R}^{M_{j-1}} \rightarrow \mathbb{R}^{M_j}$ is the mapping on the jth layer. The mapping relies on the output vector $\boldsymbol{l}_{j-1}$ from the earlier layer and a series of parameters ν_j. $\boldsymbol{\nu} = \nu_1, ..., \nu_J$ presents the set of parameters in each layer of all J layers. In this work, the transmitter x applies a transformation $f : M \rightarrow \mathbb{R}^{2n}$ to the message s_i to generate the transmitted signal $x_i = f(s_i) \in \mathbb{R}^{2n}$. Note that the output of the transmitter is an n-dimensional complex vector which is transformed to a $2n$ real vector. Following the similar definition in [10], the transmitter is constrained by either an energy constraint: $x_i^2 \leq n$ or an average power constraint: $\mathbb{E}[|x_i^2|] \leq 0.5 \ \forall \ i$. Signal is sent to the receiver using the channel n times. The communication rate of this system is $R_c = k/n$. In this work, a model of an additive white Gaussian noise (AWGN) channel is used. The channel causes distortions to the transmitted symbols and at the receiver upon reception of signal $y \in \mathbb{C}^n$. One commonly used loss function is the squared error, which is given by:

$$L(f_\theta(x), y) = ||f_{\theta(x)} - y||^2. \tag{3}$$

If f is unrestricted, minimizing the expected value of the loss function over the distribution $P(x, y)$ yields:

$$f(x) = \mathbb{E}[y \mid x = \hat{x}]. \tag{4}$$

This is the conditional expectation of y, which the neural network is trying to learn. However, when y is a discrete label, other loss function such as the Bernoulli negative log-likelihood have been proposed to be more appropriate than the squared error [37]. In this work, the receiver produces the estimate signal $\hat{s}$, where $\hat{s}$ is a realization of the original
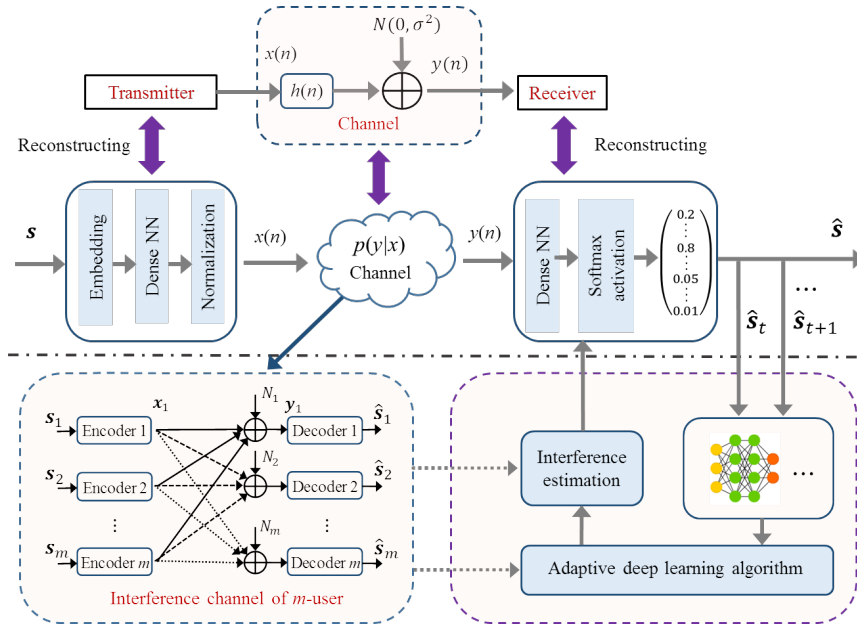
FIGURE1: System block diagram of an adaptive deep learning based AE for a wireless communication interference channel with m-user.

transmitted signal s. The network is trained to optimize the reconstruction error, which is given by:

$$L(s, \hat{s}) = -\log p(s|\hat{s}) \qquad (5)$$

The reconstruction error here is known as the cross entropy loss, which is given by [37]:

$$L(s, \hat{s}) = -\sum_{\mu}(s(\mu)\log\hat{s}(\mu) + (1 - s(\mu))\log(1 - \hat{s}(\mu))) \qquad (6)$$

where $\hat{s}(\mu) = P(s(\mu) = 1|\hat{s})$. $s(\mu)$ stands for bit μ of s and $\hat{s}(\mu)$ stands for bit μ of $\hat{s}$. The training of the network is performed by solving the following optimization problem:

$$\arg\min_{\Phi} \mathbb{E}_{s,N,\theta}[L(s, \hat{s})] \qquad (7)$$

where Φ denotes the set of trainable parameters. N and θ are generated noise and phase by the channel layer each time it is used. To achieve this, the sigmoid as non-linearity for the output layer is used. The cross entropy criterion allows gradients to pass through the output non-linearity even when the neural network produces a wrong answer, which outperforms the squared error approach coupled with a sigmoid or softmax non-linearity.

As shown in Fig.1, at the receiver side, y represents the received signal after propagating through an interference channel, which includes the original transmitted signal, the channel response, AWGN noise as well as the interference from other sources. Here, the received n-dimensional signal y noised by a channel represented as a conditional probability density function $p(y|x)$, and the DNNs receiver subsequently learns it with multiple dense layers. The last layer of the receiver is a Softmax activation layer that outputs an M-dimensional probability vector $\mathbf{p}$. The receiver applies the transformation $R^{2n} \to M$ to decode the signal, creating a signal $\hat{s}_i$ for signal recovery. In the receiver block, we propose an adaptive training loop for enhancing the decoder process. By utilizing the pilot signals, the interference strength can be estimated via a training loop. By substituting the estimated interference 'status' back into the learning, we update the training function for the decoder and obtain a more robust communication link.

To enable the comparability of the results implemented in different scenarios, we set (n, k)=(1,1), (1,2), (1,3) and (1,4) respectively throughout this work, to compare with other competitive conventional modulation schemes n-psk and n-QAM. The equivalent modulations for comparing are following the setting of $2^{k/n}$-psk/QAM for different parameters (n, k). We train the AE in an end-to-end manner using the Adam optimizer, on the set of all possible messages $s_i \in \mathbf{M}$, using the cross-entropy loss function.

B. MODEL OF MULTI-USER INTERFERENCE CHANNEL

A m-user Gaussian AWGN interference channel is shown in Fig.1 within the dashed-line rectangle block. The interference channel has m transmitter-receiver pairs that simultaneously communicate in a block of size m. Each transmitter communicates to its own receiver a message $s \in M = \{1, 2, \ldots, M\}$ by choosing a code word $C_{i,m}$. Let x_i and y_i denote the input and output signal of the i th user, respectively. $N_i \sim CN(0, 1)$ is independent and identically distributed Gaussian noise that impairs receiver i.

IEEE *Access*

Each x_i has an associated average power constraint P_i so that $\frac{1}{m}\sum_{i=1}^{m}|x_m^i|^2 \leq P_i$. Receiver i observes $\hat{y}_i$ and estimates the transmitted message $\hat{x}_i$. The average probability of error for user i is $\epsilon_m^i = \mathbb{E}[p(\hat{s}_i \neq s_i)]$, where expectation is over the random choice of message. The channel output at each receiver is a noisy linear combination of its desired signal and the sum of the interfering terms, of the form [32]:

$$y_i = x_i + \sqrt{\frac{\text{INR}}{\text{SNR}}} \sum_{j=1, j\neq i}^{m} x_j + N_i, \forall j, i = 1, 2, \ldots, m \quad (8)$$

where at the discrete index t, y_i and N_i are the channel output and AWGN respectively, at the $i\,th$ receiver and the x_i is the channel input symbol at the $i\,th$ transmitter. All symbols are real and the channel coefficients are fixed. The AWGN is normalized to have zero mean and unit variance and the input power constraint is given by [32]:

$$E[(x_i)^2] \leq \text{SNR}, \ \forall \ i \in m. \quad (9)$$

The INR is defined through the parameter α [32]:

$$\frac{\log(\text{INR})}{\log(\text{SNR})} = \alpha \ \rightarrow \ \text{INR} = \text{SNR}^\alpha \quad (10)$$

Note that the definition of INR ignores the fact that there are m-1 interferers observed at each receiver. This is for two reasons. First, this definition parallels that of the two-user case [24], which will make it easier to compare the two rate regions. Second, the receivers will often be able to treat the interference as stemming from a single effective transmitter, via interference alignment. This is not the case when the receiver treats the interference as noise. In this work, the introduced parameter $\alpha > 0$ defined by INR $=$ SNR$^\alpha$; this coupling parameter α is used to specify the corresponding linear deterministic model in [25].

In this work, different interference scenarios are studied, from noisy, weak, moderate, strong to very strong interferences. The classification of the interference is defined in [32], and we use the same definition in this work. The degrees-of-freedom (GDoF) of the symmetric m-user interference channel is used to identify the multiple-user channel with regard to the interference level. The definition is given as follows, except for a singularity at $\alpha = 1$:

$$d(\alpha) = \begin{cases} 1 - \alpha, & 0 \leq \alpha < \frac{1}{2} \ \text{(noisy)} \\ \alpha, & \frac{1}{2} \leq \alpha < \frac{2}{3} \ \text{(weak)} \\ 1 - \frac{\alpha}{2}, & \frac{2}{3} \leq \alpha < 1 \ \text{(moderate)} \\ \frac{1}{K}, & \alpha = 1 \\ \frac{\alpha}{2}, & 1 < \alpha < 2 \ \text{(strong)} \\ 1, & \alpha \geq 2 \quad \text{(very strong)} \end{cases} \quad (11)$$

The proposed AE for multiple transmitter and receiver pairs is shown in fig. 1. Recall the m-user interference channel shown in Eq. (1). We consider m pairs transmitters and receivers, which are interacting each other. A two-user (two pairs) AE model is introduced in [9], which is given by:

TABLE1: The structure of the MLP AE

Block name	Layer name	Output Dim
Block name	input:	M
	Dense+eLu	M
	Dense+Linear	$2n$
	nomalization	$2n$
m-user channel	Noise	$m*2n$
Decoder	Dense+ReLU	M
	Dense+Softmax	M

Name	$[\sigma(u)]_i$	range
Linear	U_i	$(-\infty, \infty)$
ReLU	$\max(0, U_i)$	$[0, \infty)$
Tanh	$\tanh(U_i)$	$(-1, \ 1)$
Softmax	$\frac{e^{u_i}}{\sum_j e^{u_j}}$	$(0, \ 1)$
Sigmoid	$\frac{1}{1+e^{-u_i}}$	$(0, \ 1)$

Algorithm 1: ADL algorithm to predict the interference

Input : • AE model and specifications: n, k, batch size, epochs number, optimizer, learning rate, etc
    • the training data set l_{in}
    • the variance of channel noise σ^2

Output: • the estimated interference parameter α

1 Initialize:
2 Set AE model parameters (e.g., $n \leftarrow 4$, $k \leftarrow 4$, $M \leftarrow 4$)
3 **for** i in range (training data samples) **do**
4   Set $x = f(s_i) \in \mathbb{R}^{2n}$, $s_i \in \{1, 2 \ldots M\}$, encoding
5   Create and Set $\hat{y}(n)$ for receiver layer
6   **for** i in range (numble of guessing α) **do**
7   DNN layer setting of the training data (settings in Table I)
8   Recovery pilot signal $\hat{s}_i$ according to a guessing α.
9   Calculate *reward* $\hat{R}_i$ according to Eqs. (16) and (17)
10 Set confidence interval of $\hat{R}_i$ and predict α
11 Update DNN layer with α according to Eqs. (8) to (10)

$$y_1 = x_1 + x_2 + N_1 \quad (12)$$

$$y_2 = x_2 + x_1 + N_2 \quad (13)$$

Eqs.(12) and (13) can be rewritten in a general format with a channel gain, as follow:

$$y_1 = x_1 + g_{21}x_2 + N_1 \quad (14)$$

$$y_2 = x_2 + g_{12}x_1 + N_2 \quad (15)$$

where g_{21} and g_{12} are the channel gain that from transmitter 2 to receiver 1 and from transmitter 1 to receiver 2 respectively. Depending on the values of g_{21} and g_{12} [26], the two-user Gaussian interference channel is classified into weak, strong, mixed, one-sided, and degraded Gaussian interference channel.Briefly, if $0 < g_{21} < 1$ and $0 < g_{12} < 1$, then

the channel is called weak Gaussian interference channel. If $1 < g_{21}$ and $1 < g_{12}$, then the channel is called strong Gaussian interference channel. If either $g_{21} = 0$ or $g_{12} = 0$, the channel is called one-sided Gaussian interference channel. If $g_{21} = g_{12} = 1$, then the channel is called degraded Gaussian interference channel. If either $0 < g_{21} < 1$ and $1 \leq g_{21}$, or $0 < g_{12} < 1$ and $1 \leq g_{12}$, then the channel is called mixed Gaussian interference channel. If $g_{21} = g_{12}$ and $P_1 = P_2$, the channel is defined as the symmetric Gaussian interference channel. The two-user (two pairs) AE model introduced in [9], where $g_{21} = g_{12} = 1$, is a two-user symmetric Gaussian channel under a strong interference condition. In this work, the model is derived into a more general scenario for m-pair transceivers Gaussian interference channel. Two types of channels are studied in this work: symmetric and asymmetric Gaussian interference channels.

As shown in Eq. (8), let's assume that $i, j \in \mathbf{M}$, $i \neq j$, $g_{i,j}$ is the co-channel interference channel coefficient (gain), x_j and y_j are the transmitted and received signals of the transmitter j and receiver j, respectively. N_j is a Gaussian noise vector with independent and identically distributed components of zero mean and variance $\sigma^2 = 1$. The transmitted signals has a power constraint P that $\mathbb{E}\left[||x_j||^2\right] \leq KP$. Transmitter j wants to send a message S_j which is a random variable uniformly distributed over the message set $S_i = \{1, 2, \ldots, 2^{KR_j}\}$ to receiver j by using a code of length K channel uses. Thus, it encodes S_j into $x_j \in \mathbb{R}^K$ and send x_j. After K channel observations, the receiver obtains y_j from which is decodes $\hat{S}_j$. An error event occurs if $S_j \neq \hat{S}_j$ for some $j \in \{1, 2, \ldots, m\}$, and a probability $\mathbb{P}_e^K$. The NN based AE is proposed to replace the PHY structure of the channel. Through learning, the receiver can jointly estimate the gain, interference, and do the detection simultaneously.

C. ADAPTIVE DEEP LEARNING ALGORITHM BASED RECEIVER BLOCK

The receiver block is shown in Fig.1. After propagating through an AWGN channel, the received signal $y(n)$ consists of the originally transmitted signal, the channel response, AWGN noise, and the interference from other sources. The received n-dimensional signal $y(n)$ noised by a channel can be modelled using a conditional probability density function $p(y|x)$, and the DNNs receiver subsequently learns it with multiple dense layers. The last layer of the receiver is a Softmax activation layer which outputs an M-dimensional probability vector p, in which the sum of its elements is equal to 1. The receiver first applies the transformation $f: \mathbb{R}^{2n} \rightarrow M$ to decode the signal, creating a signal $\hat{s}_i$ to recover the original transmitted signal s_i. For the structure of AE, to allow a benchmark for comparison, we use the similar AE structure settings as in [7], which are based on a multi-layer perceptron (MLP). ReLU and Softmax are used in DNNs layer. The specifications are listed in Table I.

An adaptive learning algorithm integrated with DNNs based receiver block, named *ADL algorithm*, was designed and proposed to mitigate the dynamic interference. After the

learning processing, the interference coupling parameter α can be estimated. With a predicted α, the channel function is obtained, following by Eqs. (8) to (10). Then the DNN layer is updated with this knowledge by substituting α into Eq. (8). This updating process has two steps. Firstly, a group of pilot signals is used for DNN training to predict the real-time α. Then, with this knowledge of the interference channel, channel function is updated, and DNN layers are updated with a new set of parameters for decoding signals. To choose suitable pilot signal, an adaptive pilot design is studied in [27], which can be integrated with the proposed ADL algorithm to jointly design the pilot signals and the channel estimator.

The transmitted signals contain two parts. The first part is the pilot signal, which is used for the training data set. The second part is the signal, which is used for communication. It has a similar structure as a DL based OFDM system, which is studied in [29]. However, in our proposed method, the pilot signals are used for both interference estimating and the DNN training. We introduce and explain the proposed ADL algorithm in Algorithm I. At the initialization stage, the specifications of an (n, k) AE are set, and the input training data set is loaded. After that, the DNN layer encodes the data into the messages, propagating through an AWGN channel. After a group of signals are captured by a DNNs based receiver, the learning algorithm at the receiver block starts to train the pilots and then decode the messages after a learning loop. By process of reward computation, the block normalizes the reward regarding different guessing values of α. Then the optimum α range is determined, with regard to a predefined confidence interval. Based on the plot of the reward according to the guessing α, the predicted α is obtained by computing the mean value. Next, the estimated α is substituted back into the DNNs learning layer for the decoding process with updated parameter sets. In this work, the normalized reward is defined as follows:

$$\hat{R}_i = \frac{R_i}{||R_i||} \tag{16}$$

where

$$R_i = \frac{1}{BER|_{Pilot \leftarrow (1, \ldots, i)}} \tag{17}$$

R_i is defined as the reciprocal of the mean bit error rate (BER) value for i pilots signals.

The reward calculation includes two steps. First of all, the pilot signal is decoded using different guessing value of α, e.g. ascending from $\alpha = 0$ (non-interference) to $\alpha = 3$ (very high interference channel). For each guessing α, the pilot dataset is trained using its channel function, which is associated with the guessing α, as in Eq. (8). In each iteration of α, DNN network trains the datasets and update the prediction function. In the second step, the pilot signal is recovered. The bit error rate (BER) of the recovered pilot signal is computed. Based on the BER, the reward is calculated following the definition in Eqs. (16) and (17). After a loop of DNN training (according to a range of guessing α), we have a set of reward

R_i. Then we define the confidence interval of the R_i and find out the peak of the reward. The mean of reward is calculated. Based on the mean, an optimal α is predicted. For this specific α, a DNN trained predict model is determined, which update the DNN layer for decoding the received signal. For this prediction process based on the reward performance, we give a case study with details in the Section of Numerical Evaluation.

III. SYSTEM PERFORMANCE EVALUATION

In this section we present extensive simulation studies of our proposed system operating in a range of multi-user interference channel conditions. All simulations were carried out using Python, with the libraries of PyTorch, TorchNet and TQDM. Training was done at a fixed value of E_b/N_0 = 7 dB using Adam [35] with a learning rate of 0.001. Activation functions rectified linear units (ReLU) [36] and Softmax are used in our DNNs layer. The details are listed in Table I. Detailed explanation of these can be found in [37]. The pilot symbol ratio we used in our simulation is 0.01. The group number of the bit streams is 30, which is used for jointly training and estimating the interference α.

A. CONSTELLATION STUDY

We study the constellation of the AE in different setting (n, k), where the link operates at the communication rate of $R_c = k/n$ [bits/channel use]. The AE can be split to two parts: encoder and decoder, after training the model in end-to-end manner. Then the encoder part is implemented at the transmitter side which generates encoded symbols for each message to be sent over the channel and decoder part is implemented at the receiver, which regenerates the messages from the received symbols. After completion of model training, encoder can generate all possible output signals for each message in the message alphabet. Fig.2 (a) to (h) show learned constellations for different systems we tested. When mapping $2n$-dimensional output from the encoder model to the n-dimensional complex valued vector x, the odd indexed elements of x are taken as in-phase (I) components and even elements of x are taken as quadrature (Q) components. In the scatter plots, I and Q values are plotted in x- and y-axes respectively. For example, in a 4-4 AE setting, $n=4$ and $k=4$, for a testing 16 messages using one hot vector matrix, the input 16 messages have a size of 16×16. A one hot encoding is a representation of categorical variables as binary vectors. One hot encoding is used in ML as a method to quantify categorical data. This method produces a vector with length equal to the number of categories in the data set. More details of one hot encoding can be found in [39]. After the learned encoder, the output of the encoder has 16 messages in a matrix with a size of 16×8, where odd indexed elements (column) are I components and even elements are Q components. Therefore, for each row of the date, AE using four symbols to represent 4 bits signal. For comparison, we set (n,k) in different values. Fig. 2 shows the constellation

for AE-1-1 to AE-1-5. And the constellation of each symbols for the case of AE-4-4 is plotted in the Fig. 3.

We first study the AE in a single-user case without interference. The constellation results are plotted in Fig.2. It shows that with different settings (n, k), the AE find the optimal constellation shape via a learning processing. It notices that the system predict a psk modulation shape constellation for AE-1-1, AE-1-2, AE-1-3, AE-1-4 and AE-1-5 via a sufficient learning. However, the constellation shapes of AE-3-3 and AE-4-4 look different. As we introduced above, n defines the dimension of the complex channel size. When $n = 1$, the output of the encoder has two columns, which present I and Q values, respectively. It is similar as the definition of a psk constellation. However, when we apply $n > 1$, the output of encoder has multiple columns, which represent the messages parallel. In another word, AE use longer symbols to represent messages simultaneously. E.g, for Fig. 2(c), AE-3-3. AE use every 8 dots as a symbol to represent one message, and totally 8 symbols are used for all (2^3) messages. Similarly, one symbol of single use case AE-4-4 is shown in Fig. 2(d).

Then, we study the constellation of the AE for the interference channel with multiple-pair transceivers, and analyze the effect from low to high interference conditions. In particular, for the high interference scenario, we demonstrate how does the AE overcome the interference and predict the corresponding constellation via learning. In this study, the inverse one hot vector format is used for the AE. For a two-user case, we first set the weak interference condition where $g_{12} = g_{21} = 0.5$, the constellation of an AE (4, 4) is plotted in Fig.3 (a). It shows that the constellation points (blue for user 1 and red for user 2) locate randomly but it seems that they concentrate toward to its own cluster on the I and Q map. For the dots of different users, it notices that the distances between them are quite small for low interference condition. However, for high interference scenario, the constellation dots of different user move and concentrate toward to its own cluster area which helps the receiver to decode the signals against the interference. Based on the BER performance, we notice that for the low interference scenario, although the dots from different users are quite close to each other on the I and Q map, the AE still holds the capability to decode the signals against the interference from other user. To verify this, the BER performance is plotted in next Section. Similar observation can be found in more users based cases. Fig. 4 (a) and (b) depicts the changes of learning constellation for a four-user case, AE (4, 4) for low and high interference conditions respectively. We can notice that similar performance is observed. For the low interference scenario, Fig. 4(a), the four symbol clusters (16 dots for each symbol) from four different users are close to each others. However, the decoder still can sufficiently recover the signals. In contrast, for the high interference condition (Fig. 4(b)), the clusters are separate, and concentrate toward to its own area. It reveals that the learning based encoder prevent the high interference from the constellation design.

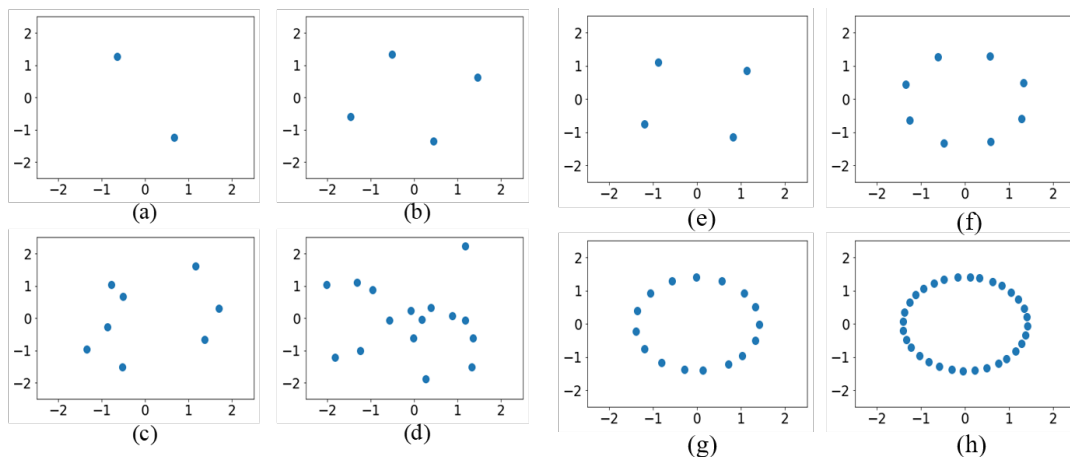The propsoed AE is based on the format of one hot

FIGURE2: The Learned AE constellation produced by AE for single user case: (a) AE-1-1, (b) AE-2-2, (c) AE-3-3 and (d) AE-4-4. (e) AE-1-2, (f) AE-1-3, (g) AE-1-4, (h) AE-1-5.
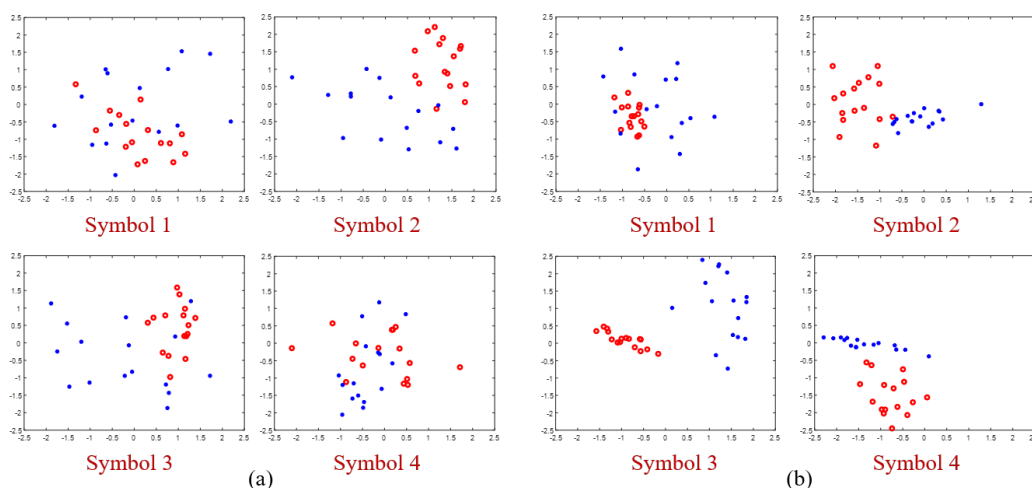

FIGURE3: The Learned constellation of AE for two-user case. Scatter plots of learned constellations for $k=4$, $n=4$ system, encoder of symbol 1 to symbol 4: (a) weak interference $g_{12} = g_{21} = 0.5$ and (b) very strong interference $g_{12} = g_{21} = 3$.

vector or inverse one hot as the compressed techniques. However, other studies suggest some alternative approaches to minimize the computational complexity when vectors get larger. E.g., dense vector methods that are used in Natural language processing (NLP) [38] or efficient sparse vector representations could be used reduce the complexity. This is interested and we will investigate it in our future work.

B. COMPARISON BETWEEN THE DL BASED AE AND CONVENTIONAL N-PSK AND N-QAM FOR SINGLE USER CASE

Recall the results in Fig. 2, we notice that an AE based system has the capability to generate the optimum constellation according to its channel condition. For example, for a single-user channel, with a maximum power constraint, AE based system generates the n-PSK shape constellation for the

encoder via learning. However, to compare the difference, bit error rate (BER) and symbol error rate (SER) against signal noise ratio ($SNR(E_b/N_0)$) are plotted in Fig. 5. For BER of AE, we can see that AE-1-1 and AE-1-2 have a similar performance, and they are very close the performance of uncoded BPSK and QPSK. For a fair comparison, the setting of AE-n-k is equivalent to a conventional modulation scheme following the communicate rate formula $R = k/n$, where n is the complex channel uses per message. The equivalent formula is given as: $2^{k/n}$-QAM. Based on this, the equivalent modulations of AE-1-1 and AE-1-2 are BPSK and QPSK. AE-1-3 and AE-1-4 are compared to 8-psk and 16-QAM, respectively. As shown in Figs. 5 (a) and (b). We can see that the AE-1-3 and AE-1-4 have around 2 dB improvement compared to the conventional schemes 8-psk and 16-QAM.
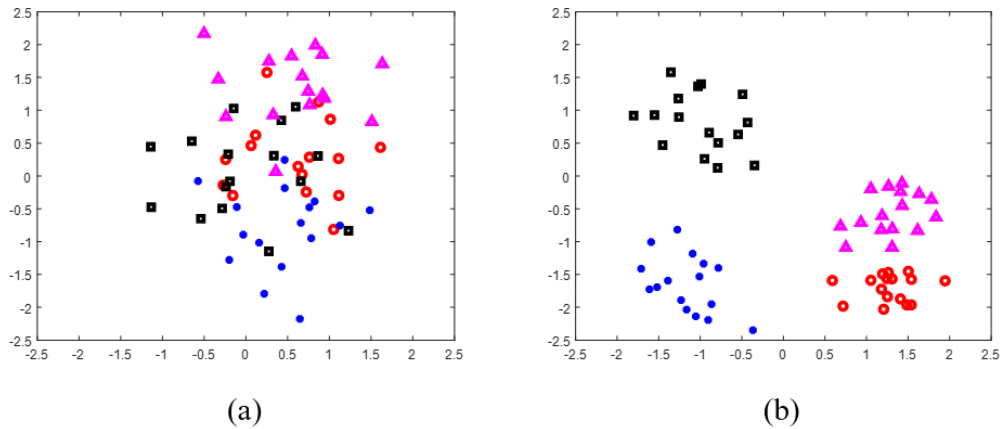
IEEE *Access*

FIGURE4: The Learned constellation of AE for four-user case. Scatter plots of learned constellations for $k = 4$, $n = 4$ system, encoder of symbol 1: (a) weak interference and (b) very strong interference.
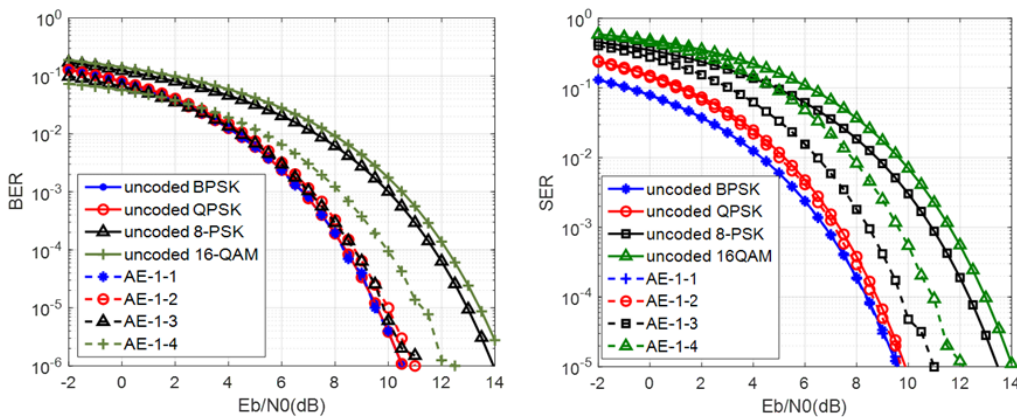

FIGURE5: Bit error rate and symbol error rate vs SNR (E_b/N_0) for the AE and other modulation schemes (single user case).

C. CONVENTIONAL N-PSK AND N-QAM MODULATION WITH EQUALIZATION TECHNIQUES

In this section, we study the two equalization techniques ZF and MMSE applied with the conventional modulation schemes for the interference channel. We first evaluate the improvement against the co-channel interference induced by two pairs of users. As shown in Fig. 6 (1), both ZF and MMSE equalizers work well against the interference at a low interference level where $g_{12} = 0.2$. However, they don't performance well for the high level interference. It is also noticed that MMSE equalizer has a slight better performance than the ZF for a strong interference. However, it is more notable for smaller E_b/N_0 values. In Fig. 6 (b), the plots give more details of how much the performance degraded with an increased interference gain g_{12}. The results indicate that for a high level of interference, the conventional ZF and MMSE equalizers have a limited capability for mitigating the interference. Therefore, an adaptive algorithm based AE is proposed for this scenario. In next section, we will evaluate

the n-psk and QAM modulations with MMSE equalizer, and compare its performance to the proposed AE scheme.

D. TWO-USER CASE, SYMMETRIC AND ASYMMETRIC INTERFERENCE CHANNEL

Figs. 7 (a) and (b) show the BER performance of some different AE schemes for two-user symmetric and asymmetric interference channels respectively. Together, the equivalent conventional psk and QAM modulations with MMSE are plotted too for comparison. In this simulation, g_{12} is set as 1 for the symmetric interference channel and $g_{12}=0.5$ for the asymmetric one. The results show that the AE demonstrates a promising solution for the strong interference, even in the case that conventional MMSE equaliser doesn't work. The improvement is quite significant and notable.

E. INTERFERENCE ESTIMATION

In this section, we evaluate the robustness of the AE scheme and introduce the interference estimation processing with the
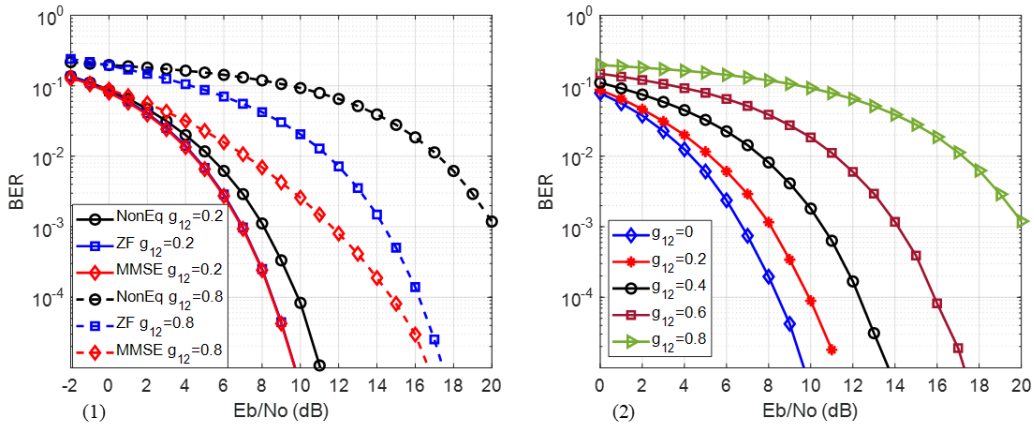
FIGURE 6: BER of BPSK two-user co-channel interference with ZF and MMSE equalization.
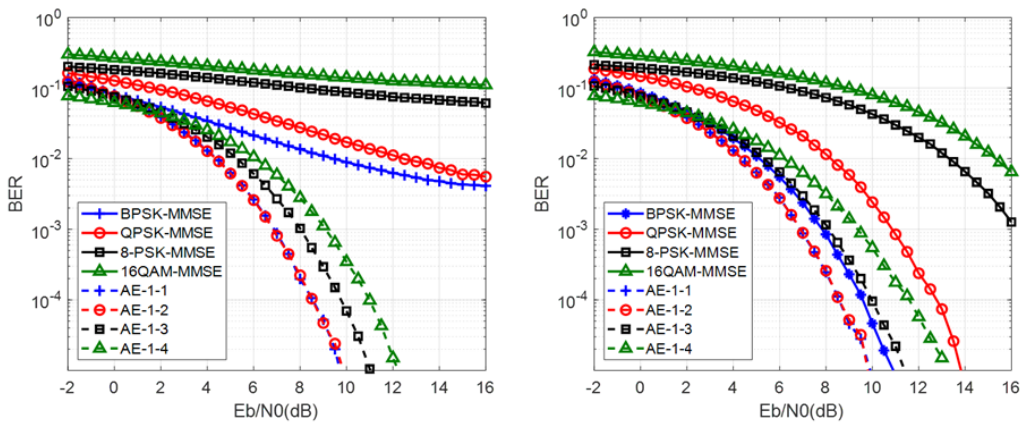

FIGURE 7: Bit error rate vs SNR (E_b/N_0) of AE and several modulation schemes with MMSE equalizer for two-user symmetric and asymmetric interference channel.
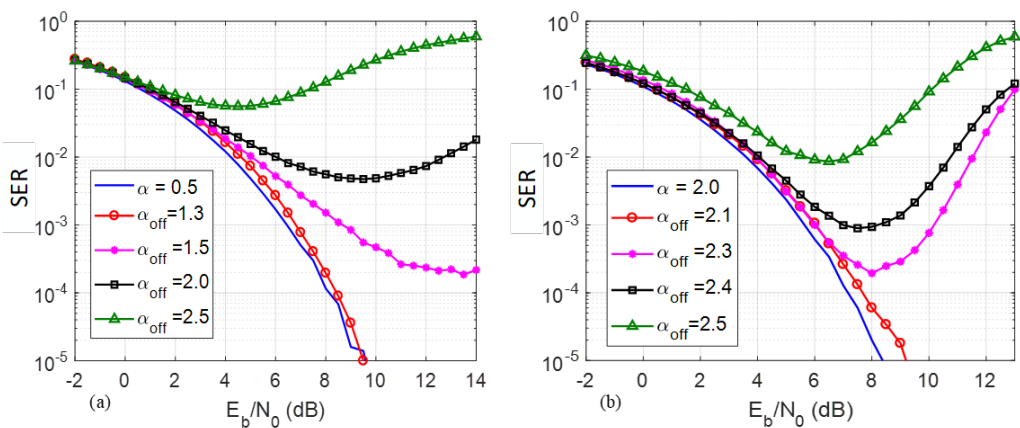

FIGURE 8: SER versus SNR performance of an AE (4, 4): (a) weak interference $\alpha = 0.5$ with offset up to $\alpha = 2.5$, (b) very strong interference $\alpha = 2.0$ with offset up to $\alpha = 2.5$.

proposed ADL algorithm.

A DL based AE with different settings of (n, k) are studied

and evaluated in [9]. It compares the performance between the M-QAM modulation and AE with similar settings. It
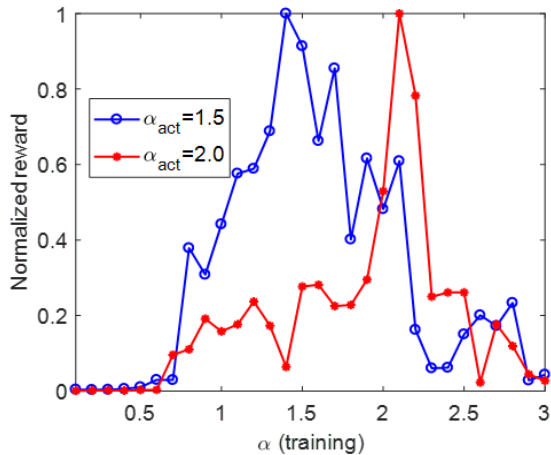
IEEE Access

FIGURE9: Normalized reward versus predicted α: strong interference $\alpha = 1.5$ and very strong interference $\alpha = 2.0$.
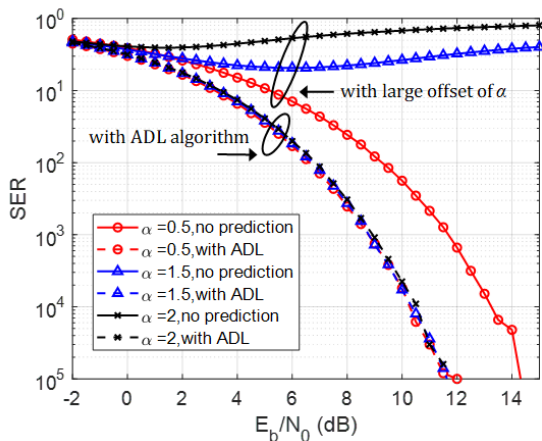

FIGURE10: SER versus SNR: comparison for strong and very strong interference channel, with and without the proposed ADL algorithm.

demonstrates that the AE (n, k) outperforms the equivalent n-psk or QAM. To enable a benchmark for comparison, we use the similar setting AE-4-4 throughout this section. However, we evaluate the performance according to our proposed interference model, as shown in Eqs. (8)-(11). We verify our algorithm through an example of a two-user interference channel case. For other multi-user case, the methodology is similar, and the enhancement is more significant.

We demonstrate that the AE approach has some robustness when it applies in an interference channel. However, we also want to characterize the robustness for difference interference strengths. It assumes that the system knows the interference channel generalized formula (Eq.(8)) and it applies a DL training for the decoder. We train the model with a predetermined α. However, we assume that α may change dynamically in a real time scenario and we want to evaluate

how robust of the decoder when α has some offset, denote as α_{off}.

Following the definition in Eqs.(5) to (8) , we simulate for weak ($\alpha = 0.5$) and very strong interference ($\alpha = 2$) respectively. Results are plotted in Figs. 8 (a) and (b). It shows that the AE approach is quite robust for a weak interference. The system works even under a very large offset: 3 times of the training α. However, the situation is slightly different for very strong interference, where $\alpha = 2$. The result in Fig. 8 (b) indicates that the system is quite sensitive to the offset under a very strong interference channel. For this scenario, it does require a technique to deal with the interference. To address this, we apply the proposed ADL algorithm and the performance evaluation is given in the next section.

Recall the proposed ADL algorithm in section II. We evaluate the ADL algorithm to estimate α in different interference strengths. We also carried more groups of study similar as above, and we found that for strong ($\alpha = 1.5$) and very strong ($\alpha = 2$) interference, the offset of α becomes more critical. Therefore, we address this and implement our algorithm for these cases. With the same setting in Figs. 8 (a) and (b), we plot the normalized reward versus a predicted α (different values at training), in Fig. 9. for $\alpha = 1.5$ and $\alpha = 2$, respectively. We can see that the peak value of the normalized reward appears around 1.5 (actual value), and it reduces gradually to both sides of the actual value. By contrast, for the very strong interference, where $\alpha = 2$, we can also found out the peak value of the normalized reward appears around the real value of α. However, it decreases rapidly towards both sides of the actual value, which agree with the achievement that it's more sensitive to the offset. As the fluctuation is quite large in Fig. 9, here we define 40% offset as the confidence interval of the reward, to estimate α. We use the mean α for evaluating the performance, as we introduced in Section II. Furthermore, the reward is computed according to the instant SNR condition. For this simulation, we use $E_b/N_0 = 7$ dB as an example. To evaluate the performance with and without applying the proposed ADL algorithm, we plot the SER performance for weak, strong and very strong interference channels for comparison, as shown in Fig. 10. In this simulation, we take a large interference effect as an example, $\alpha_{\text{off}} = 2\alpha$, to demonstrate the improvement achieved by our algorithm. Two groups of data are highlighted in Fig. 10. We can see that the SER significantly degrades due to the large offset of α. In particular, for the strong and very strong interference cases, the system does not work without the knowledge of α. However, with applying the ADL algorithm, the result shows that with an efficient interference prediction, the ADL algorithm based AE is capable of robust performance over the entire range of interference levels, even for the worst case in a very strong interference channel.

IV. CONCLUSION

An ADL algorithm based AE is proposed for a m-user interference channel with unknown interference. With the proposed ADL algorithm, interference can be estimated and

predicted, which is subsequently used for updating the DNN based decoding processing. The constellation of the proposed AE scheme has been studied for m-user interference channel. Our findings reveal the promising compressed technique to minimize the complexity when vectors get larger. The proposed algorithm shows the significant enhancement on the robustness of the system against interference and provides an AE system that is adaptable to real-time interference scenario, for the entire range of interference levels. The enhancement is more notable for strong and very strong interference scenarios, compared to the performance of conventional AE with offline learning. The performance also outperforms the conventional ZF and MMSE equalizer approaches. The proposed approach has laid the foundation of enabling adaptable constellation for 5G and beyond communication systems, where dynamic and heterogeneous network conditions are envisaged. Studies on the AE generated constellations which adapt according to varying channel conditions, is substantially different from conventional constellations (e.g., QAM signals), where the static magnitude and phase is expected under any channel conditions. Such an approach has given interesting insights into future adaptable constellation design using AI. Our future work aims at improving computational efficiency of our online learning scheme, and the implementation on real-life platforms.

ACKNOWLEDGMENT

REFERENCES

[1] R, Yuji, G, Heo and S.E, Whang, "A survey on data collection for machine learning: a big data-ai integration perspective," IEEE Transactions on Knowledge and Data Engineering, 2019.

[2] S, Wojciech, "A survey on data collection for machine learning: a big data-ai integration perspective," IEEE Transactions on Knowledge and Data EngineeringExplainable AI: interpreting, explaining and visualizing deep learning, Vol. 11700, 2019.

[3] Y. Wang et al., "Network management and orchestration using artificial intelligence: Overview of ETSI ENI," IEEE Communications Standards Magazine, vol. 2, no. 4, pp. 58-65, 2018.

[4] N. Farsad and A. Goldsmith, "Neural network detection of data sequences in communication systems," IEEE Transactions on Signal Processing, vol. 66, no. 21, pp. 5663-5678, 2018.

[5] H. Huang, J. Yang, H. Huang, Y. Song, and G. Gui, "Deep learning for super-resolution channel estimation and DOA estimation based massive MIMO system," IEEE Transactions on Vehicular Technology, vol. 67, no. 9, pp. 8549-8560, 2018.

[6] E. Nachmani, Y. Be'ery, and D. Burshtein, "Learning to decode linear codes using deep learning," in IEEE Communication, Control, and Computing (Allerton), 54th Annual Allerton Conference, pp. 341-346, 2016.

[7] H. Ye, G. Y. Li, and B.-H. Juang, "Power of deep learning for channel estimation and signal detection in OFDM systems," IEEE Wireless Communications Letters, vol. 7, no. 1, pp. 114-117, 2018.

[8] N. Farsad, M. Rao, and A. Goldsmith, "Deep Learning for Joint Source-Channel Coding of Text," arXiv preprint arXiv:1802.06832, 2018.

[9] T. O'Shea and J. Hoydis, "An introduction to deep learning for the physical layer," IEEE Transactions on Cognitive Communications and Networking," IEEE Transactions on Cognitive Communications and Networking, vol. 3, no. 4, pp. 563-575, 2017.

[10] F. A. Aoudia and J. Hoydis, "End-to-End Learning of Communications Systems Without a Channel Model," arXiv preprint arXiv:1804.02276, 2018.

[11] M. Goutay, F. A. Aoudia, and J. Hoydis, "Deep Reinforcement Learning Autoencoder with Noisy Feedback," arXiv preprint arXiv:1810.05419, 2018.

[12] F. Qamar, M.H.D.N. Hindia, K. Dimyati, et al, "Interference management issues for the future 5G network: a review," Telecommun Syst, vol. 71, no. 4, pp. 627–643, 2019.

[13] B. Soret, A. D. Domenico, S. Bazzi, N. H. Mahmood and K. I. Pedersen, "Interference Coordination for 5G New Radio," in IEEE Wireless Communications, vol. 25, no. 3, pp. 131-137, JUNE 2018.

[14] J. Xiao, C. Yang, A. Anpalagan, Q. Ni and M. Guizani, "Joint Interference Management in Ultra-Dense Small-Cell Networks: A Multi-Domain Coordination Perspective," in IEEE Transactions on Communications, vol. 66, no. 11, pp. 5470-5481, Nov. 2018.

[15] P. Fan, J. Zhao and C. I., "5G high mobility wireless communications: Challenges and solutions," in China Communications, vol. 13, no. Supplement2, pp. 1-13, 2016

[16] Z. Li, L. Song, and H. Shi, "Approaching the capacity of K-user MIMO interference channel with interference counteraction scheme," Ad Hoc Networks, vol. 58, pp. 286-291, 2017.

[17] S. Dörner, S. Cammerer, J. Hoydis, and S. ten Brink, "Deep learning based communication over the air," IEEE Journal of Selected Topics in Signal Processing, vol. 12, no. 1, pp. 132-143, 2018.

[18] T. J. O'Shea, T. Roy, N. West, and B. C. Hilburn, "Physical Layer Communications System Design Over-the-Air Using Adversarial Networks," rXiv preprint arXiv:1803.03145, 2018.

[19] F. Restuccia, T. Melodia, " Physical-Layer Deep Learning: Challenges and Applications to 5G and Beyond," arXiv preprint arXiv:2004.10113, 2020.

[20] T. J. O'Shea, T. Roy, and TC. Clancy, " Over-the-air deep learning based radio signal classification," IEEE Journal of Selected Topics in Signal Processing, vol. 12, no. 1, pp. 168-179. 2018.

[21] F. Restuccia, S. D'Oro, A. Al-Shawabka, M. Belgiovine, L. Angioloni, S. Ioannidis, K. Chowdhury, and T. Melodia, " DeepRadioID: Real-Time Channel-Resilient Optimization of Deep Learning-based Radio Fingerprinting Algorithms," in ACM MobiHoc, pp. 51–60. 2019.

[22] O. Naparstek and K. Cohen, " Deep Multi-user Reinforcement Learning for Distributed Dynamic Spectrum Access," IEEE Transactions on Wireless Communications, vol. 18, no. 1, pp. 310–323, 2019.

[23] M. Sadeghi and E. G. Larsson, "Physical Adversarial Attacks Against End-to-End Autoencoder Communication Systems," IEEE Communications Letters, 2019.

[24] I. J. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and harnessing adversarial examples," arXiv, 2018.

[25] K. Gomadam, V. R. Cadambe, and S. A. Jafar, "A distributed numerical approach to interference alignment and applications to wireless interference networks," IEEE Transactions on Information Theory, vol. 57, no. 6, pp. 3309-3322, 2011.

[26] C. Geng, N. Naderializadeh, A. S. Avestimehr, and S. A. Jafar, "On the optimality of treating interference as noise," IEEE Transactions on Information Theory, vol. 61, no. 4, pp. 1753-1767, 2015.

[27] X. Ma, and Z. Gao, "Data-Driven Deep Learning to Design Pilot and Channel Estimator for Massive MIMO," IEEE Transactions on Vehicular Technology, vol. 69, no. 5, pp. 5677-5682, 2020.

[28] T. Erpek, T. J. O'Shea, and T. C. Clancy, "Learning a Physical Layer Scheme for the MIMO Interference Channel," in 2018 IEEE International Conference on Communications (ICC), pp. 1-5, 2018.

[29] E. Balevi and J. G. Andrews, "One-bit OFDM receivers via deep learning," IEEE Transactions on Communications, 2019.

[30] H. Sun, X. Chen, Q. Shi, M. Hong, X. Fu, and N. D. Sidiropoulos, "Learning to optimize: Training deep neural networks for interference management," IEEE Transactions on Signal Processing, vol. 66, no. 20, pp. 5438-5453, 2018.

[31] D. Wu, M. Nekovee, and Y. Wang, "An adaptive deep learning algorithm based autoencoder for interference channels," International Conference on Machine Learning for Networking, pp. 342-354. Springer, Cham, 2019.

[32] S. A. Jafar and S. Vishwanath, "Generalized degrees of freedom of the symmetric Gaussian K user interference channel," IEEE Transactions on Information Theory, vol. 56, no. 7, pp. 3297-3303, 2010.

[33] R. H. Etkin, N. David, and H. Wang, "Gaussian interference channel capacity to within one bit," IEEE Transactions on information theory, vol. 54, no. 12, pp. 5534-5562, 2008.

[34] O. Ordentlich, U. Erez, and B. Nazer, "The approximate sum capacity of the symmetric Gaussian K-user interference channel," IEEE Transactions on Information Theory, vol. 60, no. 6, pp. 3450-3482, 2014.

[35] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," arXiv preprint arXiv:1412.6980, 2014.

[36] X. Jin, C. Xu, J. Feng, Y. Wei, J. Xiong, and S. Yan, "Deep Learning with S-Shaped Rectified Linear Activation Units," in AAAI, vol. 3, no. 2, p. 3.2. 2016.

[37] I. Goodfellow, Y. Bengio, A. Courville, and Y. Bengio, "Deep learning," MIT press Cambridge, 2016.

[38] T. Young, D. Hazarika, S. Poria and E. Cambria,"Recent Trends in Deep Learning Based Natural Language Processing," in IEEE Computational Intelligence Magazine, vol. 13, no. 3, pp. 55-75, Aug. 2018.

[39] P. Cerda, G. Varoquaux and B. Kégl, "Similarity encoding for learning with dirty categorical variables," Machine Learning, 107(8-10), pp.1477-1494. 2018.

YUE. WANG is a Senior Technology Manager at Samsung Electronics R&D Institute UK. At Samsung, she works on a variety of technical subjects on 5G RAN and networks, more recently focusing on the research of AI in 5G networks and beyond. She is the Samsung delegate of ETSI ISG ENI, and the Secretary and Rapporteur of ENI. Yue is the recipient of the ISG ENI award in 2019. Yue sits in the Industry Advisory Board of King's College of London and University of Sussex, and is the industry supervisor of a 5-year research programme on AI in 5G with University of Sussex. Prior to joining Samsung, she held various roles in the US and the UK, all on wireless communications. Yue is a Senior Member of IEEE.

• • •
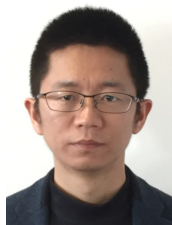
DEHAO. WU is a senior research fellow in the department of Engineering and Design at University of Sussex since April 2018. Before joining University of Sussex, he was a postdoc research fellow at Nanyang Technological University of Singapore, University of Manchester and University of Bournemouth, respectively. He received the Bachelor's degree in optical and information engineering in 2007, M.Sc. degree in microelectronics and telecommunication engineering in 2009 and Ph.D in optical wireless communications from Northumbria University, Newcastle, U.K., in 2013. His current research work focuses on machine-learning algorithms and intelligent management of 5G and beyond-5G. His research interests include the area of optical wireless communications, free space optics, visible light communications, RF communications, indoor high-accuracy positioning, optical sensing and detecting, underwater RF and acoustic communications, information-centric networking and information resilience.

MAZIAR. NEKOVEE is a Professor of Telecoms and Mobile Technology and Head of Centre for Advanced Communication, Mobile Technology, and IoT at School of Engineering and Informatics, University of Sussex, UK. Prior to joining Sussex in 2017 he was a Chief Engineer and Head of 5G Research at Samsung R&D UK. Prior to that he was with British Telecom (BT) Research and Technology, as Senior Scientist and subsequently a Team Leader. His research interests and expertise included 5G and beyond-5G/6G mobile communication RAN and Core design, Machine-learning and AI applied to communication networks, spectrum sharing, cognitive radio, millimeter-wave and THz communications. He has authored over 120 highly cited peer-reviewed papers, one best-selling book "Cognitive Radio Communications and Networks: Principle and Practice" and has 13 patents in telecommunication and mobile technologies. Prof Nekovee is a sought-after speaker at high level industry events, C-level meetings with European operators, policy makers, and international conferences. He is the recipient of a number of industry and academic awards including a Royal Society (UK Academy of Science) Industry Fellowship, BT Innovation Award and Samsung R&D Best Practice of Research Award. Maziar has a PhD in Physics, from University of Nijmegen (NL), and a first degree in Electrical and Electronic Engineering from Delft University of Technology (NL).