IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 16 April 2023, accepted 27 April 2023, date of publication 1 May 2023, date of current version 5 May 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3271993

RESEARCH ARTICLE

Auto-Encoder Based Orthogonal Time Frequency Space Modulation and Detection With Meta-Learning

JAEHYUN PARK[1], (Member, IEEE), JUN-PYO HONG[1], (Member, IEEE), HYUNGJU KIM[2], (Member, IEEE), AND BYUNG JANG JEONG[2], (Member, IEEE)

[1]School of Electronics and Communications Engineering, Pukyong National University, Busan 48513, South Korea
[2]Communication and Media Research Laboratory, Electronics and Telecommunications Research Institute, Daejeon 34129, South Korea

Corresponding author: Jun-Pyo Hong (jp_hong@pknu.ac.kr)

This work was supported in part by Electronics and Telecommunications Research Institute (ETRI) grant funded by the Korean government. [23ZH1100, Study on 3D Communication Technology for Hyperconnectivity.] It was also supported in part by the Basic Science Research Program through the National Research Foundation of Korea funded by the Ministry of Education (2018R1D1A1B07043786).

ABSTRACT To tackle a Doppler sensitivity problem of orthogonal frequency division multiplexing (OFDM), orthogonal time frequency space (OTFS) has been investigated, where information is carried over delay-Doppler domain. In this paper, to improve communication reliability in doubly dispersive channel, an auto-encoder (AE)-based OTFS modulation and detection scheme is developed, where the transmit OTFS waveform and its associated detection scheme at the receiver are jointly optimized in a deep learning framework. However, the conventional AE architecture which takes one-hot encoded input vector is hard to be reused in OTFS due to its enormous input dimensionality that increases exponentially on the number of grid points in delay-Doppler domain. To overcome it, we divide the delay-Doppler grid into multiple subblocks and associate the one-hot encoded vector with each subblock. Then, by concatenating them, one multi-hot vector is formed and exploited as the input vector for the proposed AE-based OTFS modulation and detection. We also develop a meta-learning scheme to effectively train the AE-based OTFS transceiver for newly updated channel profile.

INDEX TERMS Hierarchical auto-encoder, meta-learning strategy, OTFS waveform.

I. INTRODUCTION

Due to high frequency efficiency and immunity against multi-path fading, orthogonal frequency division multiplexing (OFDM) waveform has been extensively investigated and adopted for modern communication systems [1], [2]. The next generation of wireless communications will be a major move toward high-velocity applications such as vehicle-to-everything (V2X) communications and unmanned aerial vehicle (UAV) communications [3], [4], [5], [6]. However, under a high Doppler channel environment, orthogonality among subcarriers in OFDM waveforms is no longer maintained [7], [8].

To tackle the Doppler sensitivity problem of the OFDM, orthogonal time frequency space (OTFS) modulation has been investigated [9], [10], [11], [12] (and references therein), where information-carrying quadrature amplitude modulation (QAM) symbols are modulated in two dimensional (2D) delay-Doppler domain. Note that the channel variation is relatively smaller in the delay-Doppler domain than in the time-frequency domain [9], [13]. Accordingly, pilot overhead can be effectively reduced [14], [15].

To detect the QAM symbols in 2D delay-Doppler domain, linear minimum mean square error (LMMSE) receiver can be exploited by transforming the 2D delay-Doppler symbol matrix into 1D symbol vector [16], requiring the inversion and multiplication of matrices with a high dimension. Specifically, its complexity is in the order of $O(M^3N^3)$ when $M \times N$

The associate editor coordinating the review of this manuscript and approving it for publication was Prakasam Periasamy.

grid points are deployed in 2D delay-Doppler domain for one OTFS symbol. In [17], the low complexity LMMSE receiver is proposed without any performance degradation. In addition, general wireless channel can be represented sparse in the delay-Doppler domain. Accordingly, by modulating QAM symbols in the delay-Doppler domain, it is able to reduce the overhead for tracking channel state information (CSI) and exploit a message passing (MP) based low-complexity detection algorithm [10]. In [18], by dividing the received 2D delay-Doppler domain signal into N multiple delay-domain signals, computationally-efficient transform-domain maximal ratio combining (TD-MRC) detection method is proposed, in which the effective channel matrix is partitioned into multiple Doppler submatrices associated with different delay taps and the matrix-inversion-free MRC algorithm is developed. In [19] and [20], multi-antenna OTFS modulation schemes and channel estimation have been investigated. The OTFS-based multiple access has been considered in [21] and [22] In addition, by exploiting the sparse signal representation in delay-Doppler domain, compressive-sensing based detection and channel estimation methods have been developed [23], [24].

Recently, deep learning based auto-encoders (AEs) have been successfully applied to solving complex problems in communication systems [25], [26], [27], [28], [29] and references therein. In [25], a transmitter, channel, and a receiver are modeled as one neural network and trained as the AE. Specifically, the idea of applying the AE to communication systems is to reduce a reconstruction-based loss function to jointly optimize encoding, decoding, and signal representation. The trained encoder then yields the transmit signal which can be different from the conventional QAM constellation points. Here, the one-hot encoded vector (i.e., a vector that has only one non-zero element with a unit value) is exploited as an input of the encoder to model information bits to be transmitted. The AE-based transceiver design has been also extended to multiple-input multiple-output (MIMO) [26] and multi-user MIMO communication systems [27]. In [30] and [31], AE-based end-to-end learning has been applied to design OFDM symbols. In [30], the signal carried on each subcarrier is optimized in conjunction with the channel estimation. In [31], the constellation points over all subcarriers are jointly optimized through the AE-based end-to-end learning. However, in the AE based transceiver, the one-hot encoded input vector size increases exponentially with respect to the number of information bits to be encoded. For example, with M subcarriers and M^{c} possible messages (corresponding to M^{c}-ary QAM constellation) per subcarrier, the size of the input vector becomes $(M^{\mathrm{c}})^{M}$.

In this paper, to improve detection performance, AE-based OTFS modulation and detection scheme is developed, where the transmit OTFS waveform and its associated detection scheme at the receiver are jointly optimized in a deep learning framework. Throughout the paper, the feed-forward neural network (FNN) architecture is considered for both transmitter and receiver. We note that the number of information bits modulated in one OTFS waveform is proportional to the number of grid points in 2D delay-Doppler domain. Accordingly, if an one-hot encoded vector is exploited as the input vector in the design of AE architecture for OTFS modulation and detection, its size increases exponentially on the number of grid points in 2D delay-Doppler domain, which makes it impossible to develop the AE architecture for OTFS modulation and detection. To overcome it, we divide the 2D delay-Doppler grid into multiple subblocks and associate the one-hot encoded vector with each subblock. Then, by concatenating them, one multi-hot vector is formed and exploited as the input vector for the proposed AE-based OTFS modulation and detection.

Because the OTFS transceiver and wireless channel are modeled as one neural network, if the channel is changed, the neural network for the OTFS transceiver should be re-trained with the updated channel. To cope with the channel variation, we also develop a meta-learning scheme for the proposed AE-based OTFS modulation and detection. We note that meta-learning is a fast and flexible learning method with few training samples on a new task (or environment) with variation [32], [33]. Model-agnostic meta-learning (MAML), a well-known meta-learning method, has been successfully adopted in artificial-intelligent-enabled wireless communication systems [34], [35], where the initial parameters of the model for fast adaptation is learned by alternating between inner-task procedure and cross-task procedure [35]. Therefore, MAML can provide a good initialization for being quickly adapted to the updated channel profile. Through computer simulations, the proposed AE-based OTFS transceiver is shown to outperform the conventional OTFS modulation with MP-based detection algorithm [10]. Furthermore, it is also shown that the proposed AE-based OTFS transceiver can quickly adapt to the newly updated channel, requiring much less training iterations compared to the conventional training method.

The contributions of this paper are as follows:

- To improve the communication reliability in doubly dispersive channel, we first develop the AE architecture for OTFS modulation and detection, where the transmit OTFS waveform and its associated detection scheme at the receiver are jointly optimized through the end-to-end learning strategy.
- To reduce the input size, a hierarchical structure for AE-based OTFS modulation and detection is proposed, where the 2D delay-Doppler grid is divided into multiple subblocks and an one-hot encoded vector is associated to each subblock. Accordingly, one multi-hot vector is formed and exploited as the input vector for the proposed AE-based OTFS modulation and detection. Furthermore, by slicing the FNN in AE architecture for OTFS modulation and detection, the required weights can be further reduced.
- Finally, we also develop a meta-learning strategy to adapt the proposed AE-based OTFS modulation and

detection scheme to channel variations. The simulation result shows that meta-learning strategy greatly accelerates the training process of the proposed AE-based OTFS modulation and detection in a newly generated channel.

The rest of this paper is organized as follows. In Section II, the OTFS transmit and receive signal model is introduced. In Section III, the AE-based modulation and detection scheme is presented. In Section IV, a new partitioned AE-based architecture with multi-hot input vector is proposed. In addition, the meta-learning strategy is also developed to enhance the adaptability of the proposed AE architecture on the newly updated channel. In Section V, we provide several simulation results and in Section VI, we give our conclusions.

II. OTFS SIGNAL MODEL

The notations for the OTFS signal model are followed from [10] and [36]. As illustrated in Fig. 1, the OTFS signal frame is composed of N symbols, each with the symbol duration of T and the bandwidth of $M\Delta f$. We then define the discretized time-frequency plane and the associated delay-Doppler plane, respectively, as

$$\Lambda = \{(m\Delta f, nT), m = 0, \ldots, M-1, n = 0, \ldots, N-1\}, \tag{1}$$

$$\Gamma = \left\{ \left(\frac{l}{M\Delta f}, \frac{k}{NT} \right), l = 0, \ldots, M-1, k = 0, \ldots, N-1 \right\}, \tag{2}$$

where $\frac{1}{M\Delta f}$ and $\frac{1}{NT}$ are, respectively, the grid spacing for the delay and Doppler bins in the delay-Doppler plane.

A. TX SIGNAL MODEL

We note that, in the OTFS modulation, one information symbol is carried on each grid point of the delay-Doppler plane and then the aggregated symbols in the delay-Doppler plane are transformed into the time-frequency plane. Specifically, by letting $s[m, n]$ be the message on the (m, n)th grid point, one out of M^{c} possible messages (i.e., $\log_2 M^{\mathrm{c}}$ bits are conveyed on each message), the associated transmit symbol $X[m, n]$ can be mapped as

$$X[m, n] = f(s[m, n]) \in \mathbb{C}, \tag{3}$$

and each OTFS waveform carries K bits, where $K = MN \log_2 M^{\mathrm{c}}$. The M^{c}-ary QAM constellation mapper is a well-known example of $f(\cdot)$. Then, by applying the inverse symplectic finite Fourier transform (ISFFT) to $X[m, n]$ on delay-Doppler domain, the OTFS signal in time-frequency domain, $X_{\mathrm{tf}}[m, n]$, can be obtained as

$$X_{\mathrm{tf}}[m, n] = \frac{1}{\sqrt{MN}} \sum_{k=0}^{N-1} \sum_{l=0}^{M-1} X[l, k] e^{j2\pi \left(\frac{nk}{N} - \frac{ml}{M} \right)}. \tag{4}$$

By exploiting the pulse-shaping filter, $g_{tx}(t)$, and the inverse discrete Fourier transform (IDFT) operation, the OTFS signal in (4) is then transformed into the time domain baseband signal as

$$x_{\mathrm{bs}}(t) = \sum_{n=0}^{N-1} \sum_{m=0}^{M-1} X_{\mathrm{tf}}[m, n] g_{\mathrm{tx}}(t - nT) e^{j2\pi m\Delta f(t-nT)}. \tag{5}$$

By sampling $x_{\mathrm{bs}}(t)$ with a sampling time $\frac{T}{M}$ and concatenating it into an $N \times M$ matrix, the sampled baseband signal can be given as [36]

$$\mathbf{X}_{\mathrm{bs}} = \mathbf{G}_{\mathrm{tx}} \mathbf{F}_M^H \left(\mathbf{F}_M \mathbf{X} \mathbf{F}_N^H \right) = \mathbf{G}_{\mathrm{tx}} \mathbf{X} \mathbf{F}_N^H, \tag{6}$$

where $\mathbf{G}_{\mathrm{tx}}$ is a diagonal matrix whose elements are composed of the samples $g_{\mathrm{tx}}(t)|_{t=\frac{T}{M}m}$, $m = 0, \ldots, M-1$ and $\mathbf{F}_M$ is an M-point DFT matrix with the (m, n)th element as $\frac{1}{\sqrt{M}} e^{j2\pi \frac{mn}{M}}$. In addition, $\mathbf{X}$ is an OTFS signal matrix in delay-Doppler domain whose elements are from (4).

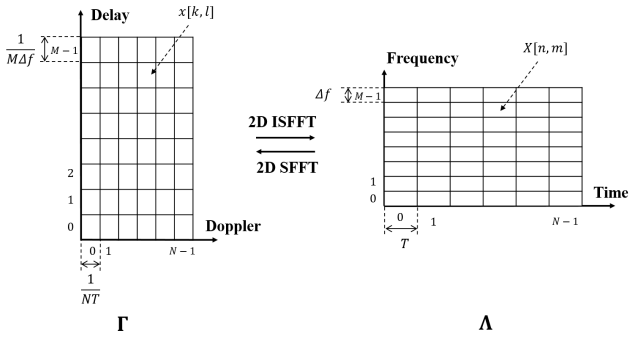

FIGURE 1. The relationship between the discretized time-frequency plane and the associated delay-Doppler plane.

B. RX SIGNAL MODEL

The time-varying multi-path channel in the delay-Doppler domain can be modeled as

$$h(\tau, \nu) = \sum_{i=1}^{P} h_i \delta(\tau - \tau_i) \delta(\nu - \nu_i), \tag{7}$$

where h_i, τ_i, and ν_i are the path gain, delay, and Doppler frequency of the i-th path, respectively. In addition, let l_i and k_i be the discrete delay and Doppler frequency for the i-th path, respectively, given as

$$l_i = M\Delta f \tau_i, \quad k_i = NT\nu_i. \tag{8}$$

That is, to shed light on the idea, it is assumed that τ_i and ν_i are the integer multiples of $\frac{1}{M\Delta f}$ and $\frac{1}{NT}$, respectively. We also note that the fractional delay and Doppler frequency can be handled by appending additional effective integer multipath taps. Then, the discrete received signal sequence $y_{\mathrm{bs}}[n]$ can then be expressed as [17], [18], and [36]

$$y_{\mathrm{bs}}[n] = \sum_{i=1}^{P} h_i e^{j2\pi \frac{k_i(n-l_i)}{MN}} x_{\mathrm{bs}}[[n - l_i]_{MN}] + \bar{w}[n], \tag{9}$$

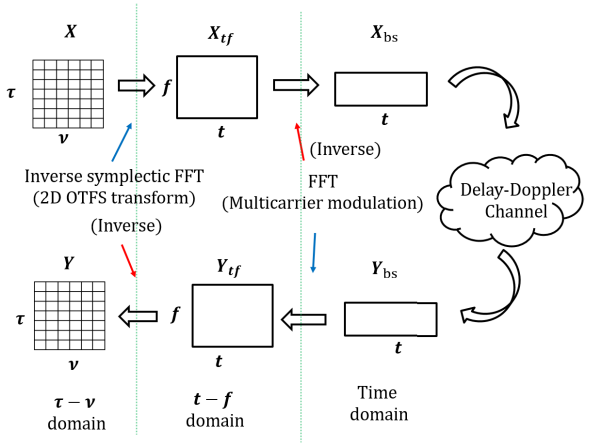

FIGURE 2. The overall process for the OTFS transceiver.

where $\bar{w}[n]$ is zero-mean Gaussian random variable with a variance, σ_{n}^2. Here, $[n]_{MN}$ indicates the MN-modulo operation due to the cyclic prefix over the OTFS frame. Then, $y_{\text{bs}}[n]$ can be expressed in a vector form as

$$\mathbf{y}_{\text{bs}} = \begin{bmatrix} y_{\text{bs}}[0] \\ y_{\text{bs}}[1] \\ \vdots \\ y_{\text{bs}}[MN-1] \end{bmatrix} = \mathbf{H}\mathbf{x}_{\text{bs}} + \mathbf{w}, \tag{10}$$

where $\mathbf{x}_{\text{bs}}$ is the vectorization of $\mathbf{X}_{\text{bs}}$, i.e., $\mathbf{x}_{\text{bs}} = \text{vec}(\mathbf{X}_{\text{bs}}) \in \mathbb{C}^{MN\times 1}$. In addition, the channel matrix $\mathbf{H}$ can be given as

$$\mathbf{H} = \sum_{i=1}^{P} h_i \mathbf{\Pi}^{l_i} \mathbf{\Delta}^{k_i}, \tag{11}$$

where $\mathbf{\Delta}$ is an $MN \times MN$ diagonal matrix as $\mathbf{\Delta} = diag\{z^0, z^1, \ldots, z^{MN-1}\}$ with $z = e^{\frac{j2\pi}{MN}}$ and $\mathbf{\Pi}^{l_i}$ is the $MN \times MN$ cyclic shift matrix given as $\mathbf{\Pi}^{l_i} = [\mathbf{e}_2, \mathbf{e}_3, \ldots, \mathbf{e}_{MN-1}, \mathbf{e}_{MN}, \mathbf{e}_1]$. Here, $\mathbf{e}_i$ is the i-th column of the $MN \times MN$ identity matrix, $\mathbf{I}_{MN}$.

We note that $\mathbf{y}_{\text{bs}}$ is the received baseband signal in time domain and it can be transformed into time-frequency domain by applying discrete Fourier transform (DFT) as $\mathbf{Y}_{\text{tf}} = \mathbf{F}_M \mathbf{G}_{\text{rx}} \text{vec}^{-1}(\mathbf{y}_{\text{bs}})$, where $\mathbf{G}_{\text{rx}}$ is a diagonal matrix whose elements are composed of the samples $g_{\text{rx}}(t)|_{t=\frac{T}{M}m}$, $m = 0, \ldots, M-1$ and $\text{vec}^{-1}(\cdot)$ is the inverse operation of $\text{vec}(\cdot)$ that restores the $M \times N$ matrix form from the $MN \times 1$ vectorization. Then $\mathbf{Y}_{\text{tf}}$ can be further transformed back into delay-Doppler domain by applying the symplectic finite Fourier transform (SFFT) as

$$\mathbf{Y} = \mathbf{F}_M^H \mathbf{Y}_{\text{tf}} \mathbf{F}_N = \mathbf{F}_M^H (\mathbf{F}_M \mathbf{G}_{\text{rx}} \text{vec}^{-1}(\mathbf{y}_{\text{bs}})) \mathbf{F}_N. \tag{12}$$

The overall process for the OTFS transceiver can be described as Fig. 2.

The received signal in delay-Doppler domain can then be derived in a vectorization form as [17], [18], and [36]

$$\begin{aligned} \mathbf{y} &= \text{vec}(\mathbf{Y}) = (\mathbf{F}_N \otimes \mathbf{G}_{\text{rx}})\mathbf{y}_{\text{bs}} \\ &= \underbrace{(\mathbf{F}_N \otimes \mathbf{G}_{\text{rx}})\mathbf{H}(\mathbf{F}_N^H \otimes \mathbf{G}_{\text{tx}})}_{\mathbf{H}_{\text{eff}}} \mathbf{x} + (\mathbf{F}_N \otimes \mathbf{G}_{\text{rx}})\mathbf{w}, \end{aligned}$$

where $\mathbf{x} = \text{vec}(\mathbf{X})$. We note that the effective channel $\mathbf{H}_{\text{eff}}$ is well analyzed in [36]. In addition, the pth column of $\mathbf{H}_{\text{eff}}$ implies the effective channel impulse response for the transmit signal at the (m, n)th grid point of delay-Doppler domain contributed to the received delay-Doppler signal domain, where $n = \lfloor \frac{p}{M} \rfloor$ and $m = p - nM$.

To make the signal in (13) compatible with the real-valued operations in general neural networks, we first transform (13) to the equivalent real-valued equations,

$$\bar{\mathbf{y}} = \bar{\mathbf{H}}_{\text{eff}} \bar{\mathbf{x}} + \bar{\mathbf{w}}, \quad \bar{\mathbf{H}}_{\text{eff}} \in \mathbb{R}^{2MN\times 2MN}, \tag{13}$$

where

$$\begin{aligned} \bar{\mathbf{y}} &= \begin{bmatrix} \Re\{\tilde{\mathbf{y}}\} \\ \Im\{\tilde{\mathbf{y}}\} \end{bmatrix}, \quad \bar{\mathbf{H}}_{\text{eff}} = \begin{bmatrix} \Re\{\mathbf{H}_{\text{eff}}\} & -\Im\{\mathbf{H}_{\text{eff}}\} \\ \Im\{\mathbf{H}_{\text{eff}}\} & \Re\{\mathbf{H}_{\text{eff}}\} \end{bmatrix}, \\ \bar{\mathbf{x}} &= \begin{bmatrix} \Re\{\mathbf{x}\} \\ \Im\{\mathbf{x}\} \end{bmatrix}, \quad \bar{\mathbf{w}} = \begin{bmatrix} \Re\{(\mathbf{F}_N \otimes \mathbf{G}_{\text{rx}})\mathbf{w}\} \\ \Im\{(\mathbf{F}_N \otimes \mathbf{G}_{\text{rx}})\mathbf{w}\} \end{bmatrix}, \end{aligned} \tag{14}$$

and $\Re\{\cdot\}$ and $\Im\{\cdot\}$ denote the real and imaginary parts, respectively.

III. AE BASED OTFS MODULATION AND DETECTION

From (3) and (13), we can have

$$\bar{\mathbf{y}} = \bar{\mathbf{H}}_{\text{eff}} \mathbf{f}(\mathbf{s}) + \bar{\mathbf{w}}, \tag{15}$$

where $\mathbf{s} = \text{vec}(\mathbf{S}) \in \mathbb{C}^{2MN\times 1}$. Here, the (m, n)th element of $\mathbf{S}$ is given by $s[m, n]$. That is, the message vector $\mathbf{s}$ can be mapped into the I-Q constellation vector through $\mathbf{f}(\cdot)$, which is now denoted as an encoder. Again, the elementwise QAM mapper is an example for $\mathbf{f}(\cdot)$.

A. DEEP LEARNING FOR AE BASED OTFS MODULATION AND DETECTION

To design the encoder, $\mathbf{f}(\cdot)$, we consider the auto-encoder techniques which are successfully applied to the transceiver design for wireless communication systems. Specifically, by denoting $\mathbf{g}(\cdot)$ as the decoder which is a pair of $\mathbf{f}(\cdot)$ for the auto-encoder, $\mathbf{f}(\cdot)$ and $\mathbf{g}(\cdot)$ are modeled by the non-linear FNN architectures. That is, they consist of fully connected linear layers, activation layers, and batch normalization layers. Then, $\mathbf{f}(\cdot)$ and $\mathbf{g}(\cdot)$ are, respectively, rewritten as $\mathbf{f}(\cdot; \boldsymbol{\theta}_{\text{e}})$ and $\mathbf{g}(\cdot; \boldsymbol{\theta}_{\text{d}})$ with the model parameters $\boldsymbol{\theta}_{\text{e}}$ and $\boldsymbol{\theta}_{\text{d}}$. In addition, for the effective training, $\mathbf{s}$ is transformed into a label one-hot encoded vector $\mathbf{1}_s \in \mathbb{R}^D$ at the transmitter. That is, $\mathbf{1}_s$ is a D dimensional vector, the s-th element of which is equal to one and zero otherwise. Then, (15) is given as

$$\bar{\mathbf{y}} = \bar{\mathbf{H}}_{\text{eff}} \mathbf{f}(\mathbf{1}_s; \boldsymbol{\theta}_{\text{e}}) + \bar{\mathbf{w}}. \tag{16}$$

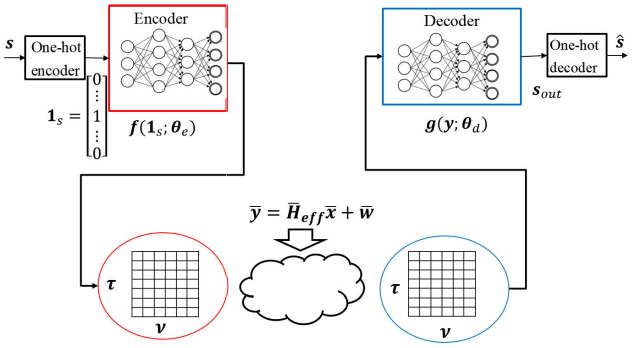

FIGURE 3. Pictorial description of the AE-based OTFS transceiver.

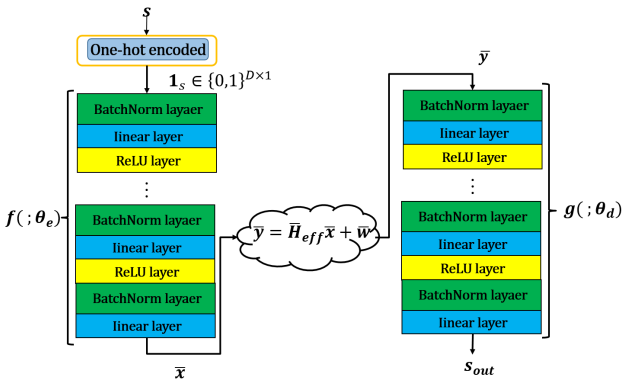

FIGURE 4. FNN module for the encoder and the decoder.

In the auto-encoder, $\mathbf{f}(\cdot;\boldsymbol{\theta}_{\rm e})$ and $\mathbf{g}(\cdot;\boldsymbol{\theta}_{\rm d})$ are jointly trained for the output of $\mathbf{g}(\bar{\mathbf{y}};\boldsymbol{\theta}_{\rm d})$ to generate the input signal $\mathbf{1}_s$ (equivalently, $\mathbf{s}$). That is, by letting

$$\mathbf{s}_{\rm out|\bar{\mathbf{H}}_{\rm eff}} \triangleq \mathbf{g}(\bar{\mathbf{H}}_{\rm eff}\mathbf{f}(\mathbf{1}_s;\boldsymbol{\theta}_{\rm e}) + \mathbf{w};\boldsymbol{\theta}_{\rm d}), \tag{17}$$

the model parameters ($\boldsymbol{\theta}_{\rm e}$ and $\boldsymbol{\theta}_{\rm d}$) are trained such as $\mathbf{s}_{\rm out} \approx \mathbf{1}_s$. The pictorial description of the AE-based OTFS transceiver is given in Fig. 3.

For the training, the cross entropy (CE) is then used as the loss function which is given as

$$\begin{aligned} L_{\rm CE}(\mathbf{s}_{out|\bar{\mathbf{H}}_{\rm eff}}, \mathbf{1}_s;\boldsymbol{\theta}_{\rm e},\boldsymbol{\theta}_{\rm d}) &= -\sum_{i=1}^{D} \mathbf{1}_s[i]\log(\mathbf{s}_{out|\bar{\mathbf{H}}_{\rm eff}}[i]) \\ &= -\log(\mathbf{s}_{out|\bar{\mathbf{H}}_{\rm eff}}[s]). \end{aligned} \tag{18}$$

Then, by using the training datasets with randomly selected $\mathbf{1}_s$, $s \in \{1,\ldots,D\}$, the network function parameter ($\boldsymbol{\theta}_{\rm e}$ and $\boldsymbol{\theta}_{\rm d}$) can be updated as

$$\begin{bmatrix} \boldsymbol{\theta}_{\rm e}^{(t)} \\ \boldsymbol{\theta}_{\rm d}^{(t)} \end{bmatrix} \leftarrow \begin{bmatrix} \boldsymbol{\theta}_{\rm e}^{(t-1)} \\ \boldsymbol{\theta}_{\rm d}^{(t-1)} \end{bmatrix} + \begin{bmatrix} \Delta\boldsymbol{\theta}_{{\rm e}|\bar{\mathbf{H}}_{\rm eff}}^{(t-1)} \\ \Delta\boldsymbol{\theta}_{{\rm d}|\bar{\mathbf{H}}_{\rm eff}}^{(t-1)} \end{bmatrix}. \tag{19}$$

where $\Delta\boldsymbol{\theta}_{{\rm x}|\bar{\mathbf{H}}_{\rm eff}}^{(t-1)}$ denotes the gradient such that the loss function is minimized with respect to the training datasets for a given effective channel $\bar{\mathbf{H}}_{\rm eff}$ and is given as

$$\Delta\boldsymbol{\theta}_{{\rm x}|\bar{\mathbf{H}}_{\rm eff}}^{(t-1)} = -\eta\nabla_{\boldsymbol{\theta}_{\rm x}} L_{\rm CE}(\mathbf{s}_{out|\bar{\mathbf{H}}_{\rm eff}}, \mathbf{1}_s;\boldsymbol{\theta}_{\rm e},\boldsymbol{\theta}_{\rm d})\Big|_{\boldsymbol{\theta}_{\rm x}=\boldsymbol{\theta}_{\rm x}^{(t-1)}}, \tag{20}$$

with a learning rate, η for ${\rm x} \in \{{\rm e},{\rm d}\}$.

B. FEED-FORWARD NEURAL NETWORK ARCHITECTURE OF ENCODER AND DECODER

Throughout the paper, a FNN module is considered for the encoder and decoder as in Fig. 4. That is, the non-linear neural network functions $\mathbf{f}(\cdot;\boldsymbol{\theta}_{\rm e})$ and $\mathbf{g}(\cdot;\boldsymbol{\theta}_{\rm d})$ are composed of batch normalization layers, linear layers, and activation layers, where each processing element (i.e., neuron) is connected only to the neurons in the following layer without any feedback loop and the input data propagates forward from the input layer to the output layer [37].

In the batch normalization layer, the covariance shift is calibrated to enhance the FNN training, which is accomplished by transforming the input vector $\mathbf{x}_i$ as

$$\mathbf{x}_o = \frac{\mathbf{x}_i - {\rm E}[\mathbf{x}_i]}{\sqrt{{\rm Var}[\mathbf{x}_i]}}, \tag{21}$$

where $E[\mathbf{x}_i]$ and ${\rm Var}[\mathbf{x}_i]$ are, respectively, the mean and the variance of $\mathbf{x}_i$ over the mini-batch.

The n-th output element of the l-th feed-forward layer, $x_n^{(l)}$, can be given as

$$x_n^{(l)} = \mathbf{w}_n^{(l)T}\mathbf{x}^{(l-1)} + b_n^{(l)}, \tag{22}$$

where $\mathbf{w}_n^{(l)}$ and $b_n^{(l)}$ are the weight vector and the bias for the n-th output element of the l-th feed-forward layer and the l-th output vector can be given as $\mathbf{x}^{(l)} = \left[x_1^{(l)}, x_2^{(l)}, \ldots, x_{N_l}^{(l)}\right]^T$, where N_l is the number of the output elements of the l-th layer.

In the activation layers, rectified linear unit (ReLU) function is used, which is given as

$$f_a(\mathbf{X}_{(i)}) = \max(\mathbf{0}, \mathbf{X}_{(i)}).$$

Note that, if $s[m,n]$ can be one of $M^{\rm c}$ message candidates, the dimension of one-hot encoded vector, D, is given as $D = (M^{\rm c})^{MN}$. Accordingly, the input size of the encoder also increases exponentially and the associated FNN archtecture becomes heavy and it is inappropriate for large M and N, which is a general case for OTFS modulation. For example, if $M^{\rm c} = 2$ (i.e., BPSK constellation for each grid point), $M = 8, N = 16, D = 2^{128}$.

IV. HIERARCHICAL STRUCTURE AND LEARNING STRATEGIES FOR AE BASED OTFS MODULATION AND DETECTION

In what follows, to overcome the large size of the input vector for the conventional FNN architecture, a new partitioned AE-based architecture with the multi-hot input vector is proposed motivated from the sparsity of the channel in delay-Doppler domain.

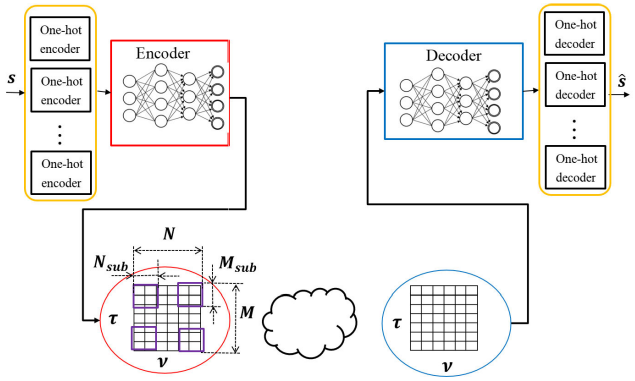

FIGURE 5. Pictorial description of the AE-based OTFS transceiver with the multi-hot input vector.

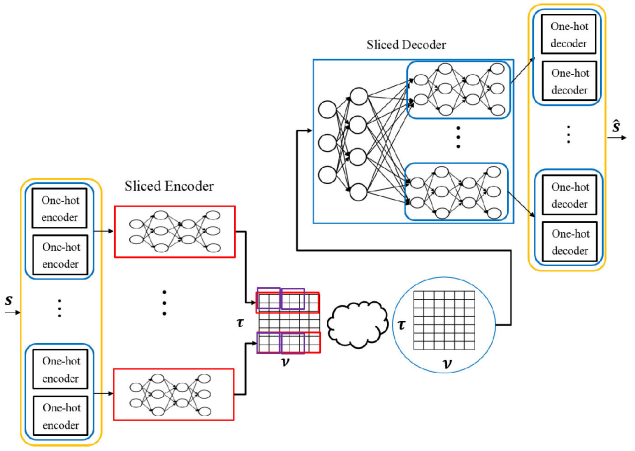

FIGURE 6. Pictorial description of the AE-based OTFS transceiver with the sliced FNN architecture.

A. MULTI-HOT ENCODED VECTOR

To reduce the input size of the encoder, we first partition the $M \times N$ grid into multiple $M_{\text{sub}} \times N_{\text{sub}}$ sub-grids as depicted in Fig. 5 and associate a D_{sub} dimensional one-hot vector with each $M_{\text{sub}} \times N_{\text{sub}}$ subgrid, where $M_{\text{sub}} = \frac{M}{P_1}$, $N_{\text{sub}} = \frac{N}{P_2}$, and $D_{\text{sub}} = (M^{\text{c}})^{M_{\text{sub}}N_{\text{sub}}}$. Here, P_1 and P_2 are properly selected such that M_{sub} and N_{sub} are forced to be integers. We then concatenate the multiple one-hot vectors, resulting in a multi-hot vector, and use it as the input vector of the encoder. The input size can be reduced from $(M^{\text{c}})^{MN}$ to $P_1P_2(M^{\text{c}})^{M_{\text{sub}}N_{\text{sub}}}$.

For example, let us consider that $M = 8$, $N = 16$, $P_1 = 4$, $P_2 = 4$, and $M^{\text{c}} = 2$. Then, the one-hot input vector size for the AE without the grid partitioning is given as 2^{128}, while the multi-hot input vector size with the partitioned subgrids is $4 \times 4 \times 2^8 = 4096$.

Remark 1: We note that, by increasing P_1 and P_2, we can reduce the input size of the encoder, but the reduced dimension of the partitioned subblocks in delay-Doppler domain may decrease the adaptability of the output waveform on the channel. That is, when the dimension of the subblocks are small compared to the delay spread and the Doppler spread, the inter-grid-point interference in (9) cannot be effectively handled. Therefore, the tradeoff between the detection performance and the computational complexity can be expected.

B. PARTITIONED AE-BASED ARCHITECTURE

To further reduce the complexity of the AE architecture, the encoder can be partitioned into multiple sub-encoders (or, sliced encoders) as Fig. 6, then each sub-encoders can take the multi-hot encoded vector with a smaller dimension. As a specific example, we assume that the encoder has L layers, each with N_{In} node elements. Then, from (22), the number of elements in the weight vectors for L layers is given as $L \times N_{\text{In}}^2$. When the encoder is partitioned into K_{sub} sliced-encoders with L layers. Then, each layer can have $\frac{N_{\text{In}}}{K_{\text{sub}}}$ node elements. Therefore, the number of elements in the weight vectors for the sliced encoder is given as $LK_{\text{sub}} \times \left(\frac{N_{\text{In}}}{K_{\text{sub}}}\right)^2 \left(= L \times \frac{N_{\text{In}}^2}{K_{\text{sub}}}\right)$. That is, the required weights can be reduced by the factor of K_{sub} by exploiting the partitioned AE-based architecture. Likewise, the decoder can be sliced as in Fig. 6.

C. META-LEARNING STRATEGY FOR AE-BASED OTFS MODULATION AND DETECTION

We note that the training based on (19) and (20) is valid only for a given $\bar{\mathbf{H}}_{\text{eff}}$. Accordingly, when the channel is changed, the transceiver should start over the training process. To increase the adaptability to the channel variation, we have also developed meta-learning strategy for AE-based OTFS modulation and detection. We note that, in MAML, initial parameters of the model for fast adaptation is learned by alternating between inner-task procedure and cross-task procedure [35]. Likewise, for AE-based OTFS modulation and detection, the initialization parameters $\boldsymbol{\theta}_{\text{x}}^{(0)}$ are trained so that the optimal parameters $\Delta\boldsymbol{\theta}_{\text{x}|\bar{\mathbf{H}}_{\text{eff}}}^{(t)}$ can be obtained from them with a small number of epochs for any given $\mathbf{H}_{\text{eff}}$. That is, $\boldsymbol{\theta}_{\text{x}}^{(0)}$ can be optimized such that

$$\begin{aligned}\min_{\boldsymbol{\theta}_{\text{x}}^{(0)}} L_{\text{CE}}^{\text{MAML}}(\boldsymbol{\theta}_{\text{e}}^{(0)}, \boldsymbol{\theta}_{\text{d}}^{(0)}) = \min_{\boldsymbol{\theta}_{\text{x}}^{(0)}} \textstyle\sum_{\bar{\mathbf{H}}_{\text{eff}} \in \mathbb{H}} L_{\text{CE}}(\mathbf{s}_{out|\bar{\mathbf{H}}_{\text{eff}}}, \mathbf{1}_s; \\ \boldsymbol{\theta}_{\text{e}}^{(0)} - \eta\Delta\boldsymbol{\theta}_{\text{e}|\bar{\mathbf{H}}_{\text{eff}}}^{(0)}, \\ \boldsymbol{\theta}_{\text{d}}^{(0)} - \eta\Delta\boldsymbol{\theta}_{\text{d}|\bar{\mathbf{H}}_{\text{eff}}}^{(0)}),\end{aligned} \tag{23}$$

where $\mathbb{H}$ is a set of possible effective channel matrices sampled from a given channel parameter distribution.

Then, by defining the locally updated parameter for $\bar{\mathbf{H}}_{\text{eff}}$ as

$$U_{\bar{\mathbf{H}}_{\text{eff}}}(\boldsymbol{\theta}_{\text{x}}^{(0)}) = \boldsymbol{\theta}_{\text{x}}^{(0)} - \eta\Delta\boldsymbol{\theta}_{\text{x}|\bar{\mathbf{H}}_{\text{eff}}}^{(0)}, \tag{24}$$

the initialization parameters can be updated as [32] and [34]

$$\begin{aligned}\boldsymbol{\theta}_{\text{x}}^{(0,t)} \leftarrow \boldsymbol{\theta}_{\text{x}}^{(0,t-1)} - \kappa \sum_{\bar{\mathbf{H}}_{\text{eff}} \in \mathbb{H}} \left(\mathbf{J}_{\boldsymbol{\theta}_{\text{x}}^{(0)}} U_{\bar{\mathbf{H}}_{\text{eff}}}(\boldsymbol{\theta}_{\text{x}}^{(0,i)})\right) \\ \cdot \nabla_{\text{x}} L_{\text{CE}}\left(\mathbf{s}_{out|\bar{\mathbf{H}}_{\text{eff}}}, \mathbf{1}_s; U_{\bar{\mathbf{H}}_{\text{eff}}}(\boldsymbol{\theta}_{\text{e}}^{(0,i)}), U_{\bar{\mathbf{H}}_{\text{eff}}}(\boldsymbol{\theta}_{\text{d}}^{(0,i)})\right),\end{aligned} \tag{25}$$

Algorithm 1 Meta-Learning Algorithm for AE Based OTFS Modulation and Detection

1: Initialize parameter vectors, $\boldsymbol{\theta}_{\mathrm{e}}^{(0,0)}$ and $\boldsymbol{\theta}_{\mathrm{d}}^{(0,0)}$
2: **for** $t = 1, \ldots, T_{\max}$ **do**
3: **for** each $\bar{\mathbf{H}}_{\mathrm{eff}} \in \mathbb{H}$ **do**
4: Compute local update as

$$U_{\bar{\mathbf{H}}_{\mathrm{eff}}}(\boldsymbol{\theta}_{\mathrm{x}}^{(0,i)}) = \boldsymbol{\theta}_{\mathrm{x}}^{(0,i)} - \eta \Delta \boldsymbol{\theta}_{\mathrm{x}|\bar{\mathbf{H}}_{\mathrm{eff}}}^{(0,i)}, \quad \text{for x} \in \{\mathrm{e}, \mathrm{d}\}$$

5: **end for**
6: $\boldsymbol{\theta}_{\mathrm{x}}^{(0,t)} \leftarrow \boldsymbol{\theta}_{\mathrm{x}}^{(0,t-1)} - \kappa \nabla_{\mathrm{x}} L_{\mathrm{CE}}^{\mathrm{MAML}}(\boldsymbol{\theta}_{\mathrm{e}}^{(0)}, \boldsymbol{\theta}_{\mathrm{d}}^{(0)})$
7: **end for**

TABLE 1. AE-based OTFS transceiver parameter.

Parameter	Value
Number of frequency bands (M)	16
Number of symbols (N)	8
Numbers of partitions (P_1, P_2)	$(4, 4)$
Size of subblocks ($M_{\mathrm{sub}} \times N_{\mathrm{sub}}$)	4×2
Carrier frequency	5.89GHz
Bandwidth	$10MHz$
Optimizer	ADAM [38]
Learning rate (η)	0.001

where $\mathbf{J}_{\boldsymbol{\theta}_{\mathrm{x}}^{(0)}} U_{\bar{\mathbf{H}}_{\mathrm{eff}}}(\boldsymbol{\theta}_{\mathrm{x}}^{(0,i)})$ is Jaccobian of $U_{\bar{\mathbf{H}}_{\mathrm{eff}}}(\boldsymbol{\theta}_{\mathrm{x}}^{(0,i)})$ with respect to $\boldsymbol{\theta}_{\mathrm{x}}^{(0)}$. The associated meta-learning algorithm for AE based OTFS modulation and detection can be simply summarized in Algorithm 1.

V. SIMULATION RESULTS

Through the computer simulations, we validate the performance of the proposed AE-based OTFS transceiver. The detailed simulation parameters are summarized in Table 1. For the weight updating, ADAM optimizer [38] is adopted with a learning rate 0.0005. The proposed AE-based OTFS transceiver is implemented by using Python with libraries of PyTorch 1.10.1. In addition, the simulations are carried out with AMD Ryzen 7 5800X 8-Core Processor 3.80 GHz and 48GB RAM. The training SNR is set as 16 dB. Here, h_i follows circularly symmetric complex Gaussian distribution with zero-mean and unit variance. In addition, l_i and k_i are uniformly distributed on $[0, M_{\tau_{\max}} - 1]$ and $[0, N_{\nu_{\max}} - 1]$.

In Fig. 7, the average cross entropy loss curves are evaluated during the training phase for the proposed AE-based OTFS transceiver with $P = \{2, 3, 4\}$. Here, the channel profile is randomly generated but fixed during the training and validation phases and each OTFS waveform carries 128 bits (i.e., $K = 128$). That is, 8 bits are carried on each subblock. In addition, $M_{\tau_{\max}} = \frac{M}{2} = 8$ and $N_{\nu_{\max}} = \frac{N}{2} = 4$. Note that, without the proposed hierarchical structure, the one-hot encoded vector size should be 2^{128}, but it can be reduced up to 4096 ($= 16 \times 2^8$) with the proposed partitioned AE-based architecture. The loss curves are averaged over 100 randomly generated channels. From the figure, as the number of iterations (i.e., epochs) increases, the overall loss decreases. After 5000 epochs, it is stuck in a local optimum, resulting in the increase of the loss, but after a few iterations, it came out of the local optimum. In Fig. 8, we also evaluate the BER performance of the proposed AE based OTFS transceiver during training phase. Again, it can be found that, as the number of iterations increases, the overall BER decreases.

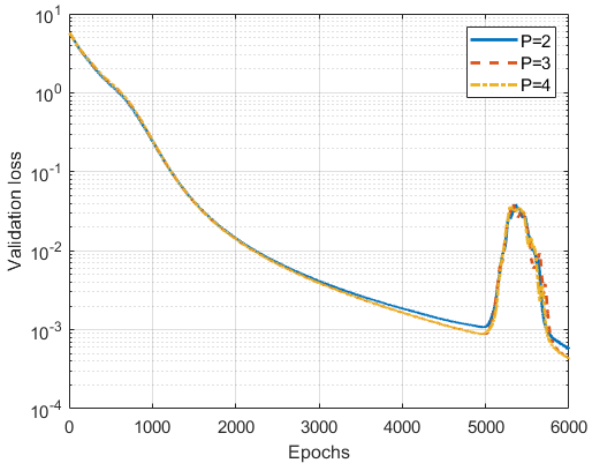

FIGURE 7. Loss curves for AE-based OTFS transceiver with $P = \{2, 3, 4\}$.

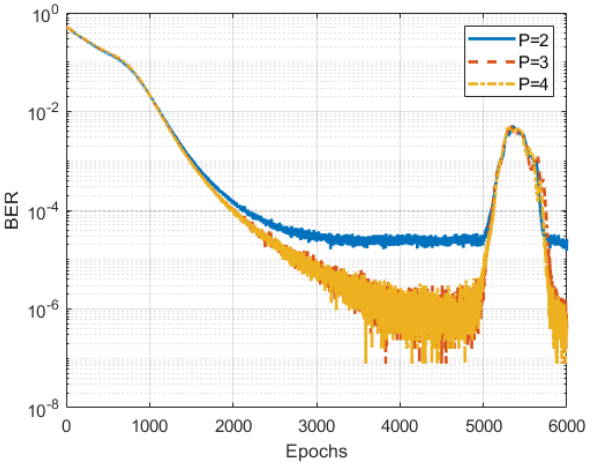

FIGURE 8. BER performances during training phase for AE-based OTFS transceiver with $P = \{2, 3, 4\}$.

In Fig. 9, we also evaluate the BER performance of the proposed AE-based OTFS transceiver for various SNRs. For comparison purpose, the BER of the MP-based detection method [10] is also evaluated. It can be found that the proposed AE-based OTFS transceiver outperforms the conventional OTFS waveform and MP-based detection especially at high SNRs. Interestingly, as the number of multipaths (i.e., P) increases, BERs are improved because more diversity can be achieved at the time-frequency domain. In Fig. 10, BER performance for three different transceivers (i.e., the AE-based OTFS transceiver, the conventional OTFS waveform with MP-based detection [10], and the conventional OTFS waveform with LMMSE detection [16]) when $P = 3$. It can be found that the proposed scheme outperforms

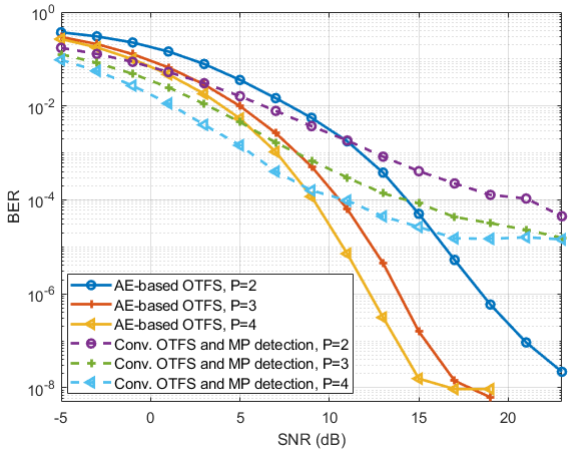

FIGURE 9. BER performances versus SNR for the AE-based OTFS transceiver and the conventional OTFS waveform and MP-based detection with $P = \{2, 3, 4\}$.

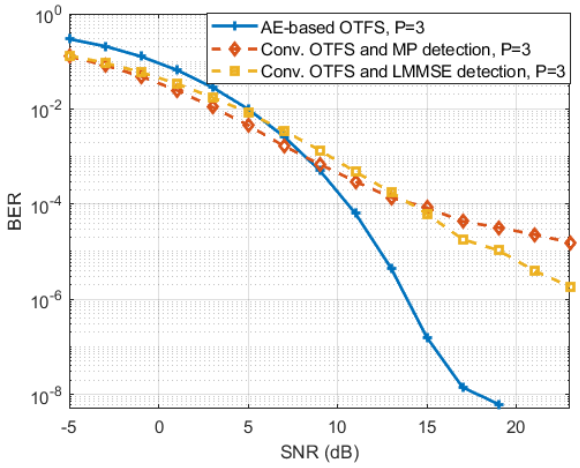

FIGURE 10. BER performances versus SNR for three different transceivers (i.e., the AE-based OTFS transceiver, the conventional OTFS waveform with MP-based detection, and the conventional OTFS waveform with LMMSE detection) when $P = 3$.

the other two schemes. That is, in the proposed AE-based OTFS transceiver, the transmitter and the receiver are jointly optimized to minimize the reconstruction loss, that is relevant to the BER, through an end-to-end approach by jointly training the encoder and the decoder. Accordingly, in the proposed AE-based OTFS transceiver, the OTFS waveform and decision region are jointly adopted to channel effect and the channel coding gain can be further achieved.

In Fig. 11, we evaluate the BER performance for different sizes of subblock, $(M_{\text{sub}}, N_{\text{sub}}) \in \{(2, 2), (2, 1), (1, 1)\}$. Here, eacch OTFS waveform carries 256 bits (i.e., $K = 256$). Accordingly, the input vector sizes for $(M_{\text{sub}}, N_{\text{sub}}) \in \{(2, 2), (2, 1), (1, 1)\}$ are, respectively, given as $D_{\text{In}} \in \{32 \times 2^8,\ 64 \times 2^4,\ 128 \times 2^2\} = \{8192, 1024, 512\}$. For the channel profile, $M_{\tau_{\max}} = \frac{M}{4} = 4$ and $N_{\nu_{\max}} = \frac{N}{4} = 2$. From the figure, the AE-based OTFS with $(M_{\text{sub}}, N_{\text{sub}}) = (1, 1)$ exhibits the worst performance, while those with $(M_{\text{sub}}, N_{\text{sub}}) = \{(2, 2), (2, 1)\}$ show a similar BER performance, 10^{-6} at SNR$=$ 20 dB. That is, the AE-based OTFS with $(M_{\text{sub}}, N_{\text{sub}}) = (1, 1)$ cannot generate the OTFS waveform to cope with a given channel fading and, to achieve the adaptability of the OTFS output waveform on the channel, the block size should be increased.

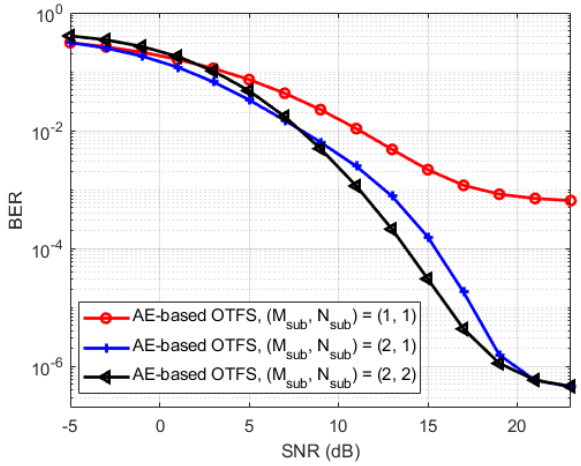

FIGURE 11. BER performances versus SNR for different sizes of subblock, $(M_{\text{sub}}, N_{\text{sub}}) \in \{(2, 2), (2, 1), (1, 1)\}$.

To validate the training strategy for the proposed AE-based OTFS transceiver, when a new channel profile is generated, the training convergence speed has been evaluated. That is, the validation loss and the BER performance are evaluated over epochs. For comparison purpose, performances of the conventional learning (i.e., training for a single task with a default parameter initialization) and the joint learning scheme [40] are also evaluated. In Fig. 12, the cross entropy loss curves are evaluated during the training phase with newly updated channel profile. Here, each OTFS waveform carries 128 bits (i.e., $K = 128$). In addition, $M_{\tau_{\max}} = \frac{M}{2} = 8$ and $N_{\nu_{\max}} = \frac{N}{2} = 4$. From the figure, the validation loss with the conventional training decreases slowly, but those with the joint-learning and the meta-learning (i.e., MAML) converge to less than 10^{-2} after 100 epochs. Furthermore, the meta-learning shows lower validation loss than the joint learning. In Fig. 13, we also compare the BER performances during training phase. Again, it can be found that, it takes about 100 epochs to reach the BER of 10^{-3} for both joint-learning and meta-learning methods, while the conventional learning exhibits BER higher than 0.1 even after 300 epochs. Again, the meta-learning shows lower BER performance than the joint-learning.

In Table 2, we compare the computational complexity of the proposed scheme with other existing works - LMMSE based detection [16], [39] and MP based detection [10]. In Table 2, n_{iter} is the number of iterations in MP detector. From [10], n_{iter} is typically given by 20 or less. In addition, L_{dec} is the number of layers at the decoder and it is set as 5 in our simulations. The parameter N_{in} is set as $P_1 P_2 (M^{\text{c}})^{M_{\text{sub}} N_{\text{sub}}}$

TABLE 2. Computational complexity comparison (in real-valued flops).

Method	The number of floating operations		From Table I (for a newly updated channel)
	Training phase	Evaluating phase	
LMMSE receiver [16], [39]		$O((2M)^3(2N)^3)$	$(2M)^3(2N)^3 = 1.34 \times 10^6$
MP receiver [10]		$O(n_{\text{iter}}(2M)(2N)(2P)M^{\text{c}})$	$n_{\text{iter}}(2M)(2N)(2P)M^{\text{c}} = 8.19 \times 10^3 n_{\text{iter}}$
Proposed AE-based receiver	$O(n_{\text{epoch}} L_{\text{dec}} \frac{N_{\text{in}}^2}{K_{\text{sub}}})$	$O(L_{\text{dec}} \frac{N_{\text{in}}^2}{K_{\text{sub}}})$	$n_{\text{epoch}} L_{\text{dec}} \frac{N_{\text{in}}^2}{K_{\text{sub}}} = 5.12 \times 10^3 n_{\text{epoch}}$ $L_{\text{dec}} \frac{N_{\text{in}}^2}{K_{\text{sub}}} = 5.12 \times 10^3$
Proposed AE-based receiver with MAML	$O(n_{\text{epoch}}^{MAML} L_{\text{dec}} \frac{N_{\text{in}}^2}{K_{\text{sub}}})$ $O(n_{\text{epoch}}^{\text{up}} L_{\text{dec}} \frac{N_{\text{in}}^2}{K_{\text{sub}}})$	$O(L_{\text{dec}} \frac{N_{\text{in}}^2}{K_{\text{sub}}})$	$n_{\text{epoch}}^{\text{up}} L_{\text{dec}} \frac{N_{\text{in}}^2}{K_{\text{sub}}} = 5.12 \times 10^3 n_{\text{epoch}}^{\text{up}}$ $L_{\text{dec}} \frac{N_{\text{in}}^2}{K_{\text{sub}}} = 5.12 \times 10^3$

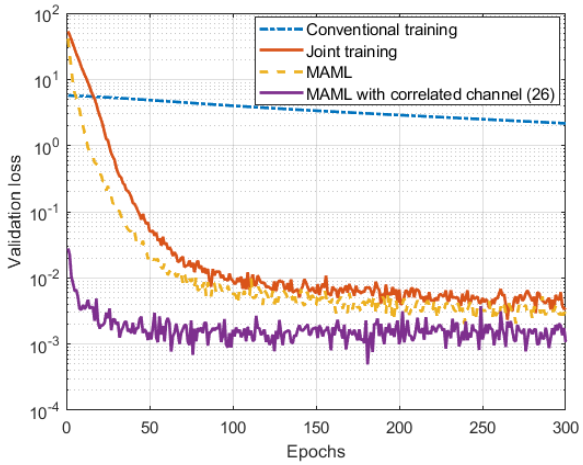

FIGURE 12. Loss curves for various training methods.

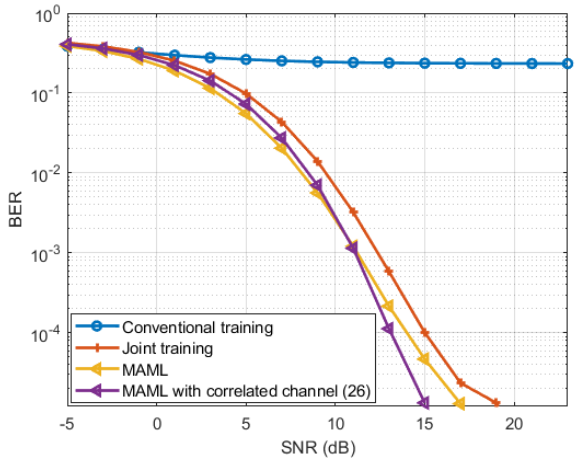

FIGURE 14. BER performances versus SNR for various training methods.

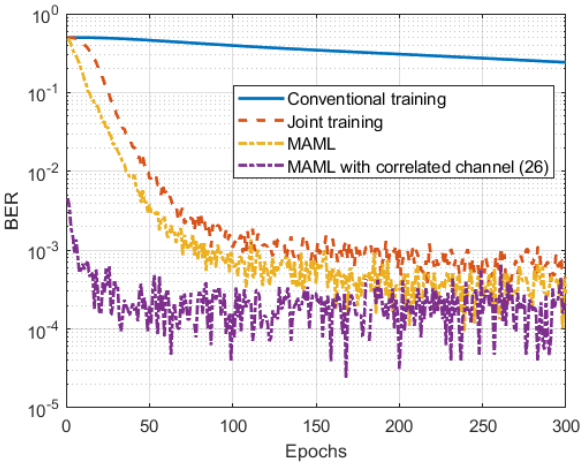

FIGURE 13. BER performances during training phase for various training methods.

and from Table 1, $N_{\text{in}} = 4 \times 4 \times (2)^8$. We note that the computational complexity of the AE-based OTFS transceiver is categorized into two parts - training and evaluating parts. In contrast, the conventional LMMSE or MP receivers do not require the training phase.

We also note that the computational complexity for one epoch in training phase of the proposed AE-based receiver is comparable to that for the MP receiver, which is less than the LMMSE receiver. However, the conventional training method requires several thousands of epochs (iterations) for a newly updated channel. In the proposed AE-based OTFS transceiver with meta-learning, the training can be done in the off-line with synthetic generated data and the required epochs for the training with randomly generated channel (denoted as $n_{\text{epoch}}^{\text{up}}$) can be drastically reduced. Specifically, in Fig. 8, the conventional training method requires over 6000 iterations (i.e., $n_{\text{epoch}} = 6000$ in Table 2) for the training. In contrast, when the MAML is used, it takes about 100 iterations (i.e., $n_{\text{epoch}}^{\text{up}} = 100$ in Table 2) for the training with randomly generated channel having no correlation with the channel used in the training period.

If the channels for the consecutive frames are correlated, the training iterations can be further reduced. Specifically, in Fig. 13, the BER performances during training phase is additionally updated when the channel is updated as

$$\mathbf{H}_{\text{new}} = \sum_{i=1}^{P} (h_{\text{prev},i} + \Delta h_i) \mathbf{\Pi}^{l_i} \mathbf{\Delta}^{k_i}, \tag{26}$$

where Δh_i follows circularly symmetric complex Gaussian distribution with zero-mean and variance of 0.01. It can be found that when the updated channel is correlated with the

previous trained channel, it takes less than 25 epochs for convergence.

In Fig. 14, we evaluate the BER performance versus SNR for various training methods. Here, the number of epochs for the training is limited as 300. It can be found that the Meta-learning outperforms the other schemes and due to the lack of training iterations, the conventional training method cannot train the AE-based transceiver efficiently. Specifically, through the meta-learning, the proposed AE-based OTFS modulation and detection can be trained much faster than the conventional approach, requiring less than 100 iterations for a newly generated random channel impulse response.

VI. CONCLUSION

In this paper, the AE-based OTFS modulation and detection scheme are developed, where the transmit OTFS waveform and its associated detection scheme at the receiver is jointly optimized through the end-to-end learning strategy. If the conventional FNN architecture is deployed for the AE-based OTFS modulation and detection, its input size grows exponentially with respect to the number of grid points in 2D delay-Doppler domain. To overcome it, the hierarchical structure for AE-based OTFS modulation and detection is proposed, where the 2D delay-Doppler grid is divided into multiple subblocks and an one-hot encoded vector is associated to each subblock. The multi-hot vector is formed and exploited as the input vector for the proposed AE-based OTFS modulation and detection, allowing a reasonable input size for AE-based OTFS modulation and detection. Finally, we also develop the meta-learning strategy to adapt the proposed AE-based OTFS modulation and detection scheme to the channel variations. Through the simulation, the proposed OTFS modulation and detection outperforms the conventional OTFS waveform with the MP-based detection at high SNRs. In addition, through the meta-learning, the proposed AE-based OTFS modulation and detection can be trained much faster than the conventional approach, requiring less than 100 iterations for a newly generated random channel impulse response.

REFERENCES

[1] B. Farhang-Boroujeny and H. Moradi, "OFDM inspired waveforms for 5G," *IEEE Commun. Surveys Tuts.*, vol. 18, no. 4, pp. 2474–2492, 4th Quart., 2016.

[2] P. Guan, D. Wu, T. Tian, J. Zhou, X. Zhang, L. Gu, A. Benjebbour, M. Iwabuchi, and Y. Kishiyama, "5G field trials: OFDM-based waveforms and mixed numerologies," *IEEE J. Sel. Areas Commun.*, vol. 35, no. 6, pp. 1234–1243, Jun. 2017.

[3] M. H. C. Garcia, A. Molina-Galan, M. Boban, J. Gozalvez, B. Coll-Perales, T. Sahin, and A. Kousaridas, "A tutorial on 5G NR V2X communications," *IEEE Commun. Surveys Tuts.*, vol. 23, no. 3, pp. 1972–2026, 3rd Quart., 2021.

[4] C. Zhang and W. Zhang, "Spectrum sharing for drone networks," *IEEE J. Sel. Areas Commun.*, vol. 35, no. 1, pp. 136–144, Jan. 2017.

[5] T. Hou, Y. Liu, Z. Song, X. Sun, and Y. Chen, "UAV-to-everything (U2X) networks relying on NOMA: A stochastic geometry model," *IEEE Trans. Veh. Technol.*, vol. 69, no. 7, pp. 7558–7568, Jul. 2020.

[6] Y. Zeng, Q. Wu, and R. Zhang, "Accessing from the sky: A tutorial on UAV communications for 5G and beyond," *Proc. IEEE*, vol. 107, no. 12, pp. 2327–2375, Dec. 2019.

[7] P. Raviteja, K. T. Phan, Y. Hong, and E. Viterbo, "Orthogonal time frequency space (OTFS) modulation based radar system," in *Proc. IEEE Radar Conf. (RadarConf)*, Apr. 2019, pp. 1–6.

[8] M. S. Khan, Y. J. Kim, Q. Sultan, J. Joung, and Y. S. Cho, "Downlink synchronization for OTFS-based cellular systems in high Doppler environments," *IEEE Access*, vol. 9, pp. 73575–73589, 2021.

[9] Z. Wei, W. Yuan, S. Li, J. Yuan, G. Bharatula, R. Hadani, and L. Hanzo, "Orthogonal time-frequency space modulation: A promising next-generation waveform," *IEEE Wireless Commun. Mag.*, vol. 28, no. 4, pp. 136–144, Aug. 2021.

[10] P. Raviteja, K. T. Phan, Y. Hong, and E. Viterbo, "Interference cancellation and iterative detection for orthogonal time frequency space modulation," *IEEE Trans. Wireless Commun.*, vol. 17, no. 10, pp. 6501–6515, Oct. 2018.

[11] A. Farhang and B. Farhang-Boroujeny, "Orthogonal time-frequency space modulation: Principles and implementation," in *Radio Access Networks Slicing and Virtualization for 5G Vertical Industries*. Hoboken, NJ, USA: Wiley, 2021, ch. 6, pp. 103–120.

[12] S. S. Das and R. Prasad, *Orthogonal Time Frequency Space Modulation: OTFS a Waveform for 6G*. River Publishers, 2021.

[13] R. Hadani, S. Rakib, S. Kons, M. Tsatsanis, A. Monk, C. Ibars, J. Delfeld, Y. Hebron, A. J. Goldsmith, A. F. Molisch, and R. Calderbank, "Orthogonal time frequency space modulation," 2018, *arXiv:1808.00519*.

[14] P. Raviteja, K. T. Phan, and Y. Hong, "Embedded pilot-aided channel estimation for OTFS in delay–Doppler channels," *IEEE Trans. Veh. Technol.*, vol. 68, no. 5, pp. 4906–4917, May 2019.

[15] Y. Liu, Y. L. Guan, and D. González, "Near-optimal BEM OTFS receiver with low pilot overhead for high-mobility communications," *IEEE Trans. Commun.*, vol. 70, no. 5, pp. 3392–3406, May 2022.

[16] R. Hadani and A. Monk, "OTFS: A new generation of modulation addressing the challenges of 5G," 2018, *arXiv:1802.02623*.

[17] S. Tiwari, S. S. Das, and V. Rangamgari, "Low complexity LMMSE receiver for OTFS," *IEEE Commun. Lett.*, vol. 23, no. 12, pp. 2205–2209, Dec. 2019.

[18] Z. Gong, S. Liu, and Y. Huang, "Doppler diversity reception for OTFS modulation," in *Proc. IEEE 95th Veh. Technol. Conf. (VTC-Spring)*, Jun. 2022, pp. 1–5.

[19] G. D. Surabhi and A. Chockalingam, "Low-complexity linear equalization for 2×2 MIMO-OTFS signals," in *Proc. IEEE 21st Int. Workshop Signal Process. Adv. Wireless Commun. (SPAWC)*, May 2020, pp. 1–5.

[20] M. K. Ramachandran and A. Chockalingam, "MIMO-OTFS in high-Doppler fading channels: Signal detection and channel estimation," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, Dec. 2018, pp. 206–212.

[21] V. Khammammetti and S. K. Mohammed, "OTFS-based multiple-access in high Doppler and delay spread wireless channels," *IEEE Wireless Commun. Lett.*, vol. 8, no. 2, pp. 528–531, Apr. 2019.

[22] R. M. Augustine and A. Chockalingam, "Interleaved time-frequency multiple access using OTFS modulation," in *Proc. IEEE 90th Veh. Technol. Conf. (VTC-Fall)*, Sep. 2019, pp. 1–5.

[23] S. Gao and J. Zheng, "Clipping noise recovery of OTFS signals using reliability-based compressed sensing," *IEEE Trans. Veh. Technol.*, vol. 70, no. 7, pp. 7187–7192, Jul. 2021.

[24] O. K. Rasheed, G. D. Surabhi, and A. Chockalingam, "Sparse delay-Doppler channel estimation in rapidly time-varying channels for multiuser OTFS on the uplink," in *Proc. IEEE 91st Veh. Technol. Conf. (VTC-Spring)*, May 2020, pp. 1–5.

[25] T. J. O'Shea and J. Hoydis, "An introduction to deep learning for the physical layer," *IEEE Trans. Cogn. Commun. Netw.*, vol. 3, no. 4, pp. 563–575, Oct. 2017.

[26] T. Erpek, T. J. O'Shea, and T. C. Clancy, "Learning a physical layer scheme for the MIMO interference channel," in *Proc. IEEE Int. Conf. Commun. (ICC)*, May 2018, pp. 1–5.

[27] D. Wu, M. Nekovee, and Y. Wang, "Deep learning-based autoencoder for m-user wireless interference channel physical layer design," *IEEE Access*, vol. 8, pp. 174679–174691, 2020.

[28] J. Tao, J. Chen, J. Xing, S. Fu, and J. Xie, "Autoencoder neural network based intelligent hybrid beamforming design for mmWave massive MIMO systems," *IEEE Trans. Cogn. Commun. Netw.*, vol. 6, no. 3, pp. 1019–1030, Sep. 2020.

[29] J.-M. Kang, C.-J. Chun, and I.-M. Kim, "Deep learning based channel estimation for MIMO systems with received SNR feedback," *IEEE Access*, vol. 8, pp. 121162–121181, 2020.

[30] B. Lin, X. Wang, W. Yuan, and N. Wu, "A novel OFDM autoencoder featuring CNN-based channel estimation for Internet of Vessels," *IEEE Internet Things J.*, vol. 7, no. 8, pp. 7601–7611, Aug. 2020.
[31] A. Felix, S. Cammerer, S. Dörner, J. Hoydis, and S. T. Brink, "OFDM-autoencoder for end-to-end learning of communications systems," in *Proc. IEEE 19th Int. Workshop Signal Process. Adv. Wireless Commun. (SPAWC)*, Jun. 2018, pp. 1–5.
[32] C. Finn, P. Abbeel, and S. Levine, "Model-agnostic meta-learning for fast adaptation of deep networks," in *Proc. 34th Int. Conf. Mach. Learn. (ICML)*, vol. 70, 2017, pp. 1126–1135.
[33] A. Nichol, J. Achiam, and J. Schulman, "On first-order meta-learning algorithms," 2018, *arXiv:1803.02999*.
[34] S. Park, O. Simeone, and J. Kang, "End-to-end fast training of communication links without a channel model via online meta-learning," in *Proc. IEEE 21st Int. Workshop Signal Process. Adv. Wireless Commun. (SPAWC)*, May 2020, pp. 1–5.
[35] Y. Yuan, G. Zheng, K.-K. Wong, B. Ottersten, and Z.-Q. Luo, "Transfer learning and meta learning-based fast downlink beamforming adaptation," *IEEE Trans. Wireless Commun.*, vol. 20, no. 3, pp. 1742–1755, Mar. 2021.
[36] P. Raviteja, Y. Hong, E. Viterbo, and E. Biglieri, "Practical pulse-shaping waveforms for reduced-cyclic-prefix OTFS," *IEEE Trans. Veh. Tech.*, vol. 68, no. 1, pp. 957–961, Jan. 2019.
[37] M. Chen, U. Challita, W. Saad, C. Yin, and M. Debbah, "Artificial neural networks-based machine learning for wireless networks: A tutorial," 2017, *arXiv:1710.02913*.
[38] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*.
[39] Y. Jiang, M. K. Varanasi, and J. Li, "Performance analysis of ZF and MMSE equalizers for MIMO systems: An in-depth study of the high SNR regime," *IEEE Trans. Inf. Theory*, vol. 57, no. 4, pp. 2008–2026, Apr. 2011.
[40] S. Park, H. Jang, O. Simeone, and J. Kang, "Learning to demodulate from few pilots via offline and online meta-learning," *IEEE Trans. Signal Process.*, vol. 69, pp. 226–239, 2021.

JAEHYUN PARK (Member, IEEE) received the B.S. and Ph.D. (M.S. and Ph.D. joint program) degrees in electrical engineering from the Korea Advanced Institute of Science and Technology (KAIST), in 2003 and 2010, respectively. From 2010 to 2013, he was a Senior Researcher with the Electronics and Telecommunications Research Institute (ETRI), where he was involved in transceiver design and spectrum sensing for cognitive radio systems. From 2013 to 2014, he was a Postdoctoral Research Associate with the Department of Electrical and Electronic Engineering, Imperial College London. He is currently a Professor with the Department of Electronic Engineering, Pukyong National University, South Korea. His research interests include signal processing for wireless communications and radar systems, with focus on detection and estimation for MIMO systems, MIMO radar, cognitive radio networks, and joint information and energy transfer.

JUN-PYO HONG (Member, IEEE) received the B.Sc. degree in electrical engineering from the Information and Communications University, Daejeon, South Korea, in 2008, and the M.S. and Ph.D. degrees in electrical engineering from the Korea Advanced Institute of Science and Technology, Daejeon, in 2010 and 2014, respectively. In 2015, he was a Researcher with the Electronics and Telecommunications Research Institute, where he was involved in RAN architecture for mobile communication systems. He is currently an Associate Professor with the Department of Information and Communications Engineering, Pukyong National University, Busan, South Korea.

HYUNGJU KIM (Member, IEEE) received the B.S. degree in electronics engineering from Kyungpook National University, Daegu, South Korea, in 2010, and the M.S. and Ph.D. degrees in electrical engineering from the Korea Advanced Institute of Science and Technology (KAIST), Daejeon, South Korea, in 2012 and 2018, respectively. Since 2018, he has been a Senior Researcher with the Electronics and Telecommunications Research Institute (ETRI), Daejeon. His current research interests include EM numerical analysis and radar signal processing.

BYUNG JANG JEONG (Member, IEEE) received the B.S. degree from Kyungpook National University, Daegu, South Korea, in 1988, and the M.Sc. and Ph.D. degrees in electrical engineering from the Korea Advanced Institute of Science and Technology (KAIST), in 1992 and 1997, respectively. He was with the Samsung Advanced Institute of Technology (SAIT), from 1994 to 2003. Since 2003, he has been a Principal Member of Research Staff with the Electronics and Telecommunications Research Institute (ETRI). His research interests include signal processing for wireless communications, MIMO systems, cognitive radio networks, and radar signal processing.