A Transformer-Based End-to-End Receiver Design for Wi-Fi 7 Physical Layer

LIU Yichen, GAO Ruixin, ZENG Chen, LIU Yingzhuang

(School of Electronic Information and Communications, Huazhong University of Science and Technology, Wuhan 430074, China)

DOI: 10.12142/ZTECOM.202504005

https://kns.cnki.net/kcms/detail/34.1294.TN.20251201.0957.002.html, published online December 1, 2025

Manuscript received: 2025-09-16

Abstract: The increasing demand for high throughput and low latency in Wi-Fi 7 necessitates a robust receiver design. Traditional receiver architectures, which rely on a cascade of complex, independent signal processing modules, often face performance bottlenecks. Rather than focusing on semantic-level tasks or simplified Additive White Gaussian Noise (AWGN) channels, this paper investigates a bit-level end-to-end receiver for a practical Wi-Fi 7 Multiple-Input Multiple-Output Orthogonal Frequency Division Multiplexing (MIMO-OFDM) physical layer. A lightweight Transformer-based encoder-only architecture is proposed to directly map synchronized OFDM signals to decoded bitstreams, replacing the conventional channel estimation, equalization, and data detection. By leveraging the multi-head self-attention mechanism of the Transformer encoder, our model effectively captures long-range spatial–temporal dependencies across antennas and subcarriers, thus learning to compensate for channel distortions without explicit channel state information. This mechanism eliminates the need for explicit channel estimation, enabling the direct extraction of crucial channel and signal features. Experimental results validate the efficacy of the proposed design, demonstrating the significant potential of deep learning for future wireless receiver architectures.

Keywords: Transformer; receiver design; Wi-Fi 7; deep learning

Citation (Format 1): LIU Y C, GAO R X, ZENG C, et al. A transformer-based end-to-end receiver design for Wi-Fi 7 physical layer [J]. ZTE Communications, 2025, 23(4): 27 – 36. DOI: 10.12142/ZTECOM.202504005

Citation (Format 2): Y. C. Liu, R. X. Gao, C. Zeng, et al., "A transformer-based end-to-end receiver design for Wi-Fi 7 physical layer," *ZTE Communications*, vol. 23, no. 4, pp. 27 – 36, Dec. 2025. doi: 10.12142/ZTECOM.202504005.

1 Introduction

The proliferation of data-intensive applications, ranging from immersive virtual reality and high-definition streaming to industrial automation and the Internet of Things (IoT), has continuously driven the evolution of wireless communication standards. As the next generation of Wi-Fi technology, the IEEE 802.11be standard, commercially known as Wi-Fi 7[1 – 2], is poised to meet these demands by delivering higher throughput, lower latency, and enhanced reliability. Key technological advancements such as 320 MHz channel bandwidth, Multi-Link Operation (MLO) [3], and higher-order 4 096-Quadrature Amplitude Modulation (QAM) are at the core of Wi-Fi 7's performance gains. While these innovations push the theoretical limits of data transmission, they simultaneously introduce significant complexity to the physical layer (PHY) receiver design.

The traditional Wi-Fi PHY receiver architecture operates as a cascaded chain of independent signal processing blocks, including time-frequency synchronization, channel estimation, equalization, and decoding. Each module is meticulously designed and optimized based on expert knowledge and mathematical models of the communication channel. However, this modular, block-by-block approach suffers from two fundamental limitations. First, the performance of each module is highly sensitive to the imperfections of its preceding stages, leading to a "domino effect" where errors propagate and accumulate. Second, the explicit channel estimation module, while crucial, can be computationally intensive and may not always accurately capture the complex, time-varying nature of wireless channels, especially in multi-path and multi-antenna environments. As Wi-Fi 7 leverages multi-antenna technologies, the intricate spatial and temporal correlations across the received signals present a formidable challenge that conventional methods struggle to address holistically.

In recent years, the paradigm of applying artificial intelligence (AI)/deep learning (DL) to wireless communication systems has attracted significant attention as a promising alternative to traditional model-based designs. Early works have dem-

This work was supported by the Huawei Technologies Co., Ltd. under Grant No. TP20250612004.

onstrated the potential of convolutional neural networks (CNNs)[4], deep neural networks (DNNs) and recurrent neural networks (RNNs)[5–7] for signal processing blocks such as channel modeling[8], coding/decoding[9], channel estimation[10] and equalization[11]. DONG et al. applied CNNs to millimeter-wave large-scale multiple-input multiple-output (MIMO) channel estimation[4], effectively utilizing the local correlations of the channel in the space-frequency-time domain. However, CNNs rely on stacked convolution layers to perceive global information, and this inherent local processing mechanism makes it insufficient when directly modeling long-distance, non-local dependencies within the channel matrix. Ref. [5] used RNNs with bidirectional Long Short-Term Memory (LSTM) to recover data directly from the received signal, effectively simplifying the receiver design. However, the model's understanding of the entire Orthogonal Frequency Division Multiplexing (OFDM) symbol is based on sequential information propagation rather than a one-time global perception. Ref. [6] used RNNs to predict MIMO channels and performed offline training using complex hybrid evolutionary algorithms. However, the inherent sequential processing paradigm of RNNs limits their ability to capture long-range correlations in channel time series effectively, and their performance is heavily dependent on specific offline training processes, making it difficult to adapt to dynamic changes in the channel environment. Ref. [7] proposed a receiver framework based on DNNs that predicts future channel coefficients by learning the time-domain correlation of the channel, thereby effectively reducing pilot overhead. However, this method relies on a recurrent structure for sequence prediction, which makes it difficult for the model to capture long-term dependencies in channel changes; as a result, prediction errors accumulate over time.

As mentioned above, the cascaded architecture suffers from the performance limitations, and emphasis has shifted to the DL-based end-to-end (E2E) PHY design by interpreting a communication system as an autoencoder that jointly optimizes the transmitter and receiver over simplified channels[12–14]. Ref. [12] provides the theoretical justification for introducing DL to PHY and incorporates Radio Transformer Networks to embed physical priors, thereby enhancing the model's interpretability and generalization ability. AIT AOUDIA and HOYDIS[13] systematically evaluated E2E learning for OFDM systems under time-frequency selective fading and channel aging scenarios, and demonstrated that learning only the receiver can maintain bit error rate (BER) performance with sparse pilot tones. SONG et al.[14] conducted a systematic benchmark evaluation of E2E autoencoders, re-evaluating the actual gains of autoencoders under more standardized training assumptions and improved baselines. However, they lacked a systematic assessment of complexity, scalability, and training costs. Moreover, these studies mainly consider single-antenna or small-scale systems with Additive White Gaussian Noise (AWGN) or simple fading channels and predominantly rely on fully connected networks or CNNs rather than Transformer architectures.

The Transformer, originally a cornerstone of natural language processing, excels at processing sequential data by employing a multi-head self-attention mechanism[15]. More recently, Transformer architectures have been extensively explored in the context of semantic communications and joint source-channel coding (JSCC). HUANG et al. proposed a JSCC framework for semantic communications of images[16], which combines deep source coding with a hyper-prior model and conventional digital channel block coding, and derived a two-step rate control algorithm that adapts the source-channel rate split to the channel signal-to-noise ratio (SNR). For image transmission, BOURTSOULATZE et al. introduced a deep JSCC scheme[9] that directly maps image pixels to complex channel symbols using a convolutional autoencoder, with the noisy channel implemented as a non-trainable layer in the network. In parallel, Transformer-based decoders have been proposed for algebraic block codes and Low-Density Parity-Check (LDPC) codes[17], showing that self-attention can effectively capture the code structure and improve soft-decoding performance.

In contrast to the above studies, this paper targets the bit-level demodulation in a practical Wi-Fi MIMO-OFDM system and proposes a novel DL-based E2E receiver for the Wi-Fi 7 PHY layer, leveraging the power of the Transformer architecture, which can inherently capture long-range spatial and temporal dependencies across subcarriers and antennas of the synchronized OFDM symbols. By learning these intricate relationships, our proposed E2E model completely bypasses the need for an explicit channel estimation block, directly extracting crucial channel and signal features from the input data. This unified approach not only simplifies the receiver architecture but also offers a more robust and adaptive solution to the complexities of Wi-Fi 7 channels. Our experimental results validate the efficacy of this design. The key contributions of this work are summarized as follows:

1) We propose a receiver-only, encoder-only Transformer architecture for the Wi-Fi 7 MIMO-OFDM system, which operates directly on the synchronized multi-antenna OFDM frequency grid and outputs the coded bitstream, thereby replacing conventional channel estimation and equalization, and bit-detection chain while keeping a standard Wi-Fi 7 transmitter.

2) We design a complexity-aware lightweight head, consisting of a Convolutional Feature Enhancement Module (CFEM) and a compact bitstream recovery layer, which jointly exploit global (Transformer) and local (1D-CNN) features, making the model suitable for resource-constrained receivers.

3) We present a comprehensive experimental validation showcasing the superior performance of the proposed design in terms of BER compared to conventional methods.

The rest of this paper is organized as follows. Section 2 describes the system model and architecture. Section 3 presents the proposed DL-based E2E receiver design. Following that, the algorithm verification and application experiment are described in Section 4. Finally, Section 5 concludes the paper.

2 System Model

We consider a single-user MIMO OFDM communication system, which serves as the equivalent model for our Wi-Fi 7 physical layer simulation. The number of transmit and receive antennas is configured with N_t=4, N_r=4. The traditional modular design of PHY transmission, from the transmitter to the receiver, is depicted in Fig. 1. The following subsections provide detailed descriptions of the key settings for the transmitter, the TGax Non-Line-of-Sight (NLOS) office channel model, and receiver.

2.1 Transmitter Design

At the transmitter, a PHY Service Data Unit (PSDU) stream is generated as the source message, followed by forward error correction coding (FEC), QAM, OFDM waveform generation, and other processes to generate the waveform to be transmitted. Specifically, the original data length of the PSDU is 2 792 bit. A 16 bit service field is then added to the data header, and after concatenation, a 2 808 bit payload is formed, which serves as the basic data unit for PHY transmission. Following this, the data is scrambled using a bitwise Exclusive OR operation with a pseudo-random sequence, and then forward error correction coding is applied using a low-density parity-check code, expanding the original data block into a 3 744 bit codeword. Finally, the data stream is modulated into data symbols using 16-QAM modulation, and OFDM symbol generation is completed through operations such as spatial stream partitioning and subcarrier mapping.

2.2 Channel Model

To simulate complex indoor environments and effectively characterize multipath effects and time-varying fading characteristics, we adopt the TGax NLOS office channel model (Model-D), as defined in the IEEE 802.11ax standard. Based on the Model-D channel, a large-scale shadow fading model characterized by a log-normal distribution is superimposed to simulate the path loss and occlusion attenuation of signals during long-distance transmission. This combined channel model conforms to the signal attenuation characteristics of typical office environments. In addition, to simulate a real noisy environment, AWGN is added to the channel output, and the SNR parameter is adjusted to achieve noise intensity control within the dynamic range of 0 – 34 dB, reproducing the signal transmission characteristics in complex wireless communication scenarios.

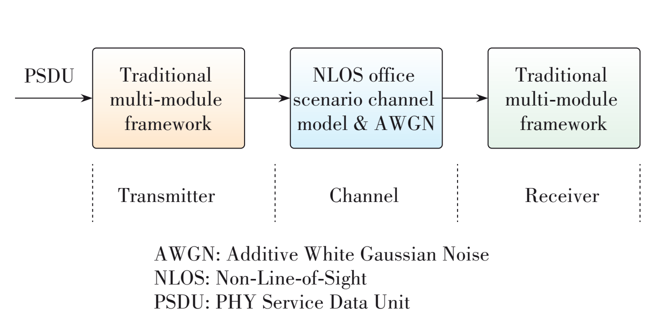

PSDU
Traditional multi-module framework
NLOS office scenario channel model & AWGN
Traditional multi-module framework
Transmitter
Channel
Receiver
AWGN: Additive White Gaussian Noise
NLOS: Non-Line-of-Sight
PSDU: PHY Service Data Unit

Figure 1. A traditional modular design of PHY transmission

For each OFDM subcarrier, the baseband input-output relation can be written as

$$\boldsymbol{y}_k = \boldsymbol{H}_k \boldsymbol{x}_k + \boldsymbol{n}_k \quad (1),$$

where $\boldsymbol{x}_k \in \mathbb{C}^{N_s \times 1}$ is the collection of the N_s spatial data streams ($1 \leqslant N_s \leqslant 4$), $\boldsymbol{y}_k \in \mathbb{C}^{N_r \times 1}$ is the received signal vector, $\boldsymbol{H}_k \in \mathbb{C}^{N_r \times N_s}$ is the effective MIMO channel matrix on subcarrier k, and $\boldsymbol{n}_k \sim \mathcal{CN}\left(0, \sigma_n^2 \boldsymbol{I}\right)$ denotes AWGN. In our main simulations, we set $N_t = N_r = N_s = 4$, so that four spatial streams are transmitted simultaneously.

The MIMO channel $\boldsymbol{H}_k$ is estimated in the traditional baseline receiver using the Wi-Fi 7 preamble structure. Specifically, least-squares (LS) channel estimation is performed based on the orthogonal pilot symbols transmitted in the Extremely High Throughput Long Training Field (EHT-LTF) fields, which enables recovery of the full 4 × 4 MIMO channel matrix across all subcarriers. The estimated channels are then used to construct a singular value decomposition (SVD)-based minimum mean square error (MMSE) equalizer, as detailed in Section 2.3. The spatial multiplexing characteristics of this multi-stream channel lead to a linear superposition of multipath components at the receiver, resulting in spatial correlation that must be handled by the equalizer or, in our proposed approach, by the Transformer-based neural receiver.

2.3 Traditional Receiver Design

The details of the traditional receiver scheme are described in this subsection. The receiver employs a multi-stage signal processing framework that incorporates time-frequency synchronization, channel estimation, and equalization to achieve precise signal recovery. The detailed flowchart of the receiver is shown in Fig. 2.

The initial synchronization phase aims to correct time and frequency deviations. By leveraging the periodic characteristics embedded in the pilot symbols, the system deploys an autocorrelation-based detection mechanism to precisely locate the starting boundary of the OFDM signal, thereby enabling packet detection and coarse timing. Subsequently, the carrier frequency offset (CFO) f_{offset} is precisely estimated by analyzing the phase difference between known periodic structures as follows:

$$\begin{aligned} \Delta\phi &= \frac{\varphi}{2\pi} \\ f_{\text{offset}} &= \Delta\phi \frac{f_s}{T_s} \end{aligned} \quad (2),$$

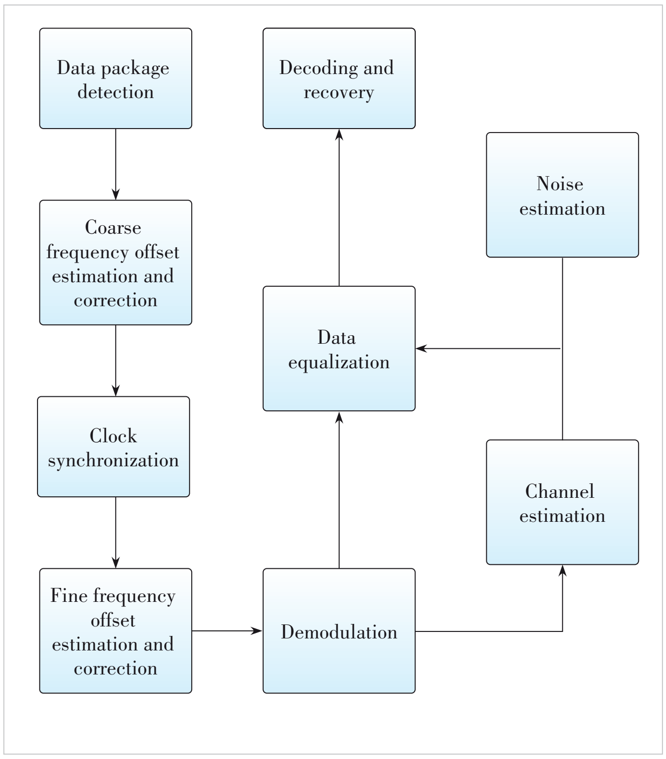


Figure 2. Operation flowchart of the receiver

where φ, $\Delta\phi$, f_s, and T_s are the phase angle, normalized phase difference, sampling rate, and symbol period, respectively. In the next stage, the channel frequency response (CFR) is estimated based on the Least Squares (LS) algorithm as follows:

$$\begin{aligned} &\boldsymbol{Y} = \boldsymbol{HX} + \boldsymbol{N} \\ &\hat{\boldsymbol{H}} = (\boldsymbol{X}^H\boldsymbol{X})^{-1}\boldsymbol{X}^H\boldsymbol{Y} \end{aligned} \qquad (3),$$

where $\boldsymbol{X}$, $\boldsymbol{H}$ and $\boldsymbol{N}$ are the original transmitted signal, CFR, and AWGN, respectively. The average channel response $\overline{\boldsymbol{H}}$ is obtained by performing a time-domain average of the pilot estimate values. Finally, the noise variance σ_n^2 is calculated as follows:

$$\begin{aligned} &\mu = \hat{\boldsymbol{H}} - \overline{\boldsymbol{H}} \\ &\sigma_n^2 = \boldsymbol{I} \cdot E\left[\left|\mu\right|^2\right] \end{aligned} \qquad (4),$$

where $\boldsymbol{I}$ is the identity matrix. Using this estimate as a basis, SVD is first used to decompose $\overline{\boldsymbol{H}}$ into independent parallel subchannels, with each subchannel i corresponding to a singular value S_i:

$$\overline{\boldsymbol{H}} = \boldsymbol{USV}^H \qquad (5),$$

where the columns of $\boldsymbol{U}$ span the received signal space, $\boldsymbol{S} = \mathrm{diag}(S_1,\cdots,S_P)$ contains the singular values corresponding to the P spatial subchannels, and $\boldsymbol{V}^H$ is the orthogonal basis of the transmitted signal space, mapping the equalized signal back to the transmitted signal space. We seek a linear equalizer $\boldsymbol{W}$ such that $\hat{\boldsymbol{x}} = \boldsymbol{Wy}$ minimizes the mean square error $\mathbb{E}\left\{\left\|\boldsymbol{x} - \boldsymbol{Wy}\right\|^2\right\}$. Under the standard assumption $\mathbb{E}\left\{\boldsymbol{xx}^H\right\} = \boldsymbol{I}$ and $\mathbb{E}\left\{\boldsymbol{nn}^H\right\} = \sigma_n^2\boldsymbol{I}$, the MMSE solution is

$$\boldsymbol{W}_{\mathrm{MMSE}} = \boldsymbol{V}(\boldsymbol{S}^H\boldsymbol{S} + \sigma_n^2\boldsymbol{I})^{-1}\boldsymbol{S}^H\boldsymbol{U}^H \qquad (6),$$

which assigns different weights to each singular mode according to its channel gain and the noise variance. Finally, the equalization matrix is applied to the received signal to obtain

$$\hat{\boldsymbol{x}} = \boldsymbol{W}_{\mathrm{MMSE}}\boldsymbol{y} = \sum_{i=1}^{P}\frac{S_i}{S_i^2 + \sigma_n^2}v_i\left(u_i^H\boldsymbol{y}\right) \qquad (7),$$

where S_i is the i-th singular value, and u_i and v_i are the i-th columns of $\boldsymbol{U}$ and $\boldsymbol{V}$, respectively. This expression shows that the MIMO channel is decomposed into P parallel subchannels, each employing an MMSE scalar coefficient for equalization. In the traditional baseline, the subsequent step is to demodulate and decode $\hat{\boldsymbol{x}}$ to reconstruct the original transmitted bitstream. In contrast, our DL receiver replaces this entire LS+MMSE equalization and bit-detection chain with the proposed neural network.

3 Transformer-Based End-to-End Receiver Design

In a single-user 4×4 MIMO baseband transmission scenario, the spatial multiplexing characteristics of the channel cause the received signals to exhibit linear superposition of multipath propagation signals, resulting in spatial correlation. Traditional receiver architectures, such as those based on MMSE or zero-forcing equalizers, rely on the accuracy of channel state information (CSI) obtained through pilots, which introduces unavoidable pilot overhead; on the other hand, in typical Wi-Fi indoor environments, characterized by multipath fading, dynamic interference, shadow effects, and high-intensity noise, accurate CSI estimation is highly challenging, leading to a significant degradation in the performance of linear equalizers. Additionally, the complex scrambling and channel coding modules on the transmitter side, while enhancing link reliability, significantly increase the system's computational complexity and processing latency.

To address the limitations of traditional approaches, we explore the feasibility of modeling physical layer signal processing as an E2E-DL network. Specifically, given the Transformer model's strong global feature extraction capabilities in sequence data modeling, we propose a lightweight receiver based on a Transformer encoder, as shown in Fig. 3. At the transmitter (TX), a random TX PSDU bitstream is processed by a Wi-Fi 7 (IEEE 802.11be EHT) baseband chain and transmitted over a TGax NLOS MIMO-OFDM channel. At the re-

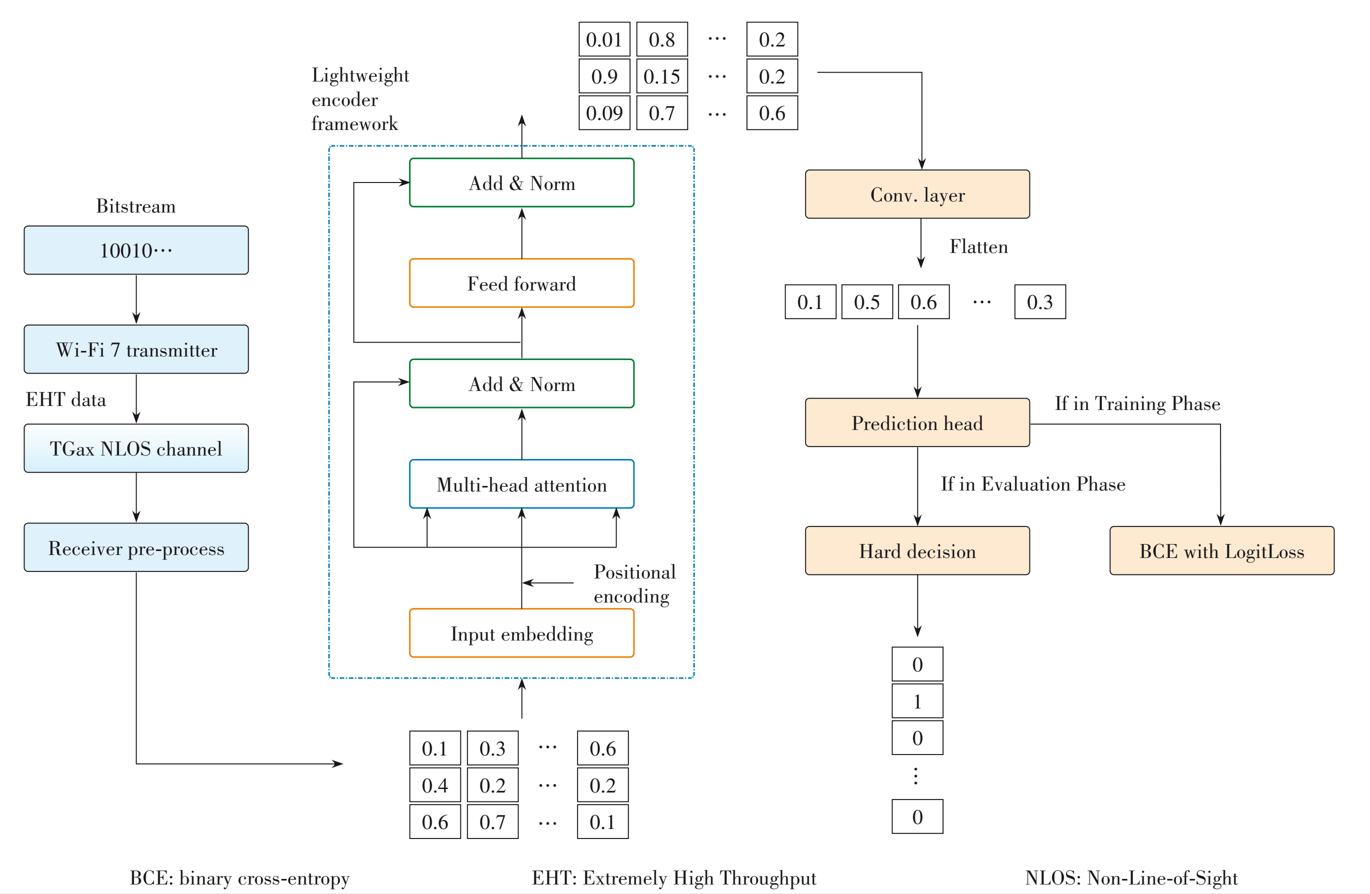


Figure 3. Lightweight encoder framework of the proposed Transformer-based receiver

ceiver side, the conventional front-end performs packet detection, time-frequency synchronization, Fast Fourier Transform (FFT), data demodulation, and phase tracking/correction. The resulting multi-antenna frequency-domain data symbols, after the above pre-processing stage, are used as the input to the proposed neural receiver, while the corresponding TX PSDU serves as the label. Thus, the network is trained to recover the original bitstream directly from the demodulated MIMO-OFDM data, effectively replacing the traditional chain of noise variance estimation, channel estimation and equalization, and bit-level detection.

The proposed neural receiver comprises three main components: an encoder-only Transformer, a CFEM, and a bitstream recovery layer. The Transformer encoder provides a robust representation of the MIMO-OFDM signal by exploiting multi-head self-attention over antenna and time/frequency dimensions. The CFEM further refines these features and reduces the effective sequence length. The final multi-layer perceptron (MLP)-based bitstream recovery layer maps the hierarchical features to bit-wise logits, which are trained with a binary cross-entropy with logits (BCE-with-Logits) loss and converted into hard decisions during evaluation. The following subsections will provide a detailed explanation of the model's network structure and operational principles.

3.1 Lightweight Encoder Framework

The encoder consists of six layers of identical structures stacked together, with each layer containing four self-attention heads, residual connections, layer normalization, and a feed-forward network (FFN). This design focuses on modeling the spatial correlation of signals across antennas, processes MIMO-OFDM signals through multi-layer feature extraction and nonlinear transformations, and provides a highly robust feature representation for subsequent signal detection and demodulation.

3.1.1 Tokenization

The number of data subcarriers in the OFDM system is set to $N_s = 234$. Each antenna on each subcarrier carries one information symbol ($N_i = 1$). After synchronization and OFDM demodulation, the received signal yields a complex signal matrix $\boldsymbol{S}_c \in \mathbb{R}^{N_s \times N_r}$. Since the model supports processing real-valued data, the complex numbers are decomposed into two independent dimensions based on their real and imaginary parts, resulting in the signal tensor $\boldsymbol{S}_r \in \mathbb{R}^{N_s \times N_r \times 2}$. The signal received by each antenna can be regarded as a "token" for the model input. By calculating the correlation weights between different antennas via self-attention, the model can dynamically learn which antenna signals are more important for restoring specific transmitted data.

The corresponding original randomly generated data bitstreams are directly used as supervision labels to construct the "signal feature-original bit" mapping pairs required for E2E training. The number of bits is $N_b = N_s \times N_i \times N_t = 936$. The data in the dataset is input into the model after being reshaped, with a signal dimension of inputs $\mathcal{I} \in \mathbb{R}^{N_d \times N_b \times 512}$, where N_d is the number of signals while also adapting to the original Transformer encoder input dimension $\left[\text{batchsize, sequencelenth}, d_{\text{model}}\right]$. The supervised label dimension is $\mathcal{L} \in \mathbb{R}^{N_d \times N_b}$.

3.1.2 Input Embedding and Positional Encoding

In the embedding module, the encoder employs a linear projection layer to expand the input to 512 dimensions. This enhancement improves the model's ability to represent features, allowing it to learn more complex representations. Based on empirical findings, the value of d_{model} is 512, demonstrating its feasibility and effectiveness in NLP and sequence modeling tasks. Following this, layer normalization is applied to each sample to stabilize the training process and help the model converge more effectively.

Finally, the Leaky Rectified Linear Unit (LeakyReLU) is used as the nonlinear activation function. Unlike the standard ReLU function that has a zero gradient in the negative value region, LeakyReLU maintains a slight positive slope in this interval[18]. This design avoids the "dying ReLU" problem, where neurons fail to update their weights due to continuously outputting negative values and thus lead to permanent failure during training. The concrete formula is as follows:

$$X_{\text{embedding}} = \text{LeakyReLU}\left(\text{LayNorm}\left(W_e \cdot X_{\text{token}} + b_e\right)\right) \qquad (8).$$

In the position encoding module, a cosine-based position encoding function is used to compensate for the Transformer's lack of sensitivity to sequence order. This function generates position vectors corresponding to the antenna index to distinguish the spatial characteristics of different receiving antennas. The position vector, whose dimension is consistent with the embedding dimension, is added element-wise to the signal features. This explicitly embeds the position information into the model so that the model can better understand the spatial characteristics between antennas. The concrete formula is as follows:

$$\begin{aligned} \text{PE}_{(\text{pos},\, 2i)} &= \sin\left(\text{pos}/10000^{2i/d_{\text{model}}}\right) \\ \text{PE}_{(\text{pos},\, 2i+1)} &= \cos\left(\text{pos}/10000^{2i/d_{\text{model}}}\right) \end{aligned} \qquad (9),$$

where $i \in \left[0, (d_{\text{model}} - 1)/2\right]$ represents the n-th element of the position vector.

3.1.3 Multi-Head Attention Mechanism

The multi-head attention mechanism can focus on feature correlations from different angles in the received signal, improving the model's accuracy and robustness. This model uses a 4-head self-attention mechanism to process different feature subspaces in parallel and independently, which helps improve overall computational efficiency. Simultaneously, the model analyzes the input sequence from multiple perspectives to integrate a more comprehensive set of features. Finally, the outputs from each head are concatenated and subjected to linear projection as follows:

$$\begin{aligned} \text{head}_i &= \text{Attention}\left(\boldsymbol{Q}\boldsymbol{W}_i^Q, \boldsymbol{K}\boldsymbol{W}_i^K, \boldsymbol{V}\boldsymbol{W}_i^V\right) \\ \text{MultiHead}\left(\boldsymbol{Q}, \boldsymbol{K}, \boldsymbol{V}\right) &= \text{Concat}\left(\text{head}_1, \ldots, \text{head}_4\right)\boldsymbol{W}^O \end{aligned} \qquad (10).$$

3.2 Convolutional Feature Enhancement Module

After the Transformer encoder completes the sequence modeling of the received signal, we design a CFEM further to extract the local spatio-temporal features of the signal. This hierarchical feature extraction strategy, which combines global and local information, significantly improves the accuracy of signal recovery.

The architecture of this module is as follows: first, a one-dimensional CNN (1D-CNN) processes the output features of the Transformer. The convolutional kernels in this layer map the input channel count from 4 to 16, with a stride of 4, thereby compressing the sequence length from 512 to 128. This operation not only expands the feature dimension but also serves as an effective downsampling mechanism to reduce computational complexity. Following the convolutional layer are two components: a BatchNorm layer to accelerate convergence and prevent overfitting, and a LeakyReLU activation function to introduce nonlinearity. Additionally, to ensure the integrity of the information flow and optimize the training process, we introduce a skip connection that adds the module's input to the output after convolution, normalization, and activation on an element-wise basis. This residual structure preserves the original global features and provides a shortcut for gradient flow during backpropagation, thereby enhancing the network's trainability.

3.3 Bitstream Recovery Layer

The bitstream prediction head serves as the terminal of the network, with its core task being to decode from hierarchical feature representations into the final bitstream. This module is implemented as a MLP, with the specific process as follows:

1) Feature integration and transformation

First, the feature tensor (16×128) output by the upstream convolutional module is flattened into a single 2 048-dimensional vector, effectively integrating all the high-level features extracted by the network across antennas and time-frequency domains. Next, the first fully-connected layer (FC, also known as the feature transformation layer) linearly maps this vector to

3 744 dimensions, aligning its dimension with the number of bits (N_b) in a single prediction.

2) Normalization and nonlinear activation

Before the nonlinear transformation, layer normalization is applied to normalize the distribution of activations, thereby accelerating convergence and improving training stability. We then use the LeakyReLU activation function. This choice is consistent with the selection for input embedding: LeakyReLU preserves a small gradient for negative inputs, enabling better handling and propagation of negative feature components related to signal phase. For phase-sensitive modulation schemes like 16-QAM, this property helps minimize information loss.

3) Regularization and output projection

The activated features pass through a Dropout layer with a dropout probability of 0.4, reducing the risk of overfitting in the FC layer. The second FC layer (output projection layer) then performs a final linear transformation of these features to refine the prediction results.

During the training phase, LogitsLoss is used to directly optimize the log odds output, avoiding the Sigmoid saturation problem. The loss expression is as follows:

$$L = \frac{1}{N}\left[-b_i \log\sigma(\hat{b}_i) - (1 - b_i)\log(1 - \sigma(\hat{b}_i))\right] \tag{11}.$$

During the testing phase, bitstreams $\hat{b} \in \{0, 1\}^{3744}$ are generated through hard decisions with a threshold of 0.5 to generate the final bits as follows:

$$\hat{b} = \begin{cases} 1, & f_{\mathrm{fc}}(x)_i \geqslant 0.5 \\ 0, & \text{otherwise} \end{cases} \tag{12},$$

where $f_{\mathrm{fc}}(x)_i$ is the FC layer function. Finally, the output results are compared with the label data bit by bit to count the number of errors, and the bit accuracy is calculated based on the total number of error bits as:

$$\mathrm{BitAc} = 1 - \frac{\sum_{i=1}^{N_{\mathrm{test}}}\left|\hat{b}_i - b_i\right|}{N_b \times N_{\mathrm{test}}} \tag{13}.$$

3.4 Complexity Analysis

The per-layer computational complexity of a standard multi-head self-attention block with sequence length L_{seq}, model dimension d_{model}, and H heads (each of size $d_{\mathrm{head}} = d_{\mathrm{model}}/H$) scales approximately as:

$$\mathcal{O}\left(L_{\mathrm{seq}}^2 d_{\mathrm{model}}\right) + \mathcal{O}\left(L_{\mathrm{seq}} d_{\mathrm{model}}^2\right) \tag{14},$$

where the first term corresponds to attention score computation ($\boldsymbol{QK}^{\mathrm{T}}$ and softmax-weighted $\boldsymbol{V}$), and the second term corresponds to the FFN. In our receiver, the effective sequence length is $L_{\mathrm{seq}} = N_r \times N_{\mathrm{sc}}$, e.g., four receive antennas and 234 active subcarriers give $L_{\mathrm{seq}} = 936$. We note that the bit-length $N_{\mathrm{bits}} = 3\,744$ is the output dimensionality of the final fully-connected head and does not directly enter the attention complexity. Moreover, to further reduce inference latency and memory footprint, we append CFEM that downsamples the sequence dimension by a stride s, and projects $d_{\mathrm{model}} \rightarrow d_{\mathrm{red}}$. We choose $d_{\mathrm{red}} \in \left[\frac{1}{8} d_{\mathrm{model}}, \frac{1}{2} d_{\mathrm{model}}\right]$ and set $d_{\mathrm{red}} = 128$ by default ($d_{\mathrm{model}} = 512$), which is approximately one head width when $H = 4$ ($d_{\mathrm{head}} = 128$). This choice preserves the information aggregated by the encoder and makes the final dense bit head operate on a compressed representation, rather than on the full $L_{\mathrm{seq}} \times d_{\mathrm{model}}$ tensor. In practice, this reduces the size of the final linear layer and lowers on-chip activation bandwidth.

4 Experiment

This section evaluates the performance of the proposed DL Transformer receiver against a traditional physical layer baseline via numerical simulation.

4.1 Experiment Settings

4.1.1 Baseline Scheme

The baseline was evaluated using the Monte Carlo method. We measured its BER performance over an SNR range of 0 to 34 dB, in 2 dB increments. To ensure statistical validity, each SNR point was assessed by transmitting a sufficient number of packets over independently realized random channels. In cases of packet detection failure, all bits within the packet were considered erroneous.

4.1.2 DL Transformer Receiver

1) Dataset generation: The proposed receiver was trained and evaluated on a large-scale simulated dataset. The data were generated using the same transmission link as the baseline, but with the transmitter's scrambling and channel coding modules disabled. To promote model generalization, each data sample corresponds to an independent channel realization. Both large-scale fading and small-scale fading were fully randomized to cover typical NLOS conditions.

2) Dataset configuration and preprocessing: The training and validation sets were generated at a fixed SNR of 30 dB, whereas dedicated test sets were generated for each evaluated SNR point (0 – 34 dB). Concretely, the training set consists of 1 200 mini-batches with 64 samples in each mini-batch, resulting in 76 800 training samples in total. The validation and test sets each contain 120 mini-batches with 64 samples per mini-batch, i.e., 7 680 samples per set, which corresponds to an approximate 10:1:1 ratio for training, validation, and test data. For each sample, we first generate a fresh random bitstream and then reset the channel model with new random seeds and parameters, such that the transmitter-receiver distance, path loss, shadowing, multipath delays, and fading co-

efficients (both large-scale and small-scale) are independently randomized. As a result, every sample is associated with an independent channel realization, covering a representative dynamic range of NLOS channel conditions and producing diverse fading characteristics, path loss levels, and noise realizations. For preprocessing, complex-valued signals were decomposed into their real and imaginary components. Furthermore, we applied data augmentation to the training set by adding zero-mean Gaussian noise ($\sigma = 0.05$) to enhance model robustness.

4.2 Training Strategy and Stabilization Techniques

To ensure stable and efficient training of the proposed neural receiver and to mitigate both underfitting and overfitting, we adopt the following training strategy.

1) Initialization and normalization

Network parameters are initialized according to the type of layer and activation. Layers with Gaussian Error Linear Unit (GELU) activations (e. g., in the Transformer encoder) use Xavier initialization, which balances the variance of forward and backward signals. Layers with LeakyReLU activations (e.g., in the CFEM) use Kaiming initialization, which is tailored to ReLU-type nonlinearities. For LayerNorm and BatchNorm, the scale parameters are initialized to 1 and the biases to 0 so that normalization does not distort the feature distribution at the beginning of training. The self-attention projection matrices in the Transformer are also initialized with Xavier to keep the variance of dot-product attention stable.

2) Optimizer and learning-rate scheduling

We use the AdamW optimizer to decouple weight decay from the gradient update, which provides more controlled regularization than classical Adam. Unless otherwise stated, the hyperparameters are set to weight decay 10^{-4}, $\beta_1 = 0.9$, $\beta_2 = 0.999$, and $\epsilon = 10^{-8}$. A ReduceLROnPlateau scheduler monitors the validation loss and reduces the learning rate by a factor of 0.8 if no improvement is observed for five consecutive epochs. This strategy accelerates convergence in the early stage while allowing finer adjustments near convergence.

3) Gradient clipping

To avoid gradient explosion and overly large parameter updates, we apply gradient-norm clipping with a maximum l_2 norm of 2. This improves the stability of training, especially given the long input sequences and the depth of the encoder.

4) Regularization and stabilization

To enhance robustness and prevent overfitting, we combine several standard regularization techniques. L2 weight decay is applied through AdamW; Dropout and LayerNorm are used within the Transformer encoder and the fully connected layers; and BatchNorm is applied after convolutional layers in the CFEM to stabilize feature statistics across mini-batches.

With this training setup, the network learns to map the demodulated MIMO-OFDM symbols directly to the transmitted bitstream in an end-to-end fashion, enabling joint optimization of signal detection and recovery under realistic Wi-Fi 7 channel conditions.

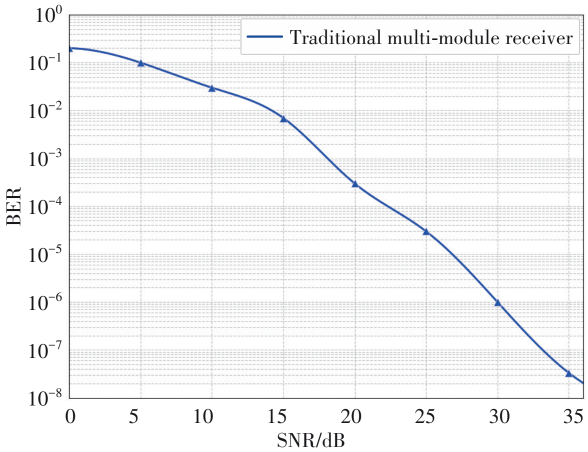
4.3 Experiment Results

The experiment used BER as the core evaluation indicator, defined as

$$\mathrm{BER} = \frac{B_{\mathrm{error}}}{B_{\mathrm{total}}} \tag{15}.$$

Fig. 4 shows the performance curve of the BER of the traditional scheme as a function of SNR. This curve exhibits three distinct regions. In the low SNR region (SNR < 10 dB), system performance is primarily constrained by intense channel noise, resulting in a sharp decline in BER from a high error level. Some errors originate from frequent packet detection failures during this phase. As the SNR enters the medium SNR range (10 dB to 26 dB), the slope of the BER curve slows significantly, indicating that system performance improvements are beginning to be constrained by non-noise factors. In the high SNR region (SNR > 26 dB), the BER decreases rapidly again, and when the SNR reaches 32 dB, the system approaches error-free transmission, achieving highly reliable communication.

Figs. 5a and 5b illustrate the learning dynamics of the proposed model over 1 300 epochs. The model exhibits a convergence pattern. In the initial training phase (approximately the first 200 epochs), the training and validation losses decrease sharply. Correspondingly, accuracy rises rapidly from the random guess baseline of 0.5, indicating that the model is effectively capturing the underlying data features. Following this initial phase, the learning process enters a stable convergence regime. A consistent generalization gap emerges between the training and validation curves, stabilizing in later epochs at approximately 0.05 for the loss and 0.045 for the accuracy.



BER: bit error rate SNR: signal-to-noise ratio

Figure 4. BER performance of the traditional scheme

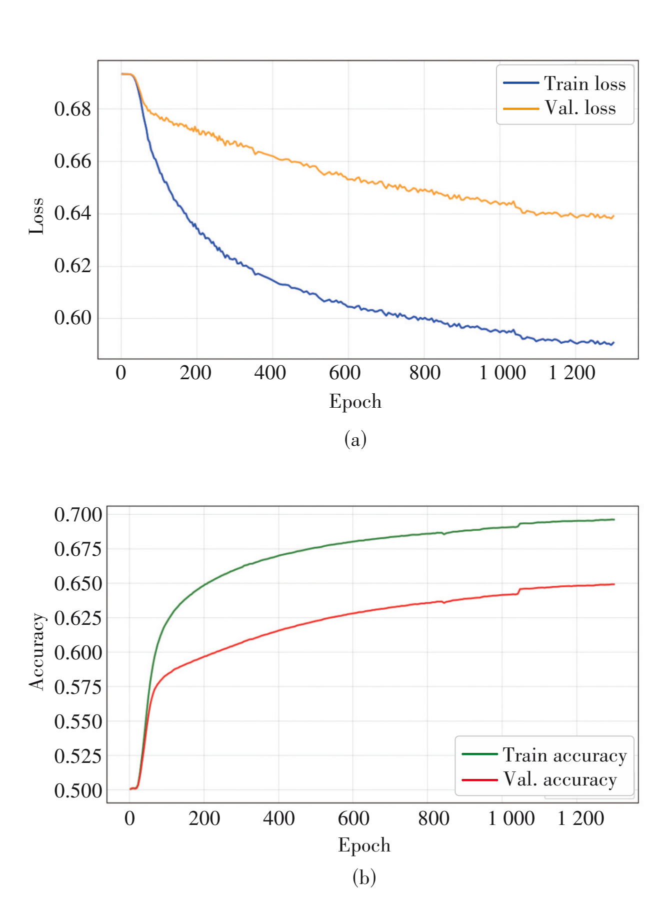
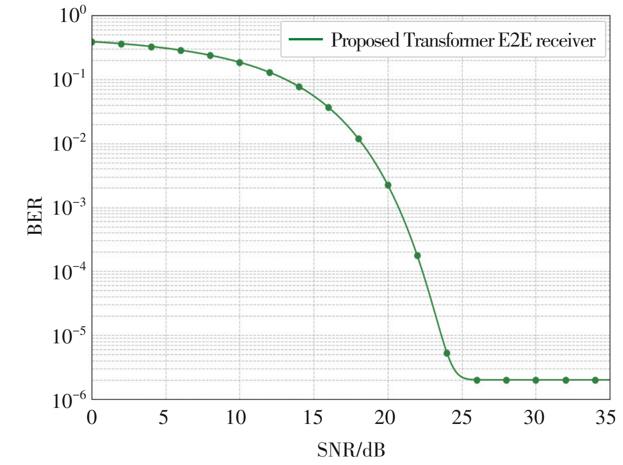


Figure 5. Training and validation curves: (a) loss versus epochs; (b) accuracy versus epochs

Crucially, the validation loss plateaus without a significant upward trend, which suggests that while minor overfitting is present, the model's generalization ability remains well-controlled. The minor fluctuations observed across the curves are attributable to the stochastic nature of minibatch gradient descent when applied to a complex and diverse dataset.

In summary, the leveling off of the validation metrics indicates that the model's performance has likely saturated, given its capacity and the complexity of the dataset. While this confirms the effectiveness of the proposed architecture, it also suggests that its performance ceiling is constrained by either its representational power or the intrinsic noise within the data.

Fig. 6 shows the BER performance of the proposed deep learning receiver, demonstrating that the model achieves excellent signal recovery capability. As shown in the figure, the BER begins to decline when the SNR reaches approximately 10 dB, demonstrating that the model has successfully learned effective features capable of robustly countering channel noise and fading. When the SNR exceeds 23 dB, the decline rate of the BER curve significantly slows down, exhibiting a convergent trend, indicating that the model has reached its performance limit. Overall, these results demonstrate that the proposed deep learning model can successfully learn the E2E signal recovery task without explicit channel estimation and equalization modules. The model exhibits strong robustness across the entire evaluated SNR range, with its achieved low BER meeting the performance standards for high-reliability communication.



BER: bit error rate E2E: end-to-end SNR: signal-to-noise ratio

Figure 6. BER performance of the proposed scheme

5 Conclusions

This paper proposes and validates a novel DL-E2E receiver based on the Transformer architecture, designed to overcome the limitations of traditional multi-module designs in Wi-Fi 7 environments. Our work demonstrates that the model's multi-head self-attention mechanism is effective at implicitly learning and jointly performing noise estimation, channel estimation, and equalization, thus bypassing the need for the explicit modular design. The experimental results underscore the superiority of this data-driven paradigm: The proposed receiver achieves an excellent BER performance across a wide range of SNRs, meeting the requirements for high-reliability communication and serving as a compelling proof-of-concept for AI-native wireless systems.

Building on these promising results, future efforts will be directed toward two critical areas. First, we will investigate the practical implementation and deployment of this architecture, focusing on complexity reduction and optimization for real-time processing on hardware platforms. Second, we will pursue research into the theoretical interpretability of the network's learned features, aiming to move beyond a "black-box" understanding and gain deeper insights into how the model makes its decoding decisions.

References

[1] DENG C L, FANG X M, HAN X, et al. IEEE 802.11be Wi-Fi 7: new challenges and opportunities [J]. IEEE communications surveys & tutorials, 2020, 22(4): 2136 – 2166. DOI: 10.1109/COMST.2020.3012715

[2] KHOROV E, LEVITSKY I, AKYILDIZ I F. Current status and directions of IEEE 802.11be, the future Wi-Fi 7 [J]. IEEE access, 2020, 8: 88664 – 88688. DOI:10.1109/ACCESS.2020.2993448

[3] MENG J, ZHAO Q L, WU W M, et al. Enhancing IEEE 802.11ax network performance: an investigation and modeling into multi-user transmission [J]. IEEE transactions on mobile computing, 2025, 24(3): 2151 – 2165. DOI: 10.1109/TMC.2024.3493032

[4] DONG P H, ZHANG H, LI G Y, et al. Deep CNN-based channel estimation for mmWave massive MIMO systems [J]. IEEE journal of selected topics in signal processing, 2019, 13(5): 989 – 1000. DOI: 10.1109/JSTSP.2019.2925975

[5] WANG S Y, YAO R G, TSIFTSIS T A, et al. Signal detection in uplink time-varying OFDM systems using RNN with bidirectional LSTM [J]. IEEE wireless communications letters, 2020, 9(11): 1947 – 1951. DOI: 10.1109/LWC.2020.3009170

[6] POTTER C, VENAYAGAMOORTHY G K, KOSBAR K. RNN based MIMO channel prediction [J]. Signal processing, 2010, 90(2): 440 – 450. DOI: 10.1016/j.sigpro.2009.07.013

[7] MATTU S R, THEAGARAJAN L N, CHOCKALINGAM A. Deep channel prediction: a DNN framework for receiver design in time-varying fading channels [J]. IEEE transactions on vehicular technology, 2022, 71(6): 6439 – 6453. DOI: 10.1109/TVT.2022.3162887

[8] YANG Y, LI Y, ZHANG W X, et al. Generative-adversarial-network-based wireless channel modeling: challenges and opportunities [J]. IEEE communications magazine, 2019, 57(3): 22 – 27. DOI: 10.1109/MCOM.2019.1800635

[9] BOURTSOULATZE E, BURTH KURKA D, GÜNDÜZ D. Deep joint source-channel coding for wireless image transmission [J]. IEEE transactions on cognitive communications and networking, 2019, 5(3): 567 – 579. DOI: 10.1109/TCCN.2019.2919300

[10] HE H T, WEN C K, JIN S, et al. Deep learning-based channel estimation for beamspace mmWave massive MIMO systems [J]. IEEE wireless communications letters, 2018, 7(5): 852 – 855. DOI: 10.1109/LWC.2018.2832128

[11] CHEN J, WANG X B. Learning-based intermittent CSI estimation with adaptive intervals in integrated sensing and communication systems [J]. IEEE journal of selected topics in signal processing, 2024, 18(5): 917 – 932. DOI: 10.1109/JSTSP.2024.3468037

[12] O'SHEA T, HOYDIS J. An introduction to deep learning for the physical layer [J]. IEEE transactions on cognitive communications and networking, 2017, 3(4): 563 – 575. DOI: 10.1109/TCCN.2017.2758370

[13] AIT AOUDIA F, HOYDIS J. End-to-end learning for OFDM: from neural receivers to pilotless communication [J]. IEEE transactions on wireless communications, 2022, 21(2): 1049 – 1063. DOI: 10.1109/TWC.2021.3101364

[14] SONG J X, HÄGER C, SCHRÖDER J, et al. Benchmarking and interpreting end-to-end learning of MIMO and multi-user communication [J]. IEEE transactions on wireless communications, 2022, 21(9): 7287 – 7298. DOI: 10.1109/TWC.2022.3157467

[15] VASWANI A, SHAZEER N, PARMAR N, et al. Attention is all you need [C]//Proc. 31st International Conference on Neural Information Processing Systems (NIPS'17). ACM, 2017: 6000 – 6010

[16] HUANG J H, YUAN K, HUANG C, et al. D2-JSCC: digital deep joint source-channel coding for semantic communications [J]. IEEE journal on selected areas in communications, 2025, 43(4): 1246 – 1261. DOI: 10.1109/JSAC.2025.3531546.

[17] CHOUKROUN Y, WOLF L. Error correction code transformer [C]//Proc. 36th International Conference on Neural Information Processing Systems (NIPS'22). ACM, 2022: 38695 – 38705

[18] XU J, LI Z S, DU B W, et al. Reluplex made more practical: leaky ReLU [C]//Proc. IEEE Symposium on Computers and Communications (ISCC). IEEE, 2020: 1 – 7. DOI: 10.1109/ISCC50000.2020.9219587

Biographies

LIU Yichen is currently pursuing the PhD degree in the School of Electronic Information and Communications, Huazhong University of Science and Technology, China. His research interests include wireless communications, artificial intelligence, and WLAN systems.

GAO Ruixin is currently pursuing the MS degree in the School of Electronic Information and Communications, Huazhong University of Science and Technology, China. Her research interests include machine learning for wireless communications, artificial intelligence, and WLAN systems.

ZENG Chen is currently pursuing the PhD degree in the School of Electronic Information and Communications, Huazhong University of Science and Technology, China. His research interests include machine learning for wireless communications, artificial intelligence, and WLAN systems.

LIU Yingzhuang (liuyz@hust.edu.cn) is currently a professor with the School of Electronic Information and Communications, Huazhong University of Science and Technology, China. Prior to that, he was a postdoctoral researcher with University of Paris XI, France from 2000 to 2001. Since 2003, he has led more than 10 national key projects, published more than 100 papers, and obtained more than 50 patents in broadband wireless communications. His main research interests are in broadband wireless communications, including LTE-Advanced, 5G/6G, and WLAN systems.