HIENet: A Hardware Impairment Estimation Network for OFDM Systems

Siqi Liu, Tianyu Wang, and Shaowei Wang

School of Electronic Science and Engineering, Nanjing University, Nanjing 210023, China
Email: dg1823047@smail.nju.edu.cn, {tianyu.alex.wang, wangsw}@nju.edu.cn

GLOBECOM 2022 - 2022 IEEE Global Communications Conference | 978-1-6654-3540-6/22/$31.00 ©2022 IEEE | DOI: 10.1109/GLOBECOM48099.2022.10000792

***Abstract*—In orthogonal frequency division multiplexing (OFDM) systems, carrier frequency offset (CFO) and in- and quadrature-phase (IQ) imbalance are two critical hardware impairments that may lead to severe amplitude and phase mismatches and therefore greatly degrade the demodulation performance. Due to the noise impact and the coupling effects between different hardware impairments, it is considered to be highly challenging to perform joint estimation of CFO and IQ imbalance. In this paper, we propose a novel multi-task learning-based hardware impairment estimation network (HIENet) to simultaneously estimate the CFO and IQ imbalance in OFDM systems. The proposed HIENet can address the noise impact by averaging the task-dependent noise patterns and overcome the coupling effects by extracting correlated features from different estimation tasks. Numerical results show that the proposed HIENet can achieve comparable estimation accuracy to conventional one-shot and iterative methods, while at the same time having the smallest computation time.**

***Index Terms*—Carrier frequency offset, hardware impairment estimation, IQ imbalance, multi-task learning.**

I. Introduction

Orthogonal frequency division multiplexing (OFDM) is a fundamental physical layer technology of modern wireless networks. Although OFDM system achieves high spectral efficiency, its practical performance is limited by radio-frequency hardware impairments, including carrier frequency offset (CFO), in- and quadrature-phase (IQ) imbalance, phase noise, etc [1]. These impairments can jeopardize the orthogonality between subcarriers by introducing signal distortion and phase shift, and therefore cause severe inter-subcarrier interference. Thus, it is of great importance to estimate and compensate for these hardware impairments.

Due to the coupling effect between different hardware impairments, joint estimation of CFO and IQ imbalance is considered to be a challenging task [2]. The existing methods can be roughly classified into two categories: one-shot method and iterative method. The former, such as IQ-CFO-FD [3], is developed by using a sequential estimation framework, where different hardware impairments are estimated sequentially according to a predetermined order. The one-shot method achieves low computational complexity but suffers from high estimation bias as the estimation accuracy of one type of hardware impairment can be highly affected by the existence of other hardware impairments. To address the coupling effect, an iterative method using the channel residual energy (CRE) technique is proposed [4]. It performs iterative matrix inversion calculation to search for the optimal parameter combination for the considered hardware impairments, which results in high computational complexity. To balance the estimation accuracy and the computational complexity, the IsoTx method is proposed [5]. However, IsoTx requires a large number of pilot resources, which may decrease the overall spectral efficiency. Therefore, it is considered to be difficult for conventional methods to achieve high estimation accuracy while keeping low computational complexity [6].

Recently, deep learning (DL) methods have drawn much attention in signal processing as they can extract effective features from raw data without well-defined mathematical models [7]. This property makes them highly efficient to deal with problems without an analytical form [8]. In fact, the DL-based signal processing method has shown superior performance to the conventional model-based ones in the presence of severe hardware impairments [9]–[11]. In addition, DL methods can be utilized to accelerate the computation of conventional signal processing methods by exploiting the parallel structure of deep neural networks [12], [13]. As far as the authors have known, hardware impairment estimation has not been thoroughly studied in the context of DL.

In this paper, we develop a DL-based hardware impairment estimation network (HIENet) using a multi-task learning (MTL) paradigm. The network is comprised of a shared one-dimensional convolutional neural network and multiple task-specific dense subnetworks, where the shared network is dedicated to extracting commonly preferred representations, and each subnetwork corresponds to the estimation of a specific hardware impairment. We design a feature extraction and denoising (FED) block to construct the shared network, which is able to address the overfitting and gradient vanishing problems. Numerical results demonstrate that the proposed HIENet achieves comparable estimation accuracy to the conventional one-shot and iterative methods, while having the smallest computation time.

This work was partially supported by the National Natural Science Foundation of China under Grants 61931023 and U1936202.

Notations: Throughout this paper, we use boldface uppercase letters, boldface lowercase letters, and lowercase letters to denote matrices, column vectors, and scalars, respectively. $\boldsymbol{X}^*$ and $|\boldsymbol{X}|$ correspond to the complex conjugate and modulus of $\boldsymbol{X}$, respectively.

II. System Model

Consider an OFDM system with K subcarriers. We denote by ϵ_t and ϕ_t as the amplitude and phase mismatches between the I/Q mixers of the transmitter, ϵ_r and ϕ_r as the amplitude and phase mismatches of the receiver, and θ as the normalized phase shift due to the CFO.

Let $\mathbf{s} \in \mathbb{C}^K$ denote the pilot signal in the time domain. The transmit signal with transmitter IQ imbalance is then given by

$$\tilde{\mathbf{s}} = \mu_t \mathbf{s} + \nu_t \mathbf{s}^*, \tag{1}$$

where μ_t and ν_t are the complex weights of pilot signal $\mathbf{s}$ and image interference $\mathbf{s}^*$, respectively, written as [5]

$$\mu_t = \frac{1 + \epsilon_t e^{-j\phi_t}}{2}, \quad \nu_t = \frac{1 - \epsilon_t e^{-j\phi_t}}{2}. \tag{2}$$

The CFO introduces circular phase shifts for each subcarrier. For subcarrier k, the phase rotation is

$$e_k = e^{j2\pi\theta(k+G-1)/K}, \tag{3}$$

where G represents the cyclic prefix length [3]. Thus, the received signal in the time domain with CFO is given by

$$\tilde{\mathbf{r}} = \mathbf{E}_\theta \mathbf{H} \tilde{\mathbf{s}} + \mathbf{n}, \tag{4}$$

where $\mathbf{E}_\theta = \text{diag}(e_1, e_2, \cdots, e_K)$ is the phase shift matrix, $\mathbf{H}$ is the circulant channel matrix in the time domain, and $\mathbf{n} \in \mathbb{C}^K$ is the additive white Gaussian noise [4].

The received signal is then given by

$$\mathbf{r} = \mu_r \tilde{\mathbf{r}} + \nu_r \tilde{\mathbf{r}}^*, \tag{5}$$

where μ_r and ν_r are the complex weights of signal $\tilde{\mathbf{r}}$ and image interference $\tilde{\mathbf{r}}^*$, respectively, given by [5]

$$\mu_r = \frac{1 + \epsilon_r e^{-j\phi_r}}{2}, \quad \nu_r = \frac{1 - \epsilon_r e^{-j\phi_r}}{2}. \tag{6}$$

By substituting Eqs. (1) and (4) into (5), we have

$$\begin{aligned} \mathbf{r} =& \mu_r [\mathbf{E}_\theta \mathbf{H} (\mu_t \mathbf{s} + \nu_t \mathbf{s}^*) + \mathbf{n}] \\ &+ \nu_r [\mathbf{E}_\theta \mathbf{H} (\mu_t \mathbf{s} + \nu_t \mathbf{s}^*) + \mathbf{n}]^*. \end{aligned} \tag{7}$$

To simplify the symbols, we define

$$\alpha \triangleq \frac{\nu_t}{\mu_t}, \ \beta \triangleq \frac{\nu_r}{\mu_r^*}, \tag{8}$$

and an equivalent channel matrix

$$\widetilde{\mathbf{H}} \triangleq \mu_t \mu_r \mathbf{H}. \tag{9}$$

Thus, Eq. (7) can be rewritten as

$$\mathbf{r} = (\mathbf{E}_\theta \widetilde{\mathbf{H}} (\mathbf{s} + \alpha \mathbf{s}^*) + \mathbf{n}) + \beta (\mathbf{E}_\theta \widetilde{\mathbf{H}} (\mathbf{s} + \alpha \mathbf{s}^*) + \mathbf{n})^*. \tag{10}$$

The overall CFO and IQ imbalance estimation problem is

$$\max_{\theta, \alpha, \beta} \Pr(\mathbf{r}|\mathbf{s}). \tag{11}$$

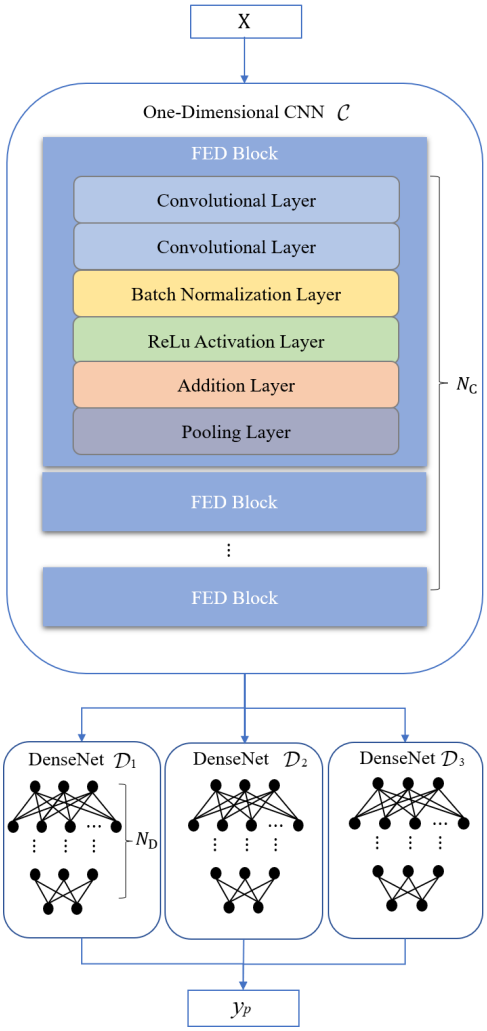

Fig. 1. Architecture of the proposed HIENet.

III. HIENet for Joint CFO and IQ Imbalance Estimation

The HIENet based on MTL adopts a hard parameter sharing paradigm to estimate CFO and IQ imbalance jointly. In this section, we present the architecture of HIENet and the corresponding training process. Besides, we analyze the online computational complexity of HIENet and explain its advantages in addressing the coupling effect and averaging the noise impact.

A. Architecture

The architecture of HIENet is shown in Fig.1. As we can see, it consists of a one-dimensional convolutional neural network $\mathcal{C}$ and three dense subnetworks $\mathcal{D}_1$, $\mathcal{D}_2$, and $\mathcal{D}_3$. The input $\mathbf{X}$ is a $2K \times 2$ matrix representing the real and imaginary parts of both the original pilot $\mathbf{s}$ and the received

signal $\mathbf{r}$, given by

$$\mathbf{X} = \begin{bmatrix} Re(s_1) & Im(s_1) \\ Re(r_1) & Im(r_1) \\ \vdots & \vdots \\ Re(s_K) & Im(s_K) \\ Re(r_K) & Im(r_K) \end{bmatrix}. \tag{12}$$

Input $\mathbf{X}$ goes through the network $\mathcal{C}$, which consists of N_{C} FED blocks. These blocks have the same architecture, including two convolutional layers, a batch normalization layer, a ReLu activation layer, an addition layer, and a pooling layer. The kernel size of the first convolutional layer is equally given by 1 in all blocks. While the kernel size of the second convolutional layer is given by M_n for any FED block $1 \leq n \leq N_C$. The output of $\mathcal{C}$ is then forwarded to all three subnetworks.

The subnetworks $\{\mathcal{D}_i\}_{i=1,2,3}$ share the same architecture consisting of N_{D} fully-connected layers, while their outputs constitute the final output $\mathbf{y}_p$, where $y_{p,1}$ is the output of $\mathcal{D}_1$ representing the estimated θ, $y_{p,2}$ and $y_{p,3}$ are the output of $\mathcal{D}_2$ representing the estimated real and imaginary parts of α, and $y_{p,4}$ and $y_{p,5}$ are the output of $\mathcal{D}_3$ representing the estimated real and imaginary parts of β.

B. Loss Function

We consider a mini-batch with N_b samples and the n-th sample is represented by $(\cdot)^{<n>}$. For any input $\mathbf{X}$, the corresponding label $\mathbf{y}_l$ is comprised of the related hardware impairment parameters, given by

$$\mathbf{y}_l = (\theta,\ Re(\alpha),\ Im(\alpha),\ Re(\beta),\ Im(\beta)). \tag{13}$$

We adopt the MSE criterion as below

$$\mathrm{MSE} = \frac{1}{N_b}\sum_{n=1}^{N_b}(y_p^{<n>} - y_l^{<n>})^2. \tag{14}$$

The losses of different hardware impairments are then given by

$$\mathcal{L}_\theta = \frac{1}{N_b}\sum_{n=1}^{N_b}(y_{p,1}^{<n>} - y_{l,1}^{<n>})^2, \tag{15}$$

$$\mathcal{L}_\alpha = \frac{1}{N_b}\sum_{n=1}^{N_b}\sum_{i=2}^{3}(y_{p,i}^{<n>} - y_{l,i}^{<n>})^2, \tag{16}$$

and

$$\mathcal{L}_\beta = \frac{1}{N_b}\sum_{n=1}^{N_b}\sum_{i=4}^{5}(y_{p,i}^{<n>} - y_{l,i}^{<n>})^2. \tag{17}$$

Note that θ, α and β are on different orders of magnitude, which may lead to the overfitting problem. Thus, we introduce two weighting parameters λ and γ to weight the general loss function $\mathcal{L}$, given by

$$\mathcal{L} = \mathcal{L}_\theta + \lambda\mathcal{L}_\alpha + \gamma\mathcal{L}_\beta. \tag{18}$$

C. Complexity

The computational complexity in DL is usually measured by floating operations (FLOPs). For the l-th convolutional layer of any FED block, the FLOPs are given by $(2C_{l-1}M_l - 1)W_lC_l$, where W_l is the width of the input feature map, M_l is the kernel size, and C_{l-1} and C_l are the channel numbers of the $(l-1)$-th and the l-th layers, respectively. For the l-th fully connected layer in any subnetwork $\{\mathcal{D}_i\}_{i=1,2,3}$, the number of FLOPs are given by $(2N_{l-1} - 1)N_l$, where N_{l-1} and N_l present the input and the output dimensions, respectively. The overall FLOPs are given by

$$\mathrm{F} = \sum_{1}^{N_C}\sum_{l=1}^{2}(2C_{l-1}M_l - 1)W_lC_l + 3\sum_{l=1}^{N_D}(2N_{l-1} - 1)N_l. \tag{19}$$

D. Advantages

The estimation of θ, α and β can be treated as three highly correlated tasks, for which the proposed HIENet shows advantages in the following aspects.

- **Bias Reduction:** For joint estimation problems, the coupling effect may highly decrease the individual estimation accuracy. In the proposed HIENet, independent subnetworks $\{\mathcal{D}_i\}_{i=1,2,3}$ force the shared network $\mathcal{C}$ to give representations that are preferred by all estimation tasks. Thus, the inherent correlation between CFO and IQ imbalance can be utilized to extract common features, which in return decreases the estimation bias and alleviates the impact of the coupling effect [14].
- **Noise Averaging:** For classic DL methods, all estimation tasks suffer from sample noise with different patterns, which deviates the learned model from the ground truth. In the proposed HIENet, the weighted loss function is designed to learn multiple tasks simultaneously, which averages the task-dependent noise patterns and enables the model to learn more general representations [15]. Thus, it improves the estimation accuracy with noisy samples.

IV. Numerical Results

We compare the performance of the proposed HIENet with three classical hardware impairment estimation methods, i.e., a one-shot estimation method, IQ-CFO-FD [3], and two iterative methods, CRE [4] and IsoTx [5]. First, we compare the estimation accuracy of θ, α and β with different SNRs using normalized MSE (NMSE), given by

$$\mathrm{NMSE} = \frac{1}{N}\sum_{n=1}^{N}\frac{|y_p^{<n>} - y_l^{<n>}|^2}{|y_l^{<n>}|^2}, \tag{20}$$

where N is the number of samples for online estimation. Then, we show the coupling effect by presenting the CFO (IQ imbalance) estimation performance as a function of the level of IQ imbalance (CFO). At last, we measure the average computation time on Intel i7 central processing unit and RTX 2060 graphic processing unit (GPU).

TABLE I
LAYER SETTING FOR THE ONE DIMENSIONAL CNN

Type of layer	Annotation	Output size
FED Block 1		
Convolutional	Kernel = 1	128 × 128
Convolutional	Kernel = 16	128 × 128
Normalization	Power Normalization	128 × 128
ReLU	Activation	128 × 128
Addition	Add Convolutional layers	128 × 128
Pooling	Pool Size = 4, Step = 2	64 × 128
FED Block 2		
Convolutional	Kernel = 1	64 × 64
Convolutional	Kernel = 8	64 × 64
Normalization	Power Normalization	64 × 64
ReLU	Activation	64 × 64
Addition	Add Convolutional layers	64 × 64
Pooling	Pool Size = 4, Step = 2	32 × 64
FED Block 3		
Convolutional	Kernel = 1	32 × 64
Convolutional	Kernel = 8	32 × 64
Normalization	Power Normalization	32 × 64
ReLU	Activation	32 × 64
Addition	Add Convolutional layers	32 × 64
Pooling	Pool Size = 4, Step = 2	16 × 64
FED Block 4		
Convolutional	Kernel = 1	16 × 64
Convolutional	Kernel = 4	16 × 64
Normalization	Power Normalization	16 × 64
ReLU	Activation	16 × 64
Addition	Add Convolutional layers	16 × 64
Flatten	Tensor Conversion	256
Dense	Activation = Linear	400

TABLE II
LAYER SETTING FOR SUBNETWORKS

Type of layer	Annotation	Output size
Subnetwork $\mathcal{D}_1$		
Dense	Activation = ReLU	100
Dense	Activation = ReLU	100
Dense	Activation = Linear	25
Dense	Activation = Linear	2
Dense	Activation = Linear	1
Subnetworks $\mathcal{D}_2$ and $\mathcal{D}_3$		
Dense	Activation = ReLU	100
Dense	Activation = ReLU	100
Dense	Activation = Linear	25
Dense	Activation = Linear	2
Dense	Activation = Linear	2

A. Parameter Setting

1) OFDM System: In the considered OFDM system, the number of subcarriers is $K = 64$, the subcarrier frequency interval is $\Delta f = 15$ kHz, the cyclic prefix length is $G = 16$, and the modulation scheme is quadrature phase-shift keying. We adopt a 6-path channel model according to COST 207, where channel propagation delays are $[0, 0.1, 0.2, 0.3, 0.4, 0.5]$ μs and power attenuations are $[0, -4, -8, -12, -16, -20]$ dB.

2) Network Parameter: In HIENet, the number of FED blocks is $N_{\mathrm{C}} = 4$, and the layer numbers of subnetworks $\{\mathcal{D}_i\}_{i=1,2,3}$ is $N_{\mathrm{D}} = 5$. The network parameters of FED blocks and subnetworks are detailed in Table I and Table II, respectively. The overall FLOPs are F = 75.7 million, and the memory usage is 29.6 MB. The numbers of training, validation and test samples are given by $N_s = 2 \times 10^5$, $N_v = 2 \times 10^3$, and $N_t = 1 \times 10^3$, respectively. The training samples are generated by varying the CFO parameter $\theta \in [-0.8, 0.8]$ and the IQ imbalance parameters $\epsilon_t/\epsilon_r \in [0.8, 1.2]$, $\phi_t/\phi_r \in [-0.35, 0.35]$, as well as the SNR in $[-15, 20]$ dB.

3) Network Traning: We adopt the mini-batch gradient descent algorithm and the Adam optimizer for network training, where the batch size is $N_b = 100$, and the learning rate is 1×10^{-4} for initial training and 0.8×10^{-5} for finetuning. The exponential decay rates for moment estimates are given by 0.9 and 0.999. The weight parameters in (18) are equally set as 1 at the beginning and then adjusted in the finetuning phase. The final values are $\lambda = 0.2$ and $\gamma = 0.1$.

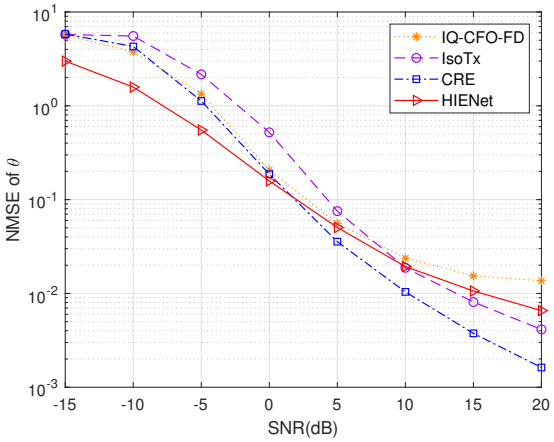

Fig. 2. NMSE of the CFO parameter θ against the SNR.

B. Estimation Accuracy

We compare the estimation accuracy of all methods by presenting the NMSEs of θ, α and β as a function of SNR. Specifically, we fix $\theta = 0.25$, $\epsilon_t = \epsilon_r = \epsilon = 1.1$, $\phi_t = \phi_r = \phi = 0.25$, and SNR $\in [-15, 20]$ dB.

Fig. 2 shows the NMSE of the CFO parameter θ against the SNR. In the low SNR region $[-15, -5]$ dB, conventional methods show low estimation accuracy as they cannot effectively address the noise impact. HIENet can average the noise patterns of different tasks by using the weighted loss function, which implies it can fit the random noise and alleviate the noise impact by using the training samples. Thus, HIENet provides higher estimation accuracy. In the high SNR region $[5, 20]$ dB, the estimation performance of IQ-CFO-FD is limited by the coupling effect, and the NMSE flattens with the SNR. However, HIENet utilizes the shared network to extract the correlated feature to address the coupling effect, and iterative methods can minimize the effect by using alternative optimization. Thus, their NMSEs continue to decrease.

Fig. 3 illustrates the NMSE of the transmitter IQ imbalance parameter α against the SNR. Since IQ-CFO-FD only considers the receiver IQ imbalance, it is not shown in this figure.

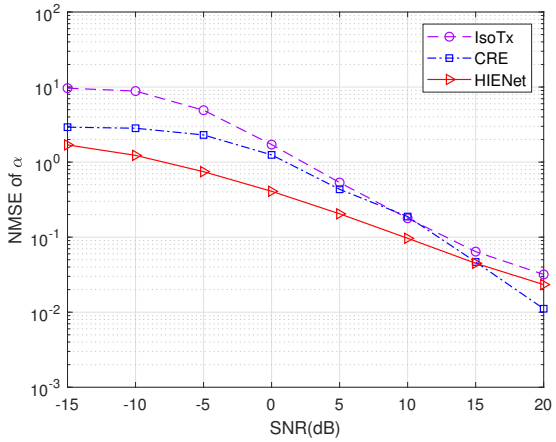

Fig. 3. NMSE of the transmitter IQ imbalance parameter α against the SNR.

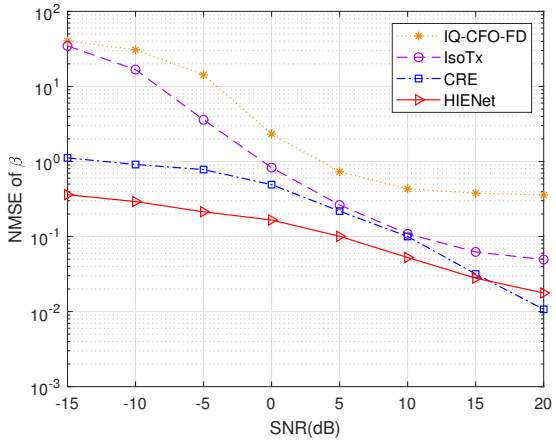

Fig. 4. NMSE of the receiver IQ imbalance parameter β against the SNR.

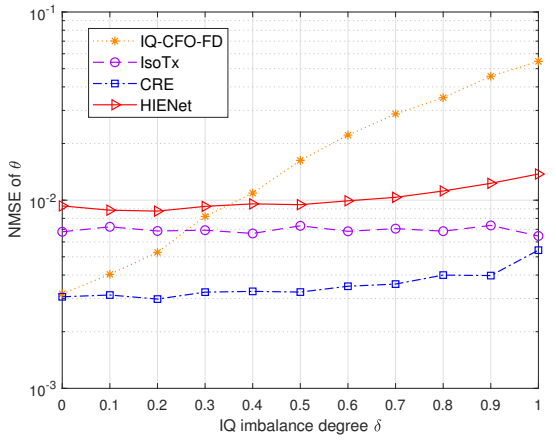

Fig. 5. NMSE of the CFO parameter θ against the IQ imbalance degree δ.

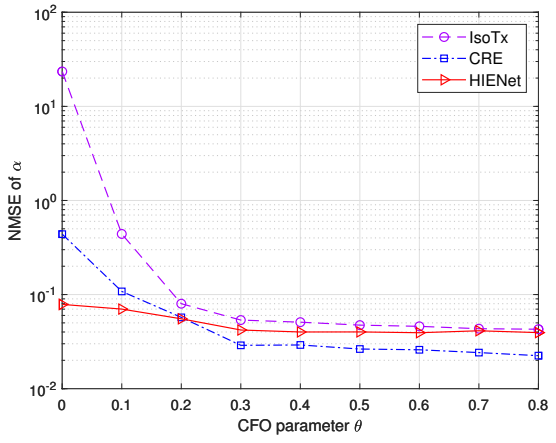

Fig. 6. NMSE of the transmitter IQ imbalance parameter α against θ.

In the low SNR region $[-15, -5]$ dB, HIENet outperforms the conventional methods because it can alleviate the noise impact by averaging the noise pattern of different tasks. In the high SNR region $[0,\ 20]$ dB, all methods can address the coupling effect due to the same reason in Fig. 2, and their NMSEs continue to decrease.

In Fig. 4, we depict the NMSE of the receiver IQ imbalance parameter β against the SNR. In the low SNR region $[-15, -5]$ dB, HIENet outperforms the conventional methods due to the same reason in Fig. 3. In the high SNR region $[10, 20]$ dB, the proposed and iterative methods can address the coupling effect, and their NMSEs keep decreasing. The NMSE of IQ-CFO-FD flattens since the estimation performance is limited by the residual CFO [3].

C. Coupling Effects

In this subsection, we show the impact of the coupling effect by presenting the NMSE of CFO (IQ imbalance) as a function of IQ imbalance (CFO). We first fix SNR = 15 dB, $\epsilon = 1.1$ and $\phi = 0.25$, and vary $\theta \in [0, 0.8]$. Then, we set $\theta = 0.25$, and vary the IQ imbalance parameters as $\epsilon = 1 + 0.2\delta$ and $\phi = 0.35\delta$, where $\delta \in [0, 1]$ is the IQ imbalance degree.

In Fig. 5, we compare the NMSE of the CFO parameter θ against the IQ imbalance parameter δ. The NMSE of IQ-CFO-FD increases with δ as it cannot address the IQ imbalance impact when estimating the CFO. The NMSE of CRE increases slightly with δ as it can reduce the IQ imbalance impact using alternative optimization. IsoTx presents even better performance and its NMSE keeps unchanged as the number of iterations increases. HIENet can alleviate the impact of IQ imbalance by using the MTL technique, which utilizes the shared network to fit the coupling effect by extracting the correlated features. Hence, it provides similar performance as iterative methods.

In Fig. 6, we present the NMSE of the transmitter IQ imbalance parameter α against the CFO parameter θ. Since IQ-CFO-FD only considers the receiver IQ imbalance, it is not shown in this figure. For conventional methods, it is difficult to estimate θ accurately or eliminate the residual CFO when the impact of the CFO is relatively small as compared to the IQ imbalance, which deteriorates the estimation accuracy of α. As θ increases, it can be accurately estimated, and the residual

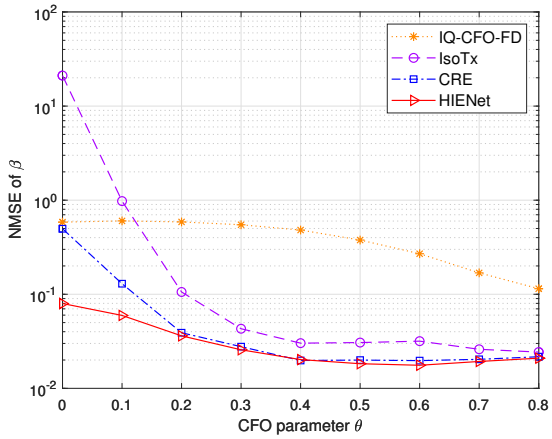

Fig. 7. NMSE of the receiver IQ imbalance parameter β against θ.

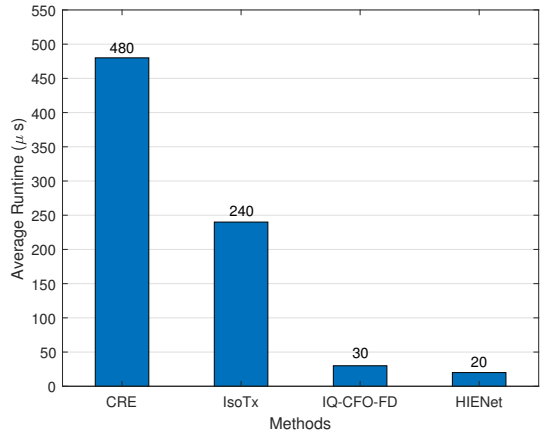

Fig. 8. Average runtime of the considered methods.

CFO can be eliminated. Thus, the NMSE of α decreases with θ. HIENet achieves a low NMSE of α for any level of θ due to the same reason in Fig. 5.

Fig. 7 shows the NMSE of the receiver IQ imbalance parameter β against the CFO parameter θ. For iterative methods, the NMSE of β decreases dramatically when θ varies from 0 to 0.2 and flattens when $\theta > 0.2$ due to the same reason in Fig. 6. HIENet achieves a low NMSE of β for any level of θ due to the same reason in Fig. 5. While IQ-CFO-FD cannot address the residual CFO, of which the NMSE remains high.

D. Runtime

In Fig. 8, we provide the average runtime of considered methods with 1000 simulations. The average runtime of CRE, IsoTx, IQ-CFO-FD, and HIENet are 480 μs, 240 μs, 30 μs, and 20 μs, respectively. CRE and IsoTx require several iterations to achieve convergency, which leads to large processing delays. IQ-CFO-FD performs simple algebraic operations with low computational complexity, which leads to small computation time. HIENet can be directly deployed on parallel processors due to its parallel architecture and achieves the lowest computation time of 20 μs.

V. Conclusion

In this paper, we have proposed an MTL-based estimator, HIENet, to jointly estimate the CFO and IQ imbalance in OFDM systems. The proposed HIENet can alleviate the noise impact by averaging the task-dependent noise patterns and address the coupling effect between the two hardware impairments by extracting correlated features from different tasks. As compared to conventional data-aided methods, the proposed HIENet can obtain comparable estimation accuracy, while having the smallest computation time.

References

[1] T. C. W. Schenk and E. R. Fledderus, "RF Impairments in High-Rate Wireless Systems - Understanding the Impact of TX/RX-Asymmetry," in *Proc. ISCCSP'08*, St Julians, Malta, Mar. 2008.

[2] Y. Meng *et al.*, "Joint CFO and I/Q Imbalance Estimation for OFDM Systems Exploiting Constant Modulus Subcarriers," *IEEE Trans. Veh. Technol.*, vol. 67, no. 10, pp. 10 076–10 080, Jul. 2018.

[3] J. Tubbax *et al.*, "Joint Compensation of IQ Imbalance and Frequency Offset in OFDM Systems," in *Proc. IEEE GLOBECOM'03*, San Francisco, CA, USA, Dec. 2003.

[4] Y.-H. Chung and S.-M. Phoong, "Joint Estimation of I/Q Imbalance, CFO and Channel Response for MIMO OFDM Systems," *IEEE Trans. Commun.*, vol. 58, no. 5, pp. 1485–1492, May 2010.

[5] Y. Zhuang and Y. Wan, "Joint Estimation of Carrier Frequency Offset and In-Phase/Quadrature-Phase Imbalances for Orthogonal Frequency Division Multiplexing Systems," *Comput. Electr. Eng.*, vol. 48, no. 5, pp. 1–11, Nov. 2015.

[6] M. Sandell *et al.*, "Estimation of Wideband IQ Imbalance in MIMO OFDM Systems With CFO," *IEEE Trans. Wireless Commun.*, vol. 20, no. 9, pp. 5821–5830, Sept. 2021.

[7] T. Wang, S. Wang, and Z.-H. Zhou, "Machine Learning for 5G and Beyond: From Model-Based to Data-Driven Mobile Wireless Networks," *China Commun.*, vol. 16, no. 1, pp. 165–175, Jan. 2019.

[8] S. Liu, T. Wang, and S. Wang, "Toward Intelligent Wireless Communications: Deep Learning-Based Physical Layer Technologies," *Digit. Commun. Netw.*, vol. 7, no. 4, pp. 589–597, Dec. 2021.

[9] A. Mohammadian, C. Tellambura, and G. Y. Li, "Deep Learning LMMSE Joint Channel, PN, and IQ Imbalance Estimator for Multicarrier MIMO Full-Duplex Systems," *IEEE Wireless Commun. Lett.*, vol. 11, no. 1, pp. 111–115, Jan. 2022.

[10] S. Liu, T. Wang, and S. Wang, "Joint Compensation of CFO and IQ Imbalance in OFDM Receiver: A Deep Learning Based Approach," in *Proc. IEEE ICCC'2021*, Xiamen, China, Nov. 2021.

[11] H. Ye, G. Y. Li, and B. Juang, "Power of Deep Learning for Channel Estimation and Signal Detection in OFDM Systems," *IEEE Commun. Lett.*, vol. 7, no. 1, pp. 114–117, Feb. 2018.

[12] W. Yu, T. Wang, and S. Wang, "Multi-Label Learning Based Antenna Selection in Massive MIMO Systems," *IEEE Trans. Veh. Technol.*, vol. 70, no. 7, pp. 7255–7260, Jun. 2021.

[13] X. Sheng and S. Wang, "Online Primary User Emulation Attacks in Cognitive Radio Networks Using Thompson Sampling," *IEEE Trans. Wireless Commun.*, vol. 20, no. 12, pp. 8264–8273, Jun. 2021.

[14] Y. Zhang and Q. Yang, "A Survey on Multi-Task Learning," *IEEE Trans. Knowl. Data Eng.*, 2021 (Early Access).

[15] K. Thung and C. Wee, "A Brief Review on Multi-Task Learning," *Multimed Tools Appl.*, vol. 77, no. 22, pp. 29 705–29 725, Aug. 2018.