Research Article

Efficient Training Scheme for Neural Network Based 4K-QAM Soft Demapper

Donlaporn Triamwichanon[1], **Patinya Muangkammuen[1]**, **Kidsanapong Puntsri[2]**, **Virasit Imtawil[1]**, **Puripong Suthisopapan[1]***

[1] Department of Electrical, Engineering Faculty of Engineering, Khon Kaen University, Khon Kaen, Thailand
[2] Faculty of Engineering, Rajamangala University of Technology Isan Khon Kaen Campus, Khon Kaen, Thailand
 E-mail: purisu@kku.ac.th

Abstract: The practicality of the densely packed and spectrally efficient 4096-Quadrature Amplitude Modulation (4K-QAM) is obstructed by the ultra-high computational complexity of its soft demapper, which is essential for generating the soft outputs required by channel decoders. In this paper, we propose an efficient training scheme to build a highly effective neural network based 4K-QAM soft demapper that can offer significantly lower computational complexity. The results demonstrate that this alternative demapper can achieve comparable decoding performance in coded 4K-QAM systems, while reducing computational complexity by up to 20% compared with the well-known low-complexity max-logarithm of maximum a posteriori (log-MAP) demapper.

Keywords: 4K-QAM, soft demapper, soft demodulator, soft bit, Log-Likelihood Ratio (LLR), machine learning, neural networks, Low-Density Parity-Check (LDPC) codes

1. Introduction

It has been recently shown that neural network can be utilized in modern digital communication system [1–3]. It is well known in communication research and society that quadrature amplitude modulation (QAM) is absolutely essential [4–6]. QAM is very popular to be chosen as the primary modulation scheme for almost all current and upcoming communication standards due to its ability to support high bandwidth efficiency by increasing the modulation order or constellation size, without requiring additional bandwidth [7, 8]. For this work, we focus only on 4K-QAM, which possesses a dense 4,096-point constellation and supports 12 bits per modulated symbol. This ultra-high-order QAM is expected to play an important role in significantly increasing bandwidth efficiency, particularly in the transition from Wi-Fi 6 to Wi-Fi 7 [9]. Moreover, due to its great bandwidth efficiency, this specific modulation scheme has been discussed in other communication contexts [10, 11].

In order to utilize 4K-QAM in modern communication systems, two major challenges are addressed in this paper. The first is the excessively high computational complexity of the receiver. As the QAM order increases, its computational

complexity grows significantly [12]. Thus, this issue becomes particularly serious for 4K-QAM. The second challenge is integrating 4K-QAM with forward error correction (FEC) codes. The dense constellation points in 4K-QAM make it more susceptible to noise and interference, requiring a high performance FEC code and a higher signal-to-noise ratio at the receiver to maintain reliable communication. The Wi-Fi 7 standard addresses this by employing low-density parity-check (LDPC) codes as the FEC scheme for 4K-QAM, due to their superior error correction capabilities. To fully apply these codes in 4K-QAM system, the 4K-QAM receiver must convert a received symbol into soft bits, typically represented as log-likelihood ratio (LLR) [13]. This process is referred to as soft demapping and the receiver can be technically called as 4K-QAM soft demapper.

The logarithm of maximum a posteriori (log-MAP) algorithm, which is the optimal approach, can be straightforwardly employed as a QAM soft demapper to provide the exact LLR calculations [14]. However, the computational complexity of this method increases exponentially with the modulation order, making it impractical for 4K-QAM systems. To address this, the max-log-MAP algorithm has been introduced as an log-MAP approximation [15]. Unfortunately, computational burden of this solution remains high for 4K-QAM. To further reduce complexity, neural network, one of the most popular machine learning techniques, has recently gained attention as an alternative approach [16–20]. From these works, there are strong evidence that the neural network based QAM soft demapper can achieve desirable performance with substantial complexity reduction across a wide range of modulation orders, from 16 to 1,024.

With the motivation from aforementioned works, we further investigate the application of neural networks in 4K-QAM systems. The contributions of this paper are summarized as follows:

1. The methodology for gathering the dataset is presented.
2. A pre-processing technique based on simple min-max scaling that facilitates training process is proposed.
3. A new accuracy metric based on sign agreement is introduced to preliminarily assess neural network performance.
4. Neural network architectures aimed at reducing complexity are explored.

These four aspects are referred to as the efficient training scheme for developing a 4K-QAM soft demapper. The remainder of this paper is structured as follows. Section II describes the system model and problem statement. Section III introduces the application of neural network as QAM demapper. The proposed training scheme is detailed in Section IV. Results are discussed in Section V, and finally, Section VI concludes the paper.

2. System model and problem statement

The system model of coded 4K-QAM system is depicted in Figuer 1. Assuming that FEC code is a type of block code, a block of k information bits, $\mathbf{m} = [m_1, \ldots, m_k]$, is firstly encoded into a codeword, $\mathbf{c} = [c_1, \ldots, c_n]$ at the transmitter side, where $n > k$. The $n - k$ redundant bits are derived from k information bits and this code is referred to as (n, k) block code with coding rate $R = k/n$. Then, the codeword of length n bits is subsequently fed into 4K-QAM mapper. This mapper is defined by complex QAM constellation, $\mathbb{Q}$, that contains $M = 4,096$ modulated symbols in which each symbol $q \in \mathbb{Q}$ is labelled with unique 12 binary digits. This implies that a 4K-QAM symbol can carry $\log_2(M) = 12$ bits and a coded 4K-QAM transmission provides $R\log_2(M)$ bits/channel use. With this kind of mapper, one codeword is mapped into $p = n/12$ modulated symbols, denoted by $\mathbf{q} = [q_1, \ldots, q_p]$.
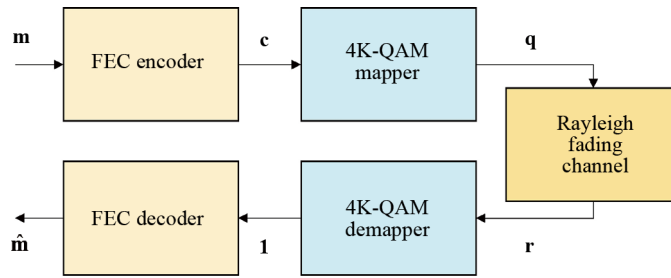
Figure 1. A system model for the communication system utilizing 4K-QAM and FEC. The information is transmitted from the transmitter (upper part) to the receiver (lower part) through a channel (rightmost part)

For this study, the LDPC codes defined by Wi-Fi 7 standard, in which 4K-QAM is considered to be useful, are chosen as FEC code [7]. The modulation constellation is Gray-coded rectangular 4K-QAM. All possible input binary sequences of this model are assumed to be equiprobable. Accordingly, all the modulated symbols in $\mathbb{Q}$ are equally likely to be transmitted. The 4K-QAM constellation is normalized so that unit average transmitted power is achieved.

Considering a baseband discrete time model for an uncorrelated Rayleigh flat fading channel, the received signal, $\mathbf{r} = [r_1, \ldots, r_p]$, for coded 4K-QAM system is given by

$$\mathbf{r} = \mathbf{h} \odot \mathbf{q} + \mathbf{n}. \tag{1}$$

The vector $\mathbf{h} = [h_1, \ldots, h_p]$ denotes the Rayleigh fading vector whose each entry is assumed to be an i.i.d. complex Gaussian random variable with zero mean and unit variance. The operator $\odot$ denotes element-wise multiplication, i.e., Hadamard product. The vector $\mathbf{n} = [n_1, \ldots, n_p]$ is a noise vector whose entry is an i.i.d. complex Gaussian random variable with zero mean and variance σ^2. The signal to noise ratio of this coded system is expressed in terms of the energy per bit to noise spectral density as $R log_2(M) E_b / N_0$.

The 4K-QAM demapper extracts soft outputs in terms of LLR for each coded bit, c_i, by using a MAP relation as follows

$$\ell_i = \ln \left(\frac{\Pr(c_i = 1 \mid r_j)}{\Pr(c_i = 0 \mid r_j)} \right), \ i = 1, \ldots, n, \ j = \lceil i/12 \rceil.$$

Following the channel model defined in (1) and assuming channel side information at the receiver, the calculation of LLR based on above log-MAP formula can be expressed as [16]

$$\ell_i = \ln \left(\frac{\sum\limits_{q \in \mathbb{Q}_d^1} \exp\left(-\frac{\|q - r_j\|^2}{\sigma^2}\right)}{\sum\limits_{q \in \mathbb{Q}_d^0} \exp\left(-\frac{\|q - r_j\|^2}{\sigma^2}\right)} \right), \ d = 1, \ldots, 12, \tag{2}$$

where $\mathbb{Q}_d^1 \subset \mathbb{Q}$ is the subset of complex QAM constellation that contains only modulated symbols whose their d-th binary digit corresponds to bit 1. The subset $\mathbb{Q}_d^0$ can be interpreted similarly but for bit 0. The log-MAP demapper defined by (2) is regarded as the optimal demapper. With this LLR extraction, the 4K-QAM demapper can provide $\mathbf{l} = [\ell_1, \ldots, \ell_n]$ to FEC decoder and the estimation, $\hat{\mathbf{m}}$, is finally made.

The computational complexity of log-MAP demapper grows with the order of constellation size. Therefore, when considering 4K-QAM and beyond, the direct implementation of such an optimal approach is impossible. To alleviate this problem, log-MAP formula can be approximated as follows [16],

$$\ell_i \approx \frac{1}{\sigma^2} \left(\min_{q \in \mathbb{Q}_d^0} \|q - r_j\|^2 - \min_{q \in \mathbb{Q}_d^1} \|q - r_j\|^2 \right). \tag{3}$$

Based on this approximation, the demapper is sub-optimal and known in the literature as max-log-MAP demapper. This demapper is superior than its optimal counterpart since its computational complexity is much lower due to the absence of complex exponential and logarithmic functions. Note that the 4K-QAM demapper converts a complex noisy received symbol into 12 real-valued LLRs, i.e., symbol-to-bit conversion. The max-log-MAP demapper is considered as the conventional demapper and its computational complexity is set to be a benchmark.

3. Neural network based 4K-QAM demapper

The universal approximation theorem suggests that neural network is capable of learning complex relationship in data [21]. Thus, it is intuitive to think that the relationship between the noisy 4K-QAM symbol and its associated LLRs of the 4K-QAM demapper, described in the previous section, can be imitated by a neural network. With this idea, the neural network needs to perform regression task to predict 12 real-valued LLRs from a received symbol. We refer to this kind of neural network as neural network based 4K-QAM soft demapper and its typical architecture is depicted in Figuer 2. It is seen that nodes or neurons in two adjacent layers are connected through the edges and each edge has its own weight, i.e., multiplying constant.
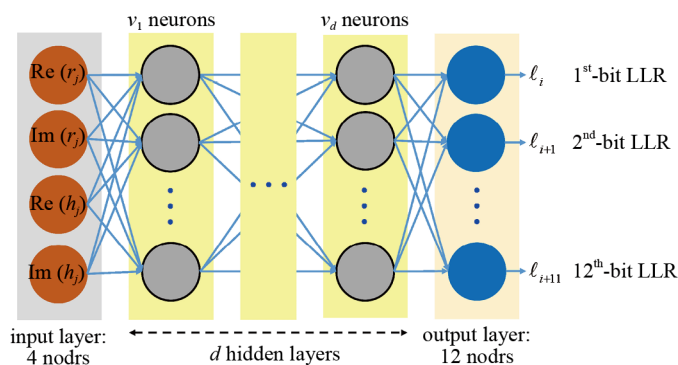

Figure 2. The regression neural network for performing 4K-QAM demapper. Typically, a neural network comprises of three main layers which are input layer, hidden layer and output layer as depicted in this figure. The neural network used in this paper is considered as a deep neural network since two or more hidden layers are employed. The number of neurons in each hidden layers is specified by v_τ, $\tau = 1, \ldots, d$ where d is a number of hidden layers

The above architecture can be briefly described as follows. A complex received symbol r_j and its corresponding complex channel fading coefficient h_j are both firstly prepared in terms of scalars. Mathematically, the vector $\mathbf{x}_0 = [\text{Re}(r_j) \, \text{Im}(r_j) \, \text{Re}(h_j) \, \text{Im}(h_j)]^T$ is served as the input of this network. Accordingly, as can be seen from the figure, the input layer of the network is defined by 4 nodes. For the τ-th hidden layer, the output vector $\mathbf{x}_\tau$ can be specified by the following equation [16]

$$\mathbf{x}_\tau = f\left(\mathbf{W}_\tau \mathbf{x}_{\tau-1} + \mathbf{b}_\tau\right), \ \tau = 1, \ \ldots, \ v_d. \tag{4}$$

The function $f(\cdot)$ represents the activation function of neuron. The matrix $\mathbf{W}_\tau$ and the vector $\mathbf{b_i}$ represent weights and biases involved in the τ-th hidden layer, respectively. The output vector of the last hidden layer, $\mathbf{x}_{v_d}$, is finally fed into a linear output layer, comprising of 12 nodes, which can be written as

$$[\ell_i, \ \ldots, \ \ell_{i+11}]^{\mathrm{T}} = \left(\mathbf{W}_\ell \mathbf{x}_{v_d} + \mathbf{b}_\ell\right), \tag{5}$$

where weights and biases for the output layer are denoted by $\mathbf{W}_\ell$ and $\mathbf{b}_\ell$, respectively. Above equation produces 12 LLRs associated with 12 bits of 4K-QAM symbol. This architecture is known in the literature as feed forward fully connected deep neural network [22].

The process of determining the appropriate weights and biases for the underlying neural network architecture is known as the training process. This process is briefly outlined as follows. By repeatedly transmitting 4K-QAM signals as defined in (1), a large dataset is collected. The dataset consists of a significant number of input-output pairs, where each input (feature) includes the received 4K-QAM symbols and Rayleigh fading coefficients, while the output (response or ground truth) consists of the corresponding 12 LLRs. Hence, each sample in the dataset comprises 4 features and 12 responses. Note that each received 4K-QAM symbol and its associated channel fading is represented as a single vector, as described for $\mathbf{x}_0$. Next, a specific loss function is defined to compute the difference between the actual LLRs and the predicted LLRs. Before training, the weights and biases are randomly initialized. An iterative gradient-based learning algorithm is then applied to minimize the predefined loss function and the weights and biases are updated during each iteration, also known as epoch. The learning process terminates when a stopping criterion is met. If the training process is successful, meaning the training loss is sufficiently small, the neural network is equipped with the appropriate weights and biases.

Despite our efforts to adjust the network architecture or increasing the size of dataset, neural network cannot learn the complex relationship of the conventional demapper. Therefore, simply inputting the dataset into the network does not yield satisfactory results. We present our approach to solve this problem in the following section.

4. Proposed training scheme for neural network based 4K-QAM soft demapper

Through several trials, it is observed that both the log-MAP and max-log-MAP demappers provide the coded 4K-QAM system with identical BER performance. However, the LLR magnitudes from the log-MAP demapper are significantly higher than those from the max-log-MAP demapper. Moreover, the range of each LLR bit produced by the log-MAP demapper varies considerably. For ease of training, the dataset is collected solely from the max-log-MAP demapper. A dataset containing up to 500,000 samples is then used in the training process to develop the neural network-based 4K-QAM soft demapper. However, the neural network still cannot learn to imitate this conventional demapper.

It is important to note that in the Gray-coded rectangular 4K-QAM constellation, the LLR of the 1-st bit is paired with the LLR of the 7-th bit, and so on. Note also that the low magnitude of LLR represents high level of ambiguity. Due to the decision regions of this very dense constellation, the level of LLR ambiguity increases in ascending order from the 1-st bit LLR to the 6-th bit LLR. Figure 3 partially illustrates the ambiguity comparison between the LLRs of the 1-st bit and the 6-th bit. This figure also confirms about the pair between two bits. This relationship suggests that the high level of ambiguity may cause issues during training. Careful examination of the training results confirms that the problem primarily occurs with the LLR of bit 5, 6, 11, and 12. To clearly show this, the sign agreement between the ground truth LLR and the predicted LLR for the s-th bit is introduced:

$$a(s) = 1 - \frac{1}{n_b} \sum_{g=1}^{n_b} \left| \text{sgn}\left(\ell_g(s)\right) - \text{sgn}\left(\hat{\ell}_g(s)\right) \right|, \ s = 1, \ldots, 12. \tag{6}$$

The total number of LLR samples from the dataset involved in the i-th LLR bit is represented by $n_b = n_d/12$. The g-th ground truth LLR for the s-th bit is $\ell_g(s)$ and $\hat{\ell}_g(s)$ is its predicted counterpart. The function $\text{sgn}(\cdot)$ denotes the sign function. With this setup, $a(s)$ represents the accuracy of sign prediction for the s-th bit and the maximum value is 1.00 (100%). As expected, the sign agreement for bits 5, 6, 11, and 12 lies in the low accuracy range of $(0.56, 0.73)$ whereas the rest of the bits show higher agreement in the range of $(0.82, 0.93)$. Thus, a solution is needed to address the low accuracy in some specific bits.
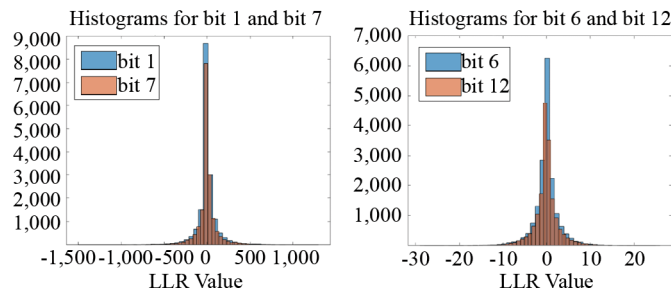

Figure 3. Histogram for the LLR. The left is for the 1-st bit and the right is for the 6-th bit. It can be seen that range of LLR for the 6-th bit is much more narrower than that of the the 1-st bit

To address the aforementioned issue, we attempt to expand the narrow range of low accuracy bits using min-max scaling. This scaling approach, commonly known as feature normalization, is applied to the s-th bit LLR as follows:

$$\ell'_g(s) = \frac{\ell_g(s) - \ell_{\min}(s)}{\ell_{\max}(s) - \ell_{\min}(s)} \cdot \left(r_{\max}(s) - r_{\min}(s)\right) + r_{\min}(s). \tag{7}$$

The scaled LLR is denoted by $\ell'_g(s)$. The original maximum and minimum LLR values are $\ell_{\max}(s)$ and $\ell_{\min}(s)$, respectively, while the new (desired) maximum and minimum LLRs are $r_{\max}(s)$ and $r_{\min}(s)$, respectively. This scaling is applied only to low accuracy bits 5, 6, 11, and 12. Based on experimental results, the appropriate value for $r_{\max}(s)$ is five times the original, and this is also true for the minimum. With the modified dataset utilizing this partial min-max scaling, Table 1 clearly demonstrates that this approach significantly enhances the training process.

Table 1. Accuracy per bit before and after min-max scaling

accuracy/s	1	2	3	4	5	6
$a(s)$	0.93	0.93	0.89	0.82	0.73	0.58
$a_{\text{scaled}}(s)$	0.96	0.95	0.94	0.94	0.93	0.90
accuracy/s	7	8	9	10	11	12
$a(s)$	0.93	0.92	0.88	0.83	0.71	0.56
$a_{\text{scaled}}(s)$	0.95	0.95	0.95	0.94	0.93	0.90

The effectiveness of min-max normalization lies in its ability to handle bits with high ambiguity, where the range of LLR is narrow. In such cases, it is easier for the neural network to predict incorrectly, both in magnitude and sign. Since the sign of LLR is critical, this min-max normalization mitigates this issue by aligning the scale of inputs, thereby improving prediction accuracy.

Before going to show the result of neural network based soft demapper, Table 2 that summarizes the differences among the three types of demappers : log-MAP, max-log-MAP, and the neural network approach is conducted. This table provides a clearer comparison of the methods and highlights their respective characteristics.

Table 2. Comparison of log-MAP, max-log-MAP, and neural network approach

Aspect	log-MAP	max-log-MAP	Neural network-based demapper
Optimality	Optimal	Suboptimal	Suboptimal
Complexity	Unacceptable	Acceptable	Acceptable
Scalability	No	More scalable	Highly scalable
Training/Data requirement	No	No	Yes
Training/Data requirement	No	No	Yes

5. Results and discussion

The performance of the LDPC coded 4K-QAM system employing the proposed neural network-based 4K-QAM demapper is presented in this section. The structure of the LDPC codes is strictly based on the Wi-Fi 7 standard. The min-sum algorithm with 100 decoding iterations is employed as the LDPC decoder. The dataset is collected from a single operating point, specifically the E_b/N_0 at which the BER is 10^{-6}. Therefore, only one neural network model is built and used across the entire testing range. The parameters and relevant setting for training are outlined as follows. Stochastic Gradient Descent with momentum (SGDM) is selected as the learning algorithm, with mean square error (MSE) defined as the loss function. The number of epochs and batch size are tuned to maximize accuracy. Additionally, well-known techniques, include batch normalization and dropout, are employed to improve performance. Table 3 summarizes the parameter used in constructing neural network. While the dataset size can be increased, training time remains a significant limitation. During training, saturation may occur in some trials, however, we pay our attention only on the BER performance.

Table 3. Parameters for neural network

Name	Value
Optimizer	SGDM
Loss	MSE
Number of epochs	2,000
Batch size	1,024
Learning rate	0.001
Number of hidden layer	6
Neurons per hidden layer	128
Dataset size	500,000

Figure 4 shows the performance of an $(n = 648, k = 324)$ LDPC-coded 4K-QAM system with three different demappers. To ensure lower computational complexity, which is the primary goal of this work, the neural network architecture is chosen to achieve approximately a 20% reduction in complexity compared to the conventional demapper.

Specifically, this deep neural network consists of six hidden layers, where each layer uses 128 neurons. To show the merit of this proposed demapper, the complexity analysis in terms of floating point operations (FLOPs) is presented in Table 4. The parameters n_i and n_o represent the number of nodes in the input and output layers, respectively. The FLOPs required for the log-MAP, max-log-MAP, and neural network demappers are approximately 30 MFLOPs, 0.25 MFLOPs, and 0.2 MFLOPs (20% reduction), respectively. It is anticipated that the structure of this neural network, with significantly lower complexity, can be further explored.

Table 4. Computational complexity of the proposed neural network based 4K-QAM demapper

	Operation	Input-to-hidden	Hidden-to-hidden	Hidden-to-output
Input normalization	$\mathbf{z} = \dfrac{\mathbf{x} - \mu}{\sigma}$	$2n_i$	-	-
Linear transformation	$\mathbf{Ax} + \mathbf{b}$	$2n_i v_1$	$2v_\tau v_{\tau+1}$	$2v_d n_o$
Activation function	$\max(0.01y, \, y)$	v_1	v_τ	-
Batch normalization	$\gamma\left(\dfrac{\mathbf{y} - \mu}{\sigma}\right) + \beta$	-	$4v_\tau$	-
Output descaling	$\dfrac{(x - min')(max - min)}{max' - min'} + min$	-	-	$3n_o$

In our analysis, the computational complexity in terms of FLOPs is evaluated. It is important to note that additions (subtractions), multiplications (divisions), exponentiations, and logarithms are inherently included in FLOPs. This approach provides a unified metric for comparing the computational workload of various methods. The advantage of using FLOPs as a metric is that it offers a holistic measure of computational complexity. It is known in literature that FLOPs is a machine-independent computational complexity measure. Moreover, a number of FLOPs are reasonable to represent how fast a method could be executed. Therefore, it is not surprising to see that this metric has been adopted in many researches in order to demonstrate computational complexity analysis [23–25].

Figure 4 clearly shows that the proposed neural network-based 4K-QAM demapper with accuracy 94% provides excellent system performance. When compared to both the optimal log-MAP demapper and the sub-optimal max-log-MAP demapper, our approach exhibits negligible coding loss. This indicates that the proposed demapper can effectively imitate the functionality of the conventional demappers, not only at the data collection point but across the entire E_b/N_o range. It is also shown in this figure that the proposed demapper with lower complexity achieve lower accuracy and cannot achieve the benchmark performance. It can also be observed that both conventional demappers provide the LDPC-coded 4K-QAM system with identical performance. This confirms the validity of selecting the max-log-MAP demapper as the benchmark and justifies collecting the dataset from this sub-optimal demapper rather than the optimal one.
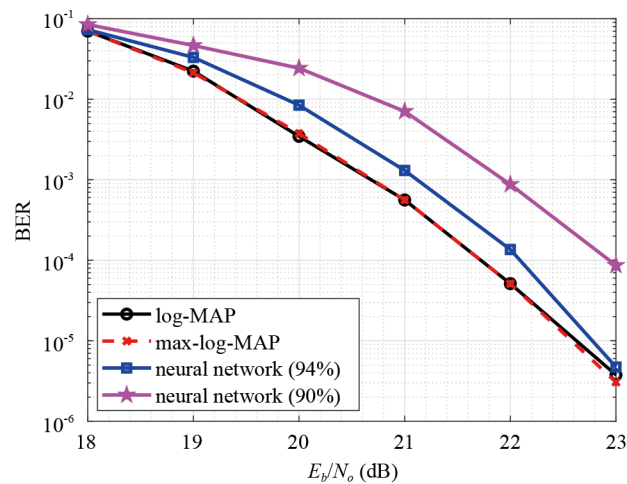
Figure 4. The BER performance of the $(n = 648, k = 324)$ LDPC coded 4K-QAM system. The coding rate is $R = 1/2$ and one LDPC codeword is equivalent to $p = 54$ 4K-QAM symbols. Three different demappers, which are log-MAP, max-log-MAP and neural network, are chosen for comparison purpose. For neural network, the average accuracy is also shown in parentheses. The proposed neural network based 4K-QAM demapper is built from dataset gathered on $E_b/N_o = 23$ dB

The performance of the proposed neural network-based soft demapper is dependent on the accuracy of the neural network model. For a neural network with 94% accuracy, the performance is comparable to the benchmark (Max Log-MAP), as the coding loss in the high SNR region is within 0.4 dB. This demonstrates that the proposed method can achieve similar BER performance to the benchmark while reducing computational complexity by approximately 20%. However, for a neural network with 90% accuracy, the goal is to illustrate the trade-off between complexity and performance. While the BER performance decreases significantly due to reduced accuracy, the significant reduction in complexity makes this approach suitable for applications where computational efficiency is a priority.

With similar motivation and designing strategy, we further demonstrate that the proposed neural network based 4K-QAM demapper can also exhibit the great performance in different coded 4K-QAM system scenarios. The performance of $(n = 1296, k = 864)$ LDPC coded 4K-QAM system with our neural network approach and max-log-MAP demapper is depicted in Figuer 5. As expected, the proposed neural network based 4K-QAM demapper can still imitate the function of max-log-MAP demapper even when the codeword size and coding rate are changed.
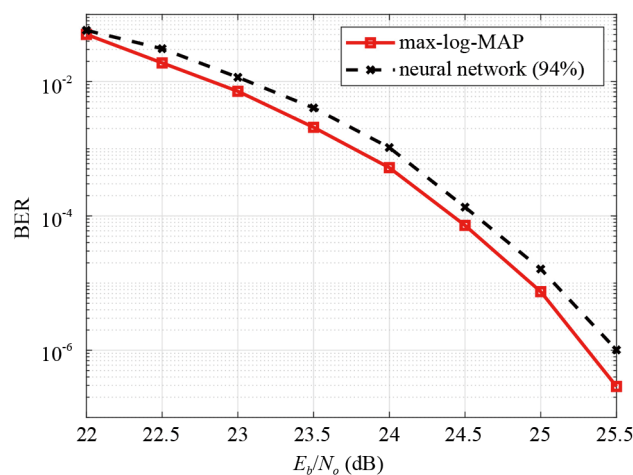

Figure 5. The BER performance of the $(n = 1, 296, = 864)$ LDPC coded 4K-QAM system. Each codeword transmission can be represented by $p = 108$ 4K-QAM symbols. The coding rate is $R = 2/3$. The proposed neural network based 4K-QAM demapper is built from dataset gathered on $E_b/N_o = 25$ dB

6. Conclusion

The 4K-QAM modulation scheme is a promising trend in high-speed communication systems due to its ability to provide superior spectral efficiency over scarce and expensive spectrum resources. However, the excessive computational complexity associated with its soft demapper has been considered as a significant obstacle for its widespread deployment. In this paper, we introduce a neural network based 4K-QAM soft demapper as an alternative to conventional methods. To enable efficient training, we discovered the necessity of employing min-max scaling for output normalization. By significantly reducing receiver complexity, our proposed neural network enables the efficient deployment of high-speed 4K-QAM communication systems. This capability is important for supporting real-time applications and interactive multimedia services in modern communication networks. It is particularly relevant for technologies such as Wi-Fi 7, Wi-Fi 8, DVB-C2, as well as power line communication (PLC) systems, where high-order modulation schemes like 4K-QAM are considered promising solutions. Furthermore, the overall framework encompassing data preparation, pre-processing, and model training can be adapted as a comprehensive methodology for developing neural network-based soft demapper not only for 4K-QAM systems but also for higher-order modulation schemes. We will explore the application of neural network to 16K-QAM as the future work. This work focuses on the performance of the neural network-based soft demapper in a SISO channel, limiting its applicability to MIMO channels, which are more common in practical systems. Additionally, the approach faces constraints related to the large dataset requirements and long training times.

Conflict of interest

The authors declare that they have no conflicts of interest related to this study.

References

[1] Xie Q, Wang Y, Ding J, Niu J. Light convolutional neural network for digital predistortion of radio frequency power amplifiers. *IEEE Communications Letters*. 2024; 28(10): 2377-2381.

[2] Mourya S, Reddy P, Amuru S, Kuchi KK. Spectral temporal graph neural network for massive MIMO CSI prediction. *IEEE Wireless Communications Letters*. 2024; 13(5): 1399-1403.

[3] Jameel ASMM, Malhotra A, Gamal AE, Hamidi-Rad S. Deep OFDM channel estimation: Capturing frequency recurrence. *IEEE Communications Letters*. 2024; 28(3): 562-566.

[4] Singya PK, Shaik P, Kumar N, Bhatia V, Alouini MS. A survey on higher-order QAM constellations: Technical challenges, recent advances, and future trends. *IEEE Open Journal of the Communications Society*. 2021; 2: 617-655.

[5] Puzko D, Batov Y, Gelgor A, Dolgikh D. QAM constellations with fractional entropy to gain in margin maximization for frequency selective channels. *IEEE Communications Letters*. 2023; 27(5): 1457-1461.

[6] Yuan M, Tang M, Li J, Wang H, Yu Y. A novel reconfigurable mapping of physical-layer network coding with M-QAM. *IEEE Communications Letters*. 2024; 28(1): 4-8.

[7] Hoefel RPF. Effects of phase noise and frequency offset on the performance of 4K-QAM and 16K-QAM in 802.11be and 802.11bn WLANs. In: *2024 IEEE Wireless Communications and Networking Conference (WCNC)*. IEEE; 2024. p.1-6.

[8] Wang M, Zhao X, Wang C, Yu J. SNR improved digital-PCM radio-over-fiber scheme supporting 65536 QAM for mobile fronthaul. *IEEE Photonics Technology Letters*. 2023; 35(15): 825-828.

[9]   Choi S, Ahn M, Jeong J. Efficient hardware implementation of soft demapper for WiFi7 4096-QAM. In: *2023 IEEE 98th Vehicular Technology Conference (VTC2023-Fall)*. IEEE; 2023. p.1-6.

[10] Chen Q, He J, Deng R, Chen M, Chen L. FFT-size efficient 4096-QAM OFDM for low-cost DML-based IMDD system. *IEEE Photonics Journal*. 2016; 8(5): 1-10.

[11] Shi J, Yu J, Zhang J, Zhu M, Zhang L, Liu J, et al. 4096-QAM OFDM THz-over-fiber MIMO transmission using delta-sigma modulation. *IEEE Photonics Technology Letters*. 2023; 35(13): 741-744.

[12] Mvone RE, Hannachi C, Hammou D, Moldovan E, Tatu SO. Optimization of 16-QAM and 32-QAM constellations for mitigating impairments of phase noise in millimeter-wave receivers. *IEEE Transactions on Wireless Communications*. 2022; 21(6): 3605-3616.

[13] He G, Sarkis G, Hemati S, Gross WJ, Bai B. Low-complexity channel-likelihood estimation for non-binary codes and QAM. *IEEE Communications Letters*. 2012; 16(6): 801-804.

[14] Xu C, Liang D, Sugiura S, Ng SX, Hanzo L. Reduced-Complexity Approx-Log-MAP and Max-Log-MAP soft PSK/QAM detection algorithms. *IEEE Transactions on Communications*. 2013; 61(4): 1415-1425.

[15] Wang Q, Xie Q, Wang Z, Chen S, Hanzo L. A universal low-complexity symbol-to-bit soft demapper. *IEEE Transactions on Vehicular Technology*. 2014; 63(1): 119-130.

[16] Shental O, Hoydis J. "Machine LLRning": Learning to softly demodulate. In: *2019 IEEE Globecom Workshops (GC Wkshps)*. IEEE; 2019. p.1-7.

[17] Toledo RN, Akamine C, Jerji F, Silva LA. M-QAM Demodulation based on machine learning. In: *2020 IEEE International Symposium on Broadband Multimedia Systems and Broadcasting (BMSB)*. IEEE; 2020. p.1-6.

[18] Vankayala SK, Kumar S, Shenoy KG, Thirumulanathan D, Yoon S, Kommineni I. Neural network architecture for LLR computation in 5G systems and related business aspects. In: *2021 24th International Symposium on Wireless Personal Multimedia Communications (WPMC)*. IEEE; 2021. p.1-6.

[19] Zhang Y, Pan Y, Tian Q, Yang Y, Zhang X, Zhuang Y, et al. Low complexity log likelihood ratio estimation algorithm based on neural network in decoding. In: *2023 6th International Conference on Electronics Technology (ICET)*. IEEE; 2023. p.652-657.

[20] Yeşil S. Log-likelihood ratio calculation for high-order digital modulations using artificial neural networks. In: *2022 30th Signal Processing and Communications Applications Conference (SIU)*. IEEE; 2022. p.1-4.

[21] Robertson P, Villebrun E, Hoeher P. A comparison of optimal and sub-optimal MAP decoding algorithms operating in the log domain. In: *Proceedings IEEE International Conference on Communications ICC'95*. IEEE; 1995. p.1009-1013.

[22] Nikbakht R, Jonsson A, Lozano A. Unsupervised learning for cellular power control. *IEEE Communications Letters*. 2021; 25(3): 682-686.

[23] Zargar B, Ponci F, Monti A. Evaluation of computational complexity for distribution systems state estimation. *IEEE Transactions on Instrumentation and Measurement*. 2023; 72: 1-12.

[24] Sinha A, Bryngelson SH. Neural networks can be flop-efficient integrators of 1D oscillatory integrands. *arXiv:240405938*. 2024. Available from: https://arxiv.org/abs/2404.05938.

[25] Cheng H, Zhang M, Shi JQ. A survey on deep neural network pruning: Taxonomy, comparison, analysis, and recommendations. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. 2024; 46(12): 10558-10578.