

SigT: An Efficient End-to-End MIMO-OFDM Receiver Framework Based on Transformer

Ziyou Ren*, Nan Cheng*, Ruijin Sun*, Xiucheng Wang*, Ning Lu[†] and Wenchao Xu[‡]

*School of Telecommunications Engineering, Xidian University, Xi'an, China

[†]Department of Electrical and Computer Engineering, Queen's University, Kingston K7L 3N6, Ontario, Canada

[‡]Department of Computing, The Hong Kong Polytechnic University, Hongkong, China

Email: {zyren_681,xcwang_1}@stu.xidian.edu.cn, dr.nan.cheng@ieee.org,sunruijin@xidian.edu.cn
ning.lu@queensu.ca, wenchao.xu@polyu.edu.hk

Abstract—Multiple-input multiple-output and orthogonal frequency-division multiplexing (MIMO-OFDM) are the key technologies in 4G and subsequent wireless communication systems. Conventionally, the MIMO-OFDM receiver is performed by multiple cascaded blocks with different functions and the algorithm in each block is designed based on ideal assumptions of wireless channel distributions. However, these assumptions may fail in practical complex wireless environments. The deep learning (DL) method has the ability to capture key features from complex and huge data. In this paper, a novel end-to-end MIMO-OFDM receiver framework based on *transformer*, named SigT, is proposed. By regarding the signal received from each antenna as a token of the transformer, the spatial correlation of different antennas can be learned and the critical zero-shot problem can be mitigated. Furthermore, the proposed SigT framework can work well without the inserted pilots, which improves the useful data transmission efficiency. Experiment results show that SigT achieves much higher performance in terms of signal recovery accuracy than benchmark methods, even in a low SNR environment or with a small number of training samples. Code is available at <https://github.com/SigTransformer/SigT>.

Index Terms—Deep learning, Transformer, MIMO-OFDM, self-attention.

I. INTRODUCTION

The multiple-input multiple-output and orthogonal frequency-division multiplexing (MIMO-OFDM) are key technologies in the physical layer for 4G and 5G broadband wireless systems and will keep playing an important role in 6G wireless systems. Benefiting from the beamforming gain, the diversity gain, and the multiplexing gain, MIMO technology can significantly increase the system spectral efficiency and reliability [1]. By transforming the digital signal from the time domain to the frequency domain to transmit, OFDM technology can effectively deal with the frequency-selective property of the broadband channel [2]. With the MIMO-OFDM technology, the conventional receiver consists of multiple cascaded blocks, i.e., fast Fourier transformation (FFT), MIMO channel estimation (CE), MIMO signal detection (SD), and demodulation. These blocks are designed independently to perform different functions and algorithms, which are based on ideal assumptions of channel distributions and linear interference. However, in practice, these assumptions may fail due to the complex wireless environment and the non-linear effect of hardware circuits,

resulting in deteriorated bit error rate (BER) performance. Deep learning (DL) techniques [3], which has achieved great success in the field of natural language processing (NLP) and computer vision (CV), have been widely applied to the field of communication in recent years [4], [5]. Since DL can capture the complex and non-linear features of wireless channels by feeding with huge labeled data, it can be a potential way to solve these issues. To reap this advantage, many researchers introduced the DL technologies into the physical layer of communications and studied the DL-based OFDM receiver [6]–[9]. Both DL-based individual receiver block, such as CE [10], MIMO SD [6], channel state information feedback [8], and autoencoder-based end-to-end communication system [11], and DL-based end-to-end OFDM receiver [9], [10], [12] have been investigated. For the DL-based individual receiver blocks, the signal recovery process can be split into subnets. Each subnet is designated to serve the function of a conventional receiver's block, e.g., CE and SD, between which hand-crafted information are inserted and locally optimizes the subnet. However, this framework is based on expert knowledge and requires sophisticated design. Therefore, the DL-based end-to-end receiver framework is recently proposed and studied where little expert knowledge is exploited.

For the DL-based end-to-end receiver, the signal recovery process is regarded as a whole block. It can be optimized from a global perspective, which can deal with the complicated wireless environment and the mismatch of block splitting in the conventional receiver. In [7], an end-to-end DL-based receiver, named FC-DNN, is explored, which maps the received OFDM subcarriers to constellation symbols. Although such a network performs well in signal detection tasks, expert knowledge is still required for quadrature amplitude modulation (QAM) demodulation block. FC-DNN is further evaluated in [9], which reveals that FC-DNN performs well even in mismatched channels and achieves competitive performance compared to DL-based individual-block cascading networks. However, compared with the DL-based individual-block cascading receiver design, the DL-based end-to-end method treats the signal recovery in the receiver as a black box, and cannot exploit the abundant expert knowledge in the field of communication systems. This leads to the fact that

when the length of input digits increases, the possibility of recovered signal increases exponentially, and so does the signal recovery difficulty. This issue is often referred to as zero-shot problem [13]. Especially when it comes to the MIMO case, the issue gets worse since the input digit sequence is usually much longer than in the single-input single-output (SISO) case.

Furthermore, existing literature reveals that the structure of the neural network significantly affects the convergence and performance of the DL-based models. Except for the FC-DNN and convolutional neural network (CNN) adopted in the literature, many emerging DL structures have shown powerful ability in NLP and CV. In [14], a novel sequence-to-sequence (Seq2Seq) neural model for machine translation, *transformer*, is proposed and achieves state-of-the-art performance. In [15], Vision Transformer (ViT) is proposed to extend transformer to computer vision tasks, e.g., object recognition, by employing transformer encoder as the model backbone.

Motivated by the discussion above, in this paper, we propose a Transformer-based DL model to design an end-to-end MIMO-OFDM receiver, named signal Transformer (SigT). By treating the signal received at each antenna as a token of the Transformer framework, the spatial correlation of different antennas can be learned and the critical zero-shot problem can be mitigated. The main contributions of this paper are summarized as follows.

- A novel neural network framework called SigT is proposed in this paper to design the end-to-end MIMO-OFDM receiver. The proposed SigT can serve as a reference and baseline for future study of end-to-end learning-based MIMO-OFDM receiver design.
- Compared with most existing DL-based receivers, our proposed SigT MIMO-OFDM receiver can recover the data signal with high accuracy without any foreknown pilots. By saving the time-frequency resource occupied by pilots, the proposed SigT-based wireless communication systems can transmit more useful data signals in each frame and thus enlarge the effective channel capacity.
- We thoroughly evaluate the performance of the proposed SigT through extensive experiments with a real dataset. Experiment results show that SigT outperforms existing DL-based end-to-end models by a large margin.

The remainder of the paper is organized as follows. Section II gives the preliminaries on the MIMO-OFDM system and describes the task of end-to-end MIMO-OFDM receiver design. In Section III, the proposed SigT framework is described in detail. In Section IV, the performance of SigT is evaluated through experiments. Finally, Section V concludes the paper.

II. PRELIMINARIES AND PROBLEM STATEMENT

In this section, we first shortly introduce the preliminaries on the MIMO-OFDM system and DL-based receiver. Then, we describe the specific task of end-to-end DL-based MIMO-OFDM receiver design.

A. MIMO-OFDM System

As illustrated in Fig. 1(a), a typical MIMO-OFDM system includes N_t -antenna transmitter, N_r -antenna receiver, N_s subcarriers, and N_i information in each subcarrier. For the baseband process in the transmitter, the digital bit stream X successively goes through the QAM modulation block, the MIMO precoding block, the IFFT block, and the CP adding a block. During the QAM modulation, the bit stream is modulated as the data symbol stream in complex form. Then, the MIMO precoding is designed to map the data symbol stream to the transmitter antennas. In each antenna, the useful data symbol sequence and a few pilot symbols are grouped together to perform the IFFT operation and generate the OFDM symbol. Thereafter, the CP is added before each OFDM symbol to combat the inter-symbol interference (ISI) caused by the multi-path effect of wireless channels.

In the conventional MIMO-OFDM receiver, the received time-domain signal goes through the following blocks in turn: the CP removing, the FFT, the MIMO CE, the MIMO signal detection, and the QAM demodulation. In particular, in each receiver antenna, the CP contaminated by the ISI is first removed and the FFT is performed to transform the received time-domain signal to the frequency domain. Then, the received pilot symbols of N_r antennas are used to estimate the wireless MIMO channel with the dimension of $N_r \times N_t$. With the estimated MIMO channel, the received data symbols of N_r antennas are detected via the receiver precoding. Thereafter, the detected symbol stream is sent to the QAM demodulation block to recover the original binary bit stream \tilde{X} .

B. End-to-End MIMO-OFDM Receiver Design

In this part, we describe the task of end-to-end DL-based MIMO-OFDM receiver design, as shown in Fig. 1(b). The DL-based model serves as a receiver block that substitutes CE, SD, and QAM demodulation in the conventional MIMO-OFDM system. The input of the model $Y \in \mathbb{R}^{N_s \times N_r \times N_i \times 2}$ after FFT is fed into DL-based model and the model directly outputs the predicted label $\hat{X} \in \mathbb{R}^{N_s \times N_t \times 2}$. \hat{X} is a string of float numbers predicted during the training phase and is used to compute mean square error (MSE) loss during the model training phase. In the test phase, the model outputs \tilde{X} , which is a bitstream after applying a hard decision to \hat{X} . The dimension of \hat{X} and \tilde{X} should be the same as X , i.e., $\hat{X}, \tilde{X} \in \mathbb{R}^{N_s \times N_t \times 2}$.

For the performance of the MIMO-OFDM receiver, we follow the average accuracy (AACC) criteria. AACC is calculated using the following equation:

$$AACC = 1 - \sum_{n=1}^N \frac{|\tilde{X}_n - X_n|}{N}, \quad (1)$$

where N is the total number of test signals, \tilde{X}_n and X_n is the n_{th} signal in the test set.

In end-to-end receiver design, the model needs to complete tasks of CE, SD, and QAM demodulation. This is far more complicated and difficult than simply fitting the function of an individual module. Since no expert knowledge is exploited,

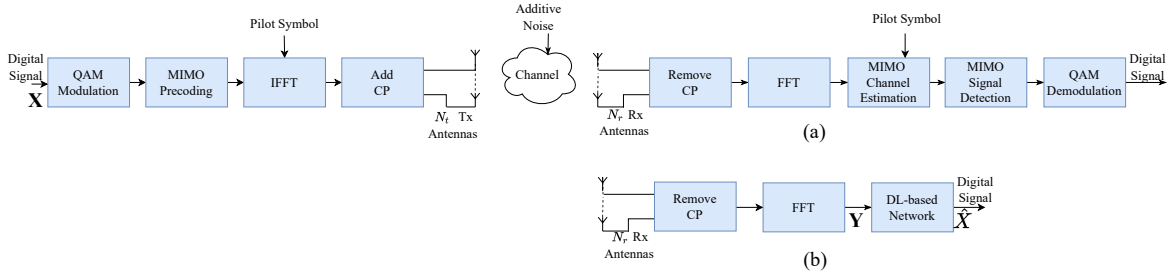


Fig. 1. Block diagram of MIMO-OFDM system. (a) is the conventional MIMO-OFDM receiver system (b) is the receiver design of an end-to-end DL-based model.

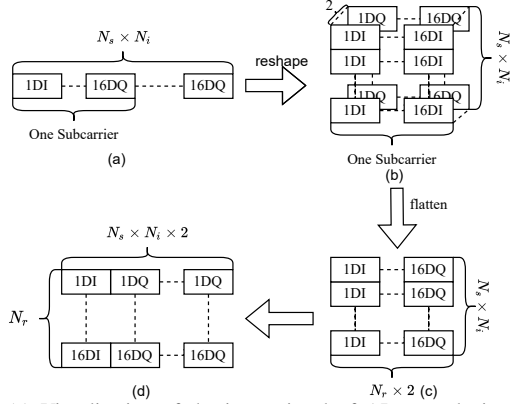


Fig. 2. (a) Visualization of the input signal of AI network, i.e. Y in our denotation. We plot using $N_r = 16$, $N_t = 4$ for demonstration. (b) we reshape the signal by corresponding each row to one of the 256 subcarriers. (c) we flatten the $3D$ tensor to be $2D$ to meet the input requirement of the Transformer module. (d) transpose the tensor and split each row as a token, which contains the information of an antenna, to feed into our network.

the model learns by making the predicted label approaching the ground truth. However, in this task, the dimension of the ground truth labels increases exponentially with the bitstream length, i.e., $2^{N_s \times N_t \times 2}$. Therefore, it is extremely hard for the dataset to cover all labels due to both the hardware storage limitation and the cost to collect the data, which requires the model to preserve the capability of predicting labels that are not included in the training set, leading to the zero-shot problem. The zero-shot problem may result in a large gap between training accuracy and test accuracy, which is referred to as overfitting. In the sequel, we design a transformer-based model, named SigT, to better learn the correlation among antennas and other higher semantics to improve the model generalization performance.

III. SIGNAL TRANSFORMER

In this section, we first briefly introduce the multi-head self-attention mechanism (MSA), which is the key module of the transformer. Then, we propose a transformer-based end-to-end MIMO-OFDM receiver model, SigT, and give the design details.

A. Multi-Head Self-Attention Mechanism

Self-attention is a basic block in Transformer, which captures the feature for each vector and aggregates them together to get an output set of features that has the same dimension and number as the input set of vectors. The pivotal issue is to get query (Q), key (K), and value (V) for each input vector

by applying a linear transformation. Then, we can compute the attention by the following equations.

$$Q = W^Q \cdot I, W^Q \in \mathbb{R}^{d_Q \times d}, \quad (2a)$$

$$K = W^K \cdot I, W^K \in \mathbb{R}^{d_K \times d}, \quad (2b)$$

$$V = W^V \cdot I, W^V \in \mathbb{R}^{d_V \times d}, \quad (2c)$$

$$\text{Attention}(I) = V \cdot \text{Softmax}\left(\frac{K^T Q}{\sqrt{d_K}}\right), \quad (2d)$$

where I is the tensor concatenated by the input sequence of vectors, d is the dimension of input vectors, d_Q, d_K, d_V are dimension of Q, K, V , respectively. Normally, we set $d_Q = d_K$ such that we can scaled $K^T Q$ by multiplying it with $\frac{1}{\sqrt{d_K}}$.

Then, a multi-head attention block can be built based on the aforementioned self-attention block.

$$\text{MultiHead} = W^O \cdot \text{Concat}(\text{head}_1, \text{head}_2, \dots, \text{head}_h), \quad (3a)$$

$$\text{head}_i = \text{Attention}(I_i), \quad (3b)$$

B. Signal Transformer Model

In MIMO systems, each receiver antenna receives the signal from all transmitting antennas. Thereby, the received signals of each receiver antenna are strongly correlated, and this correlation plays an important role in recovering the signal. To capture this correlation, we propose SigT, which is based on the transformer, where the correlation among antennas is learned by self-attention during signal recovery. The overall architecture of SigT is shown in Fig. 3.

1) *Tokenization*: The standard Transformer model takes in a sequence of token embeddings, then outputs features for each token. To better fit the transformer model and learn the correlation among antennas, proper tokenization is required. In Signal transformer, the input data is permuted and reshaped to form the tokens, as shown in Fig. 2. Given a high dimension vector, we first need to reshape it according to its physical properties, e.g. number of antennas, number of subcarriers, etc., then we reduce the dimension to form a $2D$ tensor. Finally, we split the tensor on the dimension of N_r into N_r vectors, each serves as a token, and feed the sequence of tokens into the transformer Encoder module.

2) *Transformer Encoder*: After tokenization, token sequences are fed into the transformer Encoder. The design of the transformer Encoder is the same as the original design [14]. This structure is able to capture the relation between different tokens, which, in this circumstance, complements missing

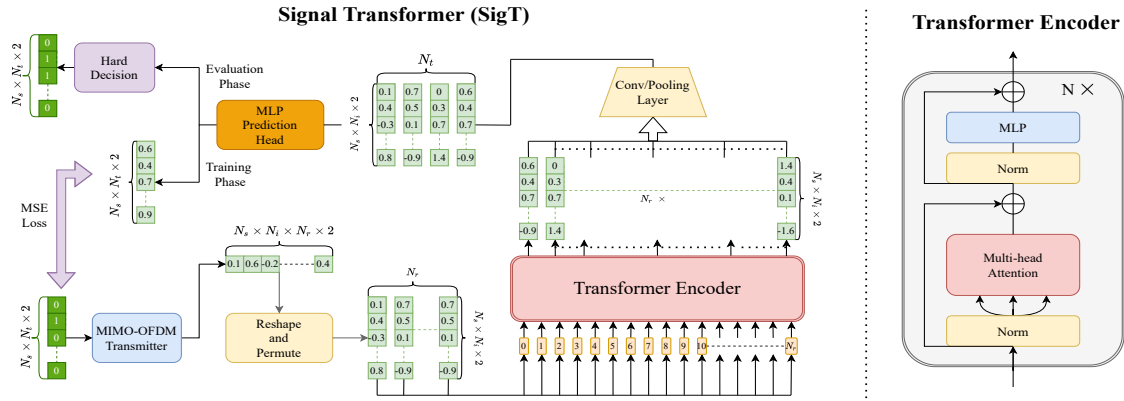


Fig. 3. Model review (left). We first reshape and permute the input signal, and retain all subcarriers from one antenna to a single vector. Then the processed signal is fed into the transformer Encoder, which will output a sequence of features each has the same dimension as the inputs. After that, we reduce the number of vectors by going through pooling layers or convolutional layers to N_t vectors. Finally, we concatenate N_t vectors and pass through an MLP prediction head to extract the features and get the target vector that has the same dimension as the ground truth signal. Note that during the test phase, we apply an additional hard decision layer to convert predicted float numbers to a bitstream. The detail of the Transformer Encoder (right) is inspired by [14].

information of some antennas and enhances the important information shared among antennas.

Normally, MSA performs better than single-head attention while not increasing asymptotic complexity. This is because multiple types of relations can be learned rather than only one type of relation in the single-head attention mechanism.

3) *Aggregation*: The output features of the transformer Encoder are passed through pooling layers or convolutional layers to aggregate the information from nearby features. Here, we denote the output features of transformer Encoder, i.e., the input features of pooling/convolutional layers, by $f_i \in \mathbb{R}^{N_s \times N_i \times N_r \times 2}$, $i = 1, 2, \dots, N_r$, and the output features of pooling/convolutional layer by $p_i \in \mathbb{R}^{N_s \times N_t \times 2}$, $i = 1, 2, \dots, N_t$, respectively.

4) *Output and Loss Calculation*: Finally, we concatenate $p_i \in \mathbb{R}^{N_s \times N_t \times 2}$, $i = 1, 2, \dots, N_t$ to form P and feed it into a two-layer MLP to get \hat{X} as follows.

$$P = \text{Concat}(p_1, p_2, \dots, p_{N_t}), \quad P \in \mathbb{R}^{N_s \times N_t \times 2}, \quad (4a)$$

$$\hat{X} = \text{Sigmoid}(\text{MLP}(P)), \quad \hat{X} \in \mathbb{R}^{N_s \times N_t \times 2}, \quad (4b)$$

$$\tilde{X} = \text{Hardmax}(\hat{X}), \quad \tilde{X} \in \mathbb{R}^{N_s \times N_t \times 2}, \quad (4c)$$

where P denotes the input features, and \hat{X} denotes the output features activated by the *Sigmoid* function, which maps the real number to the range from 0 to 1. \tilde{X} is a bitstream which is the hard decision of \hat{X} .

During the training phase, we update the weights in MLP, conv/pooling layer, and transformer Encoder by computing mean square error (MSE) loss between \hat{X} and Y as $Loss = \frac{1}{N_g} \sum_{i=1}^{N_g} (\hat{x}_i - x_i)^2$, where $N_g = 2N_s N_t$ denotes the number of digits of signals, and x_i and \hat{x}_i denote the i_{th} bit in X and \hat{X} , respectively.

IV. EXPERIMENT AND DISCUSSION

In this section, we first describe the dataset, experiment configurations, and two DL-based models which are used as benchmarks. Then, experiment results are shown and discussed to evaluate the performance of the proposed SigT.

A. Experiment Configurations

To evaluate the performance of the proposed SigT, we use the channel data provided by Peng Cheng Laboratory (PCL) in national artificial intelligence competition (NAIC)¹. The channel data is obtained in typical city area. The central frequency of the system is 3.5 GHz, and was sampled from 500 locations, each with 20 measurements. We use this open data to generate our dataset.

B. Performance Comparison with Benchmark Methods

In the experiment, we set the number of subcarriers for OFDM to $N_s = 256$, number of transmitting antennas to $N_t = 4$, number of receiving antennas to $N_r = 16$. In each subcarrier, we have one information symbol for each antenna, that is $N_i = 1$. An illustration of serial signal Y is depicted in Fig. 2, where we simply permute and reshape the original data to fit our model and without any additional transformations.

To construct our dataset, we keep the size of test set over the size of training set to be 1/10. By default, the training set has 25600 signals, and the test set has 2560 signals. To load data more efficiently, we set the size of minibatch to be 640, i.e., loading 640 signals per iteration, and control the size of dataset using the number of minibatches, which is denoted by NB in Fig. 6.

There are other hyperparameters that can affect the results of AACC. For simplicity, we set the optimizer to be adaptive momentum (Adam) with learning rate 0.0001, $\beta_1 = 0.9$, $\beta_2 = 0.999$, $\epsilon = 10^{-8}$. We do not apply the weight decay, i.e., regularization, unless otherwise specified.

C. Benchmark DL-Based Models

The benchmark methods used in the experiment are the DL-based models in [8], [9]. We adapt the models to work with the MIMO-OFDM scenario and the used dataset.

1) FC-DNN. This network consists N_r independent MLPs, each following the design in [9] but doubling the size of hidden layers.

¹<https://naic.pcl.ac.cn/contest/10/34>

TABLE I
MODEL STRUCTURE EXPLORATION

Model backbone	Conv Layer	Pooling Layer	Dropout	Optimizer	Train SNR (dB)	Train AACC (%)	test aacc (%)
Transformer Encoder		✓		Adam	10	96.87	76.07
	✓			Adam	10	98.31	79.54
		✓		SGD	10	72.89	70.27
		✓	✓	Adam	10	85.57	69.42
	✓		✓	Adam	10	89.48	71.79
	✓			Adam	25	None	79.28
LSTM		✓		Adam	25	None	76.23
	✓			Adam	10	58.5	50.61
	✓		✓	Adam	10	57.75	50.86

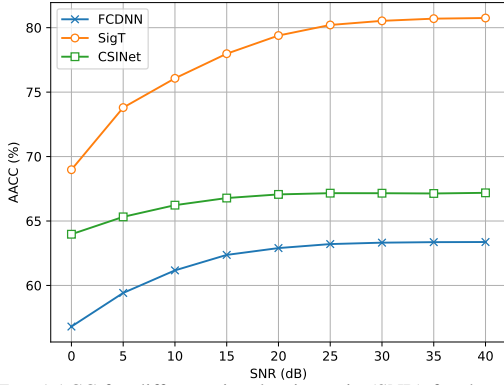


Fig. 4. Test AACC for different signal noise ratio (SNR) for three DL-based models, i.e. FC-DNN, CSINet, and SigT.

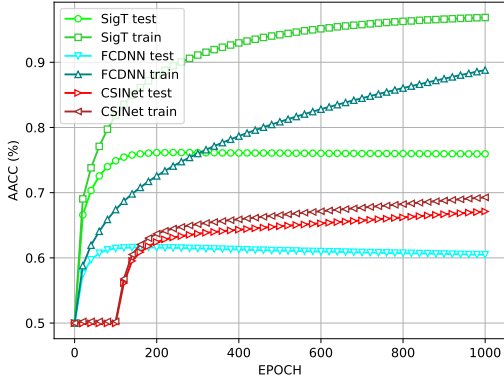


Fig. 5. The AACC on the training set and test set for the three models.

2) CSINet. CSINet is stacked from RefineNet module, which is a residual CNN in [8]. Flatten layer is added to match the signal to the ground truth.

In the experiment, we comprehensively evaluate the performance of our proposed model, SigT, and compare it with the benchmark DL-based models, namely MLP-structured FC-DNN and CNN-structured CSINet.

Fig. 4 compares the AACC performance under different SNRs. During the training phase, we construct the training set that has the same SNR as the test set. The results show that SigT outperforms the other two models by a large margin since MSA can efficiently exploit the correlation among the receiving antennas. For SNR lower than 10 dB, all three models' AACC increases fast with the increase of SNR. This is because when the SNR becomes higher, the models can gradually learn the signal patterns which are covered by the noise when SNR is low. However, when SNR gets even higher, the error floor occurs mainly due to the generalization difficulty of the end-

to-end model in the MIMO-OFDM setting.

In Fig. 5, the performance on both the training set and test set is shown with 10 dB SNR. It can be seen that the proposed SigT achieves 14.9% higher AACC than FC-DNN and 8.95% higher AACC than CSINet. For FC-DNN, the large gap between training AACC and test AACC and the waning of test AACC after 400 epochs reveals that the FC-DNN model faces the problem of overfitting. Because the receiving system of MIMO-OFDM is too complicated and needs huge expert knowledge to fully understand. Thus, simply-designed DL networks cannot perform well on unseen signals. Similarly for the CSINet model, underfitting is normally derived from poor representative power. The proposed SigT model, in comparison, reaches a rather high AACC on the training and test set and grows faster than CSINet and FC-DNN. This high AACC and increasing rate indicate that SigT is able to capture the features and expert knowledge.

The relation between the training set size and the model performance is shown in Fig. 6. In this simulation, we construct a batch with batch size equals 64 training sample signals, and test the performance of SigT on different number of training signals by changing the number of batches, i.e. $num_signals = NB \times 64$. As we can see from Fig. 6, the AACC on test set grows rapidly with the increase of NB when NB is small and starts to saturate when NB gets larger. This shows that simply increasing the size of the training set does not always result in improved learning performance.

D. Ablation Study

The ablation study of the proposed SigT model is conducted in this part.

The difference between the pooling layer and the convolutional layer is that the convolutional layer serves as a linear combination of features and is able to capture more local information and the pooling layer only serves to reduce the dimension of features. As a result, it is not unexpected that switching the pooling layer to the convolutional layer improves the performance of the model. As can be seen in the TABLE I, SigT with convolutional layers reaches higher AACC on both training set and test set than that with pooling layers.

Dropout is normally used for preventing overfitting by randomly masking part of neural units in MLP. However, such mask strategy would reduce the representative power when the size of hidden layers is not overly large. As shown in TABLE I, we apply dropout on both conv-based SigT and pool-based

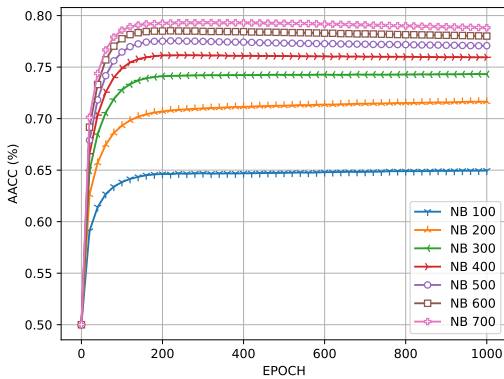


Fig. 6. Test AACC of SigT for different NB with $SNR = 10$ dB

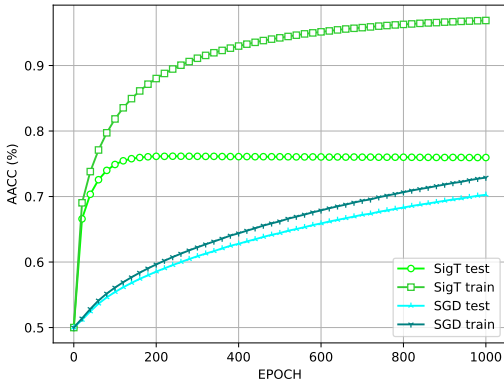


Fig. 7. The AACC on the training set and test set for three models.

SigT's MLP head, the performance is lower than not applying dropout. Such results demonstrate that SigT performs well enough and can learn knowledge from training instead of just fitting a mapping function.

Different optimizers are also compared, where we explore two commonly used optimization functions, namely stochastic gradient descent (SGD) and adaptive momentum (Adam). As shown in Fig. 7, the model using Adam converges more quickly and reaches higher AACC. In comparison, the model using SGD not only converges more slowly but also performs poorly.

At last, to directly show the representative power of the transformer model, we test the performance of another widely used RNN model, long short-term memory (LSTM), as a backbone and retain the remaining part the same as SigT. To get the best results of LSTM, we apply convolutional layers, apply no dropout method, and use Adam to train 1000 epochs. We can see that the LSTM-backbone model can hardly learn anything. With training AACC and test AACC far lower than SigT, we conclude that our model is more representative than other models.

V. CONCLUSION

In this paper, we have designed an end-to-end DL-based MIMO-OFDM receiver, named SigT, to efficiently detect and demodulate the receiving signal. The proposed SigT utilizes the transformer model to learn the knowledge of correlations among the receiving antennas to relieve the zero-shot problem confronted in end-to-end learning models. The experiment results have shown that the proposed SigT achieves the highest

performance compared with the other two benchmark end-to-end DL-based models in terms of signal recovery accuracy. The proposed SigT can serve as a performance baseline for future study of learning-based MIMO-OFDM receiver design. In future work, we will study the dynamic mechanisms in DL models to adapt the DL-based receiver to dynamic environments.

ACKNOWLEDGEMENT

This work was supported by the National Key Research and Development Program of China (2020YFB1807700), the National Natural Science Foundation of China (NSFC) under Grant No. 62071356, and the Fundamental Research Funds for the Central Universities under Grant No. JB210113.

REFERENCES

- [1] E. G. Larsson, O. Edfors, F. Tufvesson, and T. L. Marzetta, "Massive mimo for next generation wireless systems," *IEEE communications magazine*, vol. 52, no. 2, pp. 186–195, 2014.
- [2] G. L. Stuber, J. R. Barry, S. W. McLaughlin, Y. Li, M. A. Ingram, and T. G. Pratt, "Broadband mimo-ofdm wireless communications," *Proceedings of the IEEE*, vol. 92, no. 2, pp. 271–294, 2004.
- [3] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Advances in Neural Information Processing Systems*, vol. 25, pp. 1–9, 2012.
- [4] L. Ma, N. Cheng, X. Wang, R. Sun, and N. Lu, "On-demand resource management for 6g wireless networks using knowledge-assisted dynamic neural networks," in *ICC 2022 - IEEE International Conference on Communications*, 2022, pp. 1–6.
- [5] X. Wang, L. Fu, N. Cheng, R. Sun, T. Luan, W. Quan, and K. Al-dubaikhy, "Joint flying relay location and routing optimization for 6g uav-iot networks: A graph neural network-based approach," *Remote Sensing*, vol. 14, no. 17, p. 4377, 2022.
- [6] H. He, C.-K. Wen, S. Jin, and G. Y. Li, "Model-driven deep learning for mimo detection," *IEEE Transactions on Signal Processing*, vol. 68, pp. 1702–1715, 2020.
- [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] J. Guo, X. Li, M. Chen, P. Jiang, T. Yang, W. Duan, H. Wang, S. Jin, and Q. Yu, "Ai enabled wireless communications with real channel measurements: Channel feedback," *Journal of Communications and Information Networks*, vol. 5, no. 3, pp. 310–317, 2020.
- [9] P. Jiang, T. Wang, B. Han, X. Gao, J. Zhang, C.-K. Wen, S. Jin, and G. Y. Li, "Ai-aided online adaptive ofdm receiver: Design and experimental results," *IEEE Transactions on Wireless Communications*, vol. 20, no. 11, pp. 7655–7668, 2021.
- [10] J. Liu, K. Mei, X. Zhang, D. Ma, and J. Wei, "Online extreme learning machine-based channel estimation and equalization for ofdm systems," *IEEE Communications Letters*, vol. 23, no. 7, pp. 1276–1279, 2019.
- [11] S. Dörner, S. Cammerer, J. Hoydis, and S. t. Brink, "Deep learning based communication over the air," *IEEE Journal of Selected Topics in Signal Processing*, vol. 12, no. 1, pp. 132–143, 2018.
- [12] X. Gao, S. Jin, C.-K. Wen, and G. Y. Li, "Comnet: Combination of deep learning and expert knowledge in ofdm receivers," *IEEE Communications Letters*, vol. 22, no. 12, pp. 2627–2630, 2018.
- [13] R. Socher, M. Ganjoo, C. D. Manning, and A. Ng, "Zero-shot learning through cross-modal transfer," vol. 26, 2013.
- [14] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.
- [15] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly *et al.*, "An image is worth 16x16 words: Transformers for image recognition at scale," *arXiv preprint arXiv:2010.11929*, 2020.