Hardware Impairment Estimation in NB-IoT: A Parallel Multitask Learning Method

Siqi Liu, *Student Member, IEEE*, Tianyu Wang, *Member, IEEE*, and Shaowei Wang, *Senior Member, IEEE*

***Abstract*—Orthogonal frequency-division multiplexing (OFDM) is widely adopted in narrowband Internet of Things (NB-IoT). Nevertheless, the OFDM system is highly sensitive to the impairments caused by imperfect radio-frequency hardwares, which may greatly jeopardize the orthogonality between different subcarriers and degrade the demodulation performance. Though large efforts have been devoted to the hardware impairment estimation, however, it is highly challenging to jointly estimate multiple hardware impairments due to their coupling effects, especially for the NB-IoT systems which usually work in low signal-to-noise ratio (SNR) regions with limited computing and radio resources. In this article, we propose a parallel multitask learning (MTL) estimator to jointly estimate carrier frequency offset and in- and quadrature-phase imbalance in NB-IoT systems. Specifically, MTL is introduced to extract the inherent correlations between different hardware impairments so as to address the coupling effect and average the noise impact. In addition, we propose a parallel structure and a sliding window scheme to reduce the network complexity and decrease the estimation bias. Numerical results show that our proposed parallel MTL estimator can jointly estimate multiple hardware impairments with short pilot sequences, and outperform the conventional methods in terms of estimation accuracy and computation time in the typical SNR regions of the IoT devices.**

***Index Terms*—Carrier frequency offset (CFO), in- and quadrature-phase (IQ) imbalance, multitask learning (MTL), narrowband Internet of Things (NB-IoT), orthogonal frequency-division multiplexing (OFDM).**

I. Introduction

NARROWBAND-INTERNET of Things (NB-IoT) is a low-power wide-area solution, which reuses most existing design principles of 4G/5G systems, like orthogonal frequency-division multiplexing (OFDM), channel coding, and rate matching, to achieve the stringent constraints of super coverage, massive connection, and low cost [1]. In contrast, plenty of features, such as synchronization, random access, broadcast and control channels, etc., are modified to better support the specific needs of IoT communication [2]. Such modifications also give birth to a host of new technical challenges, one of which is the estimation of radio-frequency hardware impairments [3].

NB-IoT adopts OFDM for its low complexity and high-spectral efficiency, which however, is extremely sensitive to the impairments caused by the imperfection of radio-frequency hardwares, such as carrier frequency offset (CFO), phase noise, in- and quadrature-phase (IQ) imbalance, and power amplifier nonlinearity [4]. On the one hand, these hardware impairments can greatly jeopardize the orthogonality between different subcarriers and result in severe magnitude distortion and phase mismatch, which greatly degrades the demodulation performance [5]. On the other hand, due to the low cost and low-power constraints, IoT devices may suffer from severe hardware impairments [6].

In addition, low-cost chips with low-capacity flash are equipped in IoT devices, of which the memory and computing resources are highly limited, and thus most existing estimation schemes can be intractable due to their heavy computation loads [7]. Besides, the low-power constraint of IoT devices implies that the receiver generally works in the low and medium signal-to-noise ratio (SNR) regions, which leads to severe noise impact and large estimation bias [8]. Therefore, these limitations make the corresponding hardware impairment estimation highly challenging.

Besides the individual effect of each hardware impairment, the coupling effects between them make the joint estimation task even more challenging [9]. We take CFO and IQ imbalance as an example. CFO is caused by the frequency deviation of oscillators, which introduces intercarrier interference and leads to estimation bias of signal detection and demodulation [10], [11]. IQ imbalance is caused by the imperfect mixers of low-cost down-convertor receivers, which introduces mirror interference and incomplete image rejection [12]. When CFO and IQ imbalance coexist, the intercarrier interference caused by CFO introduces extra phase and amplitude mismatches in I- and Q-branches, which causes a large estimation bias in IQ imbalance estimation [13]. Also, the mirror interference caused by IQ imbalance introduces extra carrier frequency deviation, which breaks the phase inconformity in CFO estimation [14], [15]. Hence, the conventional estimation methods focusing on single hardware impairment may not be suitable when the coupling effects are severely amplified in the low-performance radio-frequency components in NB-IoT.

The existing methods for joint estimation of the multiple hardware impairments can be roughly classified into blind methods and data-aided ones. The blind methods utilize null subcarriers [16], constant modulus subcarriers [17], cyclic

Manuscript received 1 October 2022; revised 16 November 2022; accepted 2 December 2022. Date of publication 9 December 2022; date of current version 7 April 2023. This work was supported in part by the National Natural Science Foundation of China under Grant 61931023 and Grant U1936202. This article was presented in part at IEEE Global Communications Conference, Rio de Janeiro, Brazil, December 4–8, 2022. *(Corresponding author: Shaowei Wang.)*

The authors are with the School of Electronic Science and Engineering, Nanjing University, Nanjing 210023, China (e-mail: dg1823047@smail.nju.edu.cn; tianyu.alex.wang@nju.edu.cn; wangsw@nju.edu.cn).

Digital Object Identifier 10.1109/JIOT.2022.3228292

prefix (CP) [18], and redundant information of multiple antennas [5] to exploit the high-order statistical features in both the time and frequency domains. They usually apply grid search procedures or high-order polynomial rooting algorithms, which jointly process a large number of OFDM symbols for feature extraction and cause high-computational complexity [19].

The data-aided methods have lower computational complexity and higher estimation accuracy as compared to blind ones as they can exploit the inconformity between the transmitted and the received pilot sequences. The data-aided methods can be further classified into the one-shot methods [20], [21], [22], [23], [24], [25], [26] and the iterative methods [27], [28], [29], [30]. The one-shot methods adopt a serial estimation framework, where the impairment parameters are estimated sequentially in a predefined order. They usually achieve extremely low-computational complexity but suffer from a high-estimation bias as they usually neglect the coupling effects between different hardware impairments. While the iterative methods can achieve high-estimation accuracy by applying the alternative optimization algorithm, which iteratively updates a single hardware impairment parameter according to the latest estimation of other parameters. However, the iterative methods are of high-computational complexity as they usually perform iterative matrix inversion calculations. Therefore, it has become a challenging task to jointly estimate multiple hardware impairments with low-computational complexity.

Recently, deep learning (DL) has drawn much attention in the study of wireless communications. The associated neural network can directly extract features from raw data without giving rigid mathematical models [31]. In the physical layer, the DL methods have exhibited impressive performance improvement as compared to the conventional methods based on the mathematical models and the classical optimization algorithms, especially for scenarios where there are a variety of influencing factors that cannot be described in analytical forms [32]. Specifically, deep neural network (DNN)-based signal detectors have been proposed to compensate for hardware impairments, including the CFO, the IQ imbalance, and the nonlinear distortion caused by power amplifiers and clipping [33], [34], [35], [36], and also for multiple hardware impairments [37], [38]. These methods are purely data-driven methods without using any prior knowledge of the considered hardware impairments [39] and they need a neural network with a huge number of parameters, which suffers from the heavy computational burden.

In this article, we design a parallel multitask learning (MTL) estimator to jointly estimate CFO and IQ imbalance in NB-IoT systems. Specifically, we introduce MTL to address the coupling effects and the noise impact and utilize a parallel structure and a sliding window scheme to reduce the network complexity. Our main contributions are fourfold as follows.

1) We introduce MTL to extract the correlated features of both CFO and IQ imbalance, which can efficiently address the coupling effect and average the task-specific noise patterns.
2) We develop a parallel structure in the design of the proposed estimator, which greatly reduces the number of parameters of the neural network and thus decreases the computation complexity in both the training and inference phases.
3) We design a sliding window technique to reduce the input dimension of each MTL subnetwork by using the prior knowledge of CFO and IQ imbalance, which can highly accelerate the convergence speed of the training phase.
4) Numerical results show that the proposed parallel MTL estimator can effectively address the coupling effect with short pilot sequences, and achieves higher estimation accuracy than the conventional methods in typical SNR regions of IoT devices. Also, the proposed parallel MTL estimator can reduce the computation time as compared to the iterative methods and the single MTL method.

The remainder of this article is organized as follows. In Section II, we review the related works. Section III illustrates the system model. In Section IV, we give our proposed MTL estimator in detail. Section V includes numerical results and discussions. Section VI is the conclusion of this work. *Notations*: Throughout this article, we use boldface uppercase letters, boldface lowercase letters, and lowercase letters to denote matrices, column vectors, and scalars, respectively. $\mathbf{X}^*$, $\mathbf{X}^T$, $\mathbf{X}^\dagger$, and $|\mathbf{X}|$ correspond to the complex conjugate, the transpose, the transpose-conjugate, and the modulus of $\mathbf{X}$, respectively. $\mathrm{Re}(\mathbf{X})$ and $\mathrm{Im}(\mathbf{X})$ represent the real and the imaginary parts of $\mathbf{X}$, respectively. $\mathbf{Pr}(\cdot)$ and $\max[\cdot]$ denote the probability and the maximization operation, respectively.

II. Related Work

A. Data-Aided Methods

The data-aided methods can be roughly classified into the one-shot methods and the iterative methods. The one-shot methods infer all hardware impairments by a predefined order, and most of them only consider CFO and the receiver IQ imbalance, where the transmitter IQ imbalance is assumed to be perfectly calibrated before signal transmission [13]. The time-domain estimator proposed in [20] adopts three consecutive pilot sequences to estimate CFO and the receiver IQ imbalance, which aims to minimize the estimation bias. The frequency-domain estimator proposed in [21] aims to deal with tiny CFO, which achieves extremely low-computational complexity.

In [22], a novel generalized periodic pilot technique is proposed, which utilizes multiple periodic OFDM symbols to improve the estimation accuracy. However, the spectral efficiency is reduced. The least square (LS) estimation method proposed in [23] improves the spectral efficiency, which needs only two repetitive short pilot sequences and achieves extremely low-computational complexity.

The transmitter IQ imbalance can be estimated at the receiver as well, which however, requires extra pilot sequences and specific pilot architectures to avoid the impact of channel fading [24]. Hence, it may result in low-spectral efficiency. To increase the spectral efficiency, the generalized periodic pilot schemes are proposed in [25] and [26]. The former extends the effective estimation range of CFO by using three pilot sequences. The latter further improves the IQ imbalance estimation accuracy by using five long pilot sequences, each of which consists of two OFDM symbols with 256 subcarriers in the frequency domain.

The iterative methods iteratively estimate the parameters of one hardware impairment by fixing the latest estimation of the other hardware impairments [27]. An expectation maximization-based iterative estimator is developed in [28], which obtains high-estimation accuracy. However, this method has high-computational complexity and low-spectral efficiency since it adopts a large number of pilot sequences and performs iterative matrix inversion operations. The channel residual energy (CRE) method is then proposed to improve the spectral efficiency, which only requires one long pilot sequence [29]. The transmitter IQ imbalance isolated estimator proposed in [30] achieves an efficient tradeoff between the estimation accuracy and the computational complexity. In general, the iterative methods can effectively address the coupling effect but suffer from heavy computation loads.

B. DL-Based Methods

The estimation performance of the conventional data-aided methods is highly limited by the computing and radio resources, which motivates the DL-based methods. The DL methods utilize a specific neural architecture to implicitly compensate for hardware impairments [31]. In [33], a DL-based receiver is developed based on a fully connected (FC) DNN, which can implicitly compensate for the nonlinear distortions caused by the clipping operation. However, the FC structure leads to high-computational complexity. A low-complexity serial architecture that combines the conventional LS algorithm and the zero-forcing algorithm is proposed in [34], which can achieve comparable bit error rate performance with fewer network parameters. In [35], an end-to-end transceiver is developed, which implicitly compensates for single hardware impairment, including the CFO and IQ imbalance. A DenseNet-based receiver is proposed in [36] to achieve a lower end-to-end processing time delay and bit error rate.

In addition, the DL methods can also be utilized to compensate for multiple hardware impairments. In [37], a parallel CNN-based network for joint channel estimation and signal detection is proposed, which can directly recover the transmitted signals from the received signals when both CFO and IQ imbalance exist. In [38], a DNN is embedded into a conventional linear minimum mean square estimator, which can accelerate the joint estimation of phase noise and IQ imbalance.

C. MTL Technique

MTL is a supervised learning paradigm that aims to learn multiple tasks simultaneously. The MTL model usually consists of feature extraction networks and task-specific networks, which are utilized to learn correlated features and achieve task-specific objectives, respectively [40]. The design of the feature extraction networks is the most essential part of MTL, which can be roughly classified into hard and soft parameter sharing. For the hard parameter sharing paradigm, all tasks share the same feature extraction network, which significantly reduces the training costs and improves the training efficiency [41]. For the soft parameter sharing paradigm, the multiple feature extraction networks of different tasks are different, while at the same time, appropriate regularization techniques, such as LS and minimum mean square error, are introduced to encourage their network parameters to converge to similar levels [42].

The feature extraction network can learn to extract the inherent correlations between different tasks, which are forwarded into task-specific networks as shared input. In return, the task-specific networks force the feature extraction network to learn the representations that all tasks prefer, which significantly improves the generalization performance [43]. Besides, the training process for all networks shares the same loss function, which can average the task-specific noise patterns and reduce the estimation bias [44]. Owing to such advantages, MTL has shown superior performance compared to single neural networks in addressing multiple related tasks and thus has been widely utilized in the field of image recognition, text classification, and advertising recommendation [45].

III. System Model

Consider an OFDM system with K subcarriers and CP length G. Denote $\mathbf{s} \in \mathbb{C}^K$ and $\mathbf{H}$ as the original time-domain signal at the transmitter and the circulant channel matrix with the size of $K \times K$, respectively, where the first column of $\mathbf{H}$ is $[h(0) \ \cdots \ h(L) \ 0 \ \cdots \ 0]^T$ and $L < G$ [29]. If there is no hardware impairment, the received signal $\mathbf{r}$ is given by

$$\mathbf{r} = \mathbf{H}\mathbf{s} + \mathbf{n} \tag{1}$$

where $\mathbf{n}$ represents the additive white Gaussian noise.

Now suppose that there is transmitter IQ imbalance, and ϵ_t and ϕ_t indicate the amplitude and phase mismatches between the IQ mixers of the transmitter, respectively [29]. The corresponding transmitter IQ imbalance impairment parameters are defined as

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

and

$$\nu_t = \frac{1 - \epsilon_t e^{-j\phi_t}}{2} \tag{3}$$

representing the degrees of signal distortion and image interference, respectively. With the transmitter IQ imbalance, the transmitter signal is given by

$$\tilde{\mathbf{s}} = \mu_t \mathbf{s} + \nu_t \mathbf{s}^* \tag{4}$$

and the received signal is given by

$$\tilde{\mathbf{r}} = \mathbf{H}\tilde{\mathbf{s}} + \mathbf{n}. \tag{5}$$

We denote Δf as the frequency shift between the transmitter and the receiver, and f as the subcarrier bandwidth. Thus, the normalized CFO is given by [5]

$$\theta = \frac{\Delta f}{f}. \tag{6}$$

For the kth subcarrier, the phase rotation e_k is given by

$$e_k = e^{j2\pi\theta(k+G-1)/K}. \tag{7}$$

Hence, the received signal with CFO is given by

$$\begin{aligned} \mathbf{r}' &= \mathbf{E}_\theta \tilde{\mathbf{r}} \\ &= \mathbf{E}_\theta \mathbf{H}\tilde{\mathbf{s}} + \mathbf{n} \end{aligned} \tag{8}$$

where $\mathbf{E}_\theta = \text{diag}(e_1, e_2, \ldots, e_K)$ indicates the phase rotation matrix.

Now suppose that there is the receiver IQ imbalance, and ϵ_r and ϕ_r represent the amplitude and phase mismatches between IQ mixers of the receiver, respectively [29]. The corresponding receiver IQ imbalance impairment parameters are defined as

$$\mu_r = \frac{1+\epsilon_r e^{-j\phi_r}}{2} \tag{9}$$

and

$$\nu_r = \frac{1-\epsilon_r e^{-j\phi_r}}{2} \tag{10}$$

representing the degrees of signal distortion and image interference, respectively. With the receiver IQ imbalance, the received signal is given by

$$\mathbf{r}'' = \mu_r \mathbf{r}' + \nu_r \mathbf{r}'^*. \tag{11}$$

By substituting (4) and (8) into (11), the received pilot sequence $\tilde{\mathbf{r}}$ with all impairments is rewritten as

$$\begin{aligned}\mathbf{r}'' = {} & \mu_r\left[\mathbf{E}_\theta \mathbf{H}\left(\mu_t \mathbf{s} + \nu_t \mathbf{s}^*\right) + \mathbf{n}\right] \\ & + \nu_r\left[\mathbf{E}_\theta \mathbf{H}\left(\mu_t \mathbf{s} + \nu_t \mathbf{s}^*\right) + \mathbf{n}\right]^*.\end{aligned} \tag{12}$$

To simplify the notations, we define two complex parameters for the IQ imbalance [26]

$$\alpha \triangleq \frac{\nu_t}{\mu_t} \tag{13}$$

and

$$\beta \triangleq \frac{\nu_r}{\mu_r^*} \tag{14}$$

and one equivalent channel matrix

$$\widetilde{\mathbf{H}} \triangleq \mu_t \mu_r \mathbf{H}. \tag{15}$$

By substituting (13), (14), and (15) into (12), we have

$$\begin{aligned}\mathbf{r}'' = {} & \left(\mathbf{E}_\theta \widetilde{\mathbf{H}}\left(\mathbf{s} + \alpha \mathbf{s}^*\right) + \mathbf{n}\right) \\ & + \beta\left(\mathbf{E}_\theta \widetilde{\mathbf{H}}\left(\mathbf{s} + \alpha \mathbf{s}^*\right) + \mathbf{n}\right)^*\end{aligned} \tag{16}$$

and the joint estimation of CFO and IQ imbalance can be formulated as

$$\max_{\theta,\alpha,\beta} \Pr\left(\mathbf{r} = \mathbf{r}''|\mathbf{s}\right) \tag{17}$$

which maximizes the conditional probability of $\mathbf{r} = \mathbf{r}''$ when $\mathbf{s}$ is known by optimizing the values of θ, α, and β.

IV. Parallel MTL Estimator

The proposed parallel MTL estimator consists of a data dimension reduction module, multiple MTL subnetworks, and an addition layer, as illustrated in Fig. 1. The data dimension reduction module utilizes a sliding window to divide the original feature sample into T input samples, which are then forwarded into T parallel MTL subnetworks. These subnetworks share the same architecture, which consists of a shared CNN for feature extraction and three FC networks for parameter estimation. In addition, these subnetworks share the same network parameters. Through the subnetworks, we obtain T groups of estimated results, which are then averaged in the addition layer to give the final output. The remainder of this section details the design principle, the training process, and the corresponding advantages.

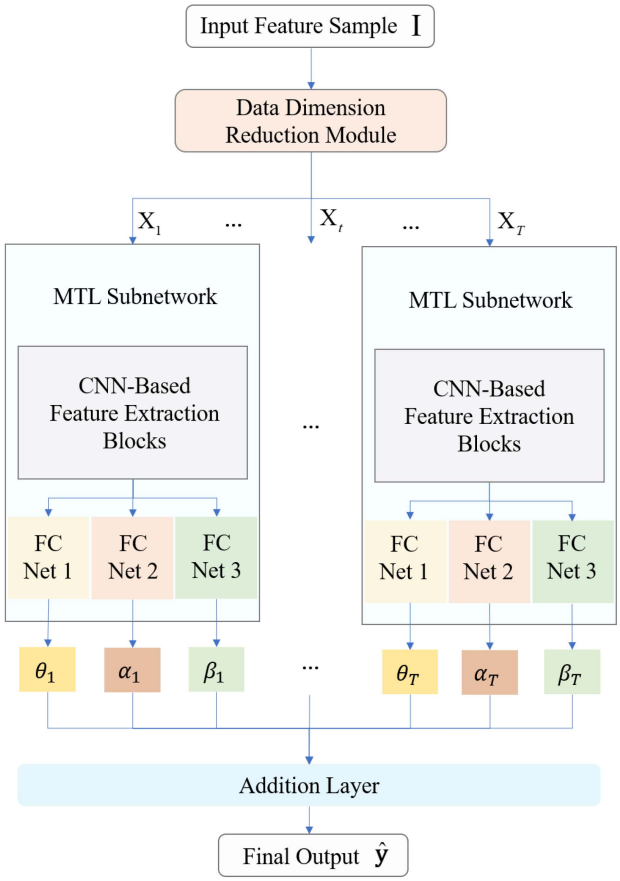


Fig. 1. Architecture of the proposed parallel MTL estimator.

A. Original Samples

First, we generate N_s, N_v, and N_t original samples for training, validation, and test data sets, respectively. Each original sample consists of a feature sample $\mathbf{I}$ and a corresponding label $\mathbf{y}_l$. Each feature sample $\mathbf{I}$ is comprised of p consecutive pilot sequences, including the transmitted and received signals, given by

$$\mathbf{I} = \begin{bmatrix} \text{Re}(S_{1,1}) & \text{Im}(S_{1,1}) \\ \text{Re}(R_{1,1}) & \text{Im}(R_{1,1}) \\ \vdots & \vdots \\ \text{Re}(S_{1,K}) & \text{Im}(S_{1,K}) \\ \text{Re}(R_{1,K}) & \text{Im}(R_{1,K}) \\ \vdots & \vdots \\ \text{Re}(S_{p,1}) & \text{Im}(S_{p,1}) \\ \text{Re}(R_{p,1}) & \text{Im}(R_{p,1}) \\ \vdots & \vdots \\ \text{Re}(S_{p,K}) & \text{Im}(S_{p,K}) \\ \text{Re}(R_{p,K}) & \text{Im}(R_{p,K}) \end{bmatrix} \tag{18}$$

where $\mathbf{S}$ and $\mathbf{R}$ correspond to the frequency-domain signals of $\mathbf{s}$ and $\mathbf{r}''$, respectively. The label $\mathbf{y}_l$ corresponds to the CFO parameter θ and IQ imbalance parameters α and β, defined by

$$\mathbf{y}_l = (\theta,\ \text{Re}(\alpha),\ \text{Im}(\alpha),\ \text{Re}(\beta),\ \text{Im}(\beta)). \tag{19}$$

Note that the dimension of the original feature sample is positively correlated with the pilot length, which is usually too

large for the input of neural networks. Hence, it is necessary to reduce the input dimension.

B. Dimension Reduction

we develop a sliding window to reduce the input dimension of the MTL subnetwork, as shown in Fig. 1, where the original feature sample $\mathbf{I}$ is divided into T smaller input blocks $\{\mathbf{X}_t\}_{t=1,2,\ldots,T}$, while the corresponding label $\mathbf{y}_l$ remains unchanged. The window size is given by $L_w \times 2$, where the window length and width are L_w and 2, respectively. We adopt a window width of 2 to guarantee that both the real and the imaginary parts of an original feature sample are scanned. The window length L_w is a hyperparameter, which is related to the training efficiency, computational complexity, and estimation accuracy of the MTL subnetwork. As a hyperparameter, the optimal value of window length L_w is determined by trial and error.

IQ imbalance leads to both signal distortion and image interference. To infer the IQ imbalance parameters, each input sample must include at least one pair of conjugated subcarriers. Note that the $([K/2]+1)$th subcarrier conjugates with itself. Thus, as long as the input sample includes the $([K/2]+1)$th subcarrier, the IQ imbalance parameters can be successfully estimated. Hence, considering that the input sample should include both transmitted and received signals, the low bound of the window size is given by

$$L_w \geq 2 \times \left(\frac{K}{2}+1\right) = K+2. \tag{20}$$

CFO causes consecutive phase shift between adjacent subcarriers, which implies that CFO is positively related to the subcarrier intervals. To infer the CFO parameter, each input sample is required to keep the same length to guarantee that the shifts of all input samples are the same, and the subcarriers of each input sample should be consecutive. These two requirements are naturally satisfied by using a sliding window. Hence, the upper bound of the window length L_w is given by

$$L_w \leq 2pK \tag{21}$$

where $2pK$ is the length of the original feature sample.

We denote L_s as the sliding step. Thus, the number of input blocks is given by

$$T = \left\lfloor \frac{2pK - L_w}{L_s} + 1 \right\rfloor \tag{22}$$

where $\lfloor \cdot \rfloor$ indicates rounding down operation.

C. MTL Subnetwork

We introduce the multiple MTL subnetworks in the proposed estimator, which share the same architecture and network parameters. Hence, only one set of network parameters is required to be trained during the offline training phase, which significantly reduces the training cost.

1) Architecture: We adopt the hardware parameter sharing paradigm for the MTL subnetwork design to extract the features of different estimation tasks. As compared to the soft parameter sharing paradigm, it has much less training parameters and highly reduces the training cost. The designed MTL subnetwork consists of a shared 1-D CNN $\mathcal{C}$ and three FC networks $\{\mathcal{D}_i\}_{i=1,2,3}$, as illustrated in Fig. 2. The shared CNN is comprised of N_C repeated feature extraction blocks, which learn to extract the inherent correlation between different tasks according to the input sample $\mathbf{X}_t$. All feature extraction blocks reuse the same layer structure, including two convolutional layers, one batch normalization layer, one activation layer, one addition layer, and one pooling layer. Each block is designed based on the basic architecture of ResNet to avoid the gradient vanishing problem. The kernel size is set to be different for different convolutional layers to improve the diversity of the receptive fields, which helps to extract features embedded in different parts of the input signal. These blocks output the correlated features, which are then forwarded into the following FC networks.

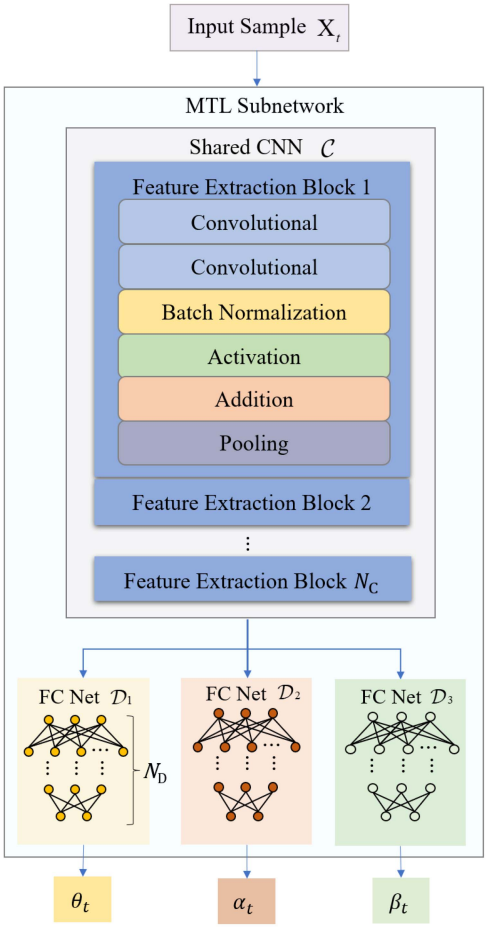


Fig. 2. Architecture of the subnetwork.

The FC networks $\mathcal{D}_1$, $\mathcal{D}_2$, and $\mathcal{D}_3$ are trained to infer the CFO parameter θ, the transmitter IQ imbalance parameter α, and the receiver IQ imbalance parameter β, respectively. For a given input sample $\mathbf{X}_t$, the corresponding outputs of $\mathcal{D}_1$, $\mathcal{D}_2$, and $\mathcal{D}_3$ are denoted by θ_t, α_t, and β_t, respectively. Note that α_t and β_t are complex numbers, thereby we assign two neurons for the output layers of $\mathcal{D}_2$ and $\mathcal{D}_3$ to match their real and imaginary parts separately. By concatenating the outputs of all FC networks, the output of the MTL subnetwork can be written as

$$\mathbf{y}_t = (\theta_t,\ \mathrm{Re}(\alpha_t),\ \mathrm{Im}(\alpha_t),\ \mathrm{Re}(\beta_t),\ \mathrm{Im}(\beta_t)). \tag{23}$$

2) Loss Function: The MTL subnetwork is trained based on a specifically designed loss function based on the minimum square error (MSE) function

$$\text{MSE} = \frac{1}{N_b}\sum_{n=1}^{N_b}|\mathbf{y}_t^n - \mathbf{y}_l^n|^2 \tag{24}$$

where N_b and $(\cdot)^n$ stand for the batch size and the nth sample of the batch. The proposed loss function is related to the estimation tasks of the CFO, the transmitter, and the receiver IQ imbalance, of which the MSEs are given by

$$\mathcal{L}_\theta = \frac{1}{N_b}\sum_{n=1}^{N_b}\left(y_{t,1}^n - y_{l,1}^n\right)^2 \tag{25}$$

$$\mathcal{L}_\alpha = \frac{1}{2N_b}\sum_{n=1}^{N_b}\sum_{i=2}^{3}\left(y_{t,i}^n - y_{l,i}^n\right)^2 \tag{26}$$

and

$$\mathcal{L}_\beta = \frac{1}{2N_b}\sum_{n=1}^{N_b}\sum_{i=4}^{5}\left(y_{t,i}^n - y_{l,i}^n\right)^2 \tag{27}$$

respectively.

According to (25), (26), and (27), the general loss function is given by

$$\mathcal{L} = \mathcal{L}_\theta + \lambda\left(\mathcal{L}_\alpha + \mathcal{L}_\beta\right). \tag{28}$$

Note that the hardware impairment parameters θ, α, and β are on different orders of magnitude, which may lead to the overfitting problem. Thus, we introduce a parameter λ to weight the general loss function, where λ is a hyperparameter and its optimal value is determined by trial and error.

3) Training: The MTL subnetwork training consists of a forward propagation phase and a successive backward propagation phase. During the forward propagation phase, the neural network calculates the average loss of each mini-batch according to (28). During the backward propagation phase, the subnetwork updates the weight parameters by using the mini-batch gradient descent algorithm. Denote $\mathbf{W}_C$ and $\mathbf{W}_i$ as the network parameters for the shared CNN and the ith FC network, respectively. The gradient direction of the FC network $\mathcal{D}_i$ is given by

$$\nabla\mathbf{W}_i = \partial\mathcal{L}/\partial\mathbf{W}_i \tag{29}$$

which can be calculated according to the chain rule. The gradient direction of the CNN $\mathcal{C}$ is given by

$$\nabla\mathbf{W}_C = \sum_{i=1}^{3}\frac{\partial\mathcal{L}}{\partial\mathbf{W}_i}\frac{1}{\partial\mathbf{W}_C}. \tag{30}$$

Based on (29) and (30), the subnetwork parameters can be updated according to

$$\begin{aligned}\mathbf{W}_C' &= \mathbf{W}_C - \eta\nabla\mathbf{W}_C\\ \mathbf{W}_i' &= \mathbf{W}_i - \eta\nabla\mathbf{W}_i\end{aligned} \tag{31}$$

where $\mathbf{W}_C'$ and $\mathbf{W}_i'$ are the updated network parameters for the shared CNN and the ith FC network, respectively, and η is the learning rate.

D. Addition Layer

The addition layer is utilized to average the outputs of all subnetworks. As shown in Fig. 1, each original sample $\mathbf{I}$ is divided into T input samples $\{\mathbf{X}_t\}_{t=1,2,\ldots,T}$ in the data dimension reduction module. The input samples are then forwarded into T parallel MTL subnetworks, where the tth MTL subnetwork executes the estimation procedure with the input sample $\mathbf{X}_t$ and outputs estimation result $\mathbf{y}_t$. Finally, the estimation results $\{\mathbf{y}_t\}_{t=1,2,\ldots,T}$ are added with the weight parameter $1/T$, and thus the final estimation result $\hat{\mathbf{y}}$ is given by

$$\hat{\mathbf{y}} = \frac{1}{T}\sum_{t=1}^{T}\mathbf{y}_t. \tag{32}$$

Since each subnetwork is considered to perform an independent sampling and estimation process, the final estimation bias can be significantly reduced by taking the average value as the final output.

E. Complexity Analysis

The complexity of the subnetwork is measured by floating-point operations (FLOPs), which is mainly related to the convolutional and dense layers. Other layers, such as the normalization, the addition, and the pooling layers, are several orders of magnitude lower than the convolutional and dense layers [39]. For convolutional layers, the complexity of the lth layer is $(2C_{l-1}M_l - 1)W_lC_l$, where W_l is the input width, M_l is the kernel size, C_{l-1} and C_l represent the number of channels for the $(l-1)$th and the lth layers, respectively. For dense layers, the complexity is given by $(2N_{l-1} - 1)N_l$, where N_{l-1} and N_l are the input and output dimensions, respectively. Thus, the overall complexity of the proposed network is given by

$$\text{F} = \sum_{1}^{N_C}\sum_{l=1}^{2}(2C_{l-1}M_l - 1)W_lC_l + 3\sum_{l=1}^{N_D}(2N_{l-1} - 1)N_l \tag{33}$$

where N_D is the number of hidden layers in each FC network.

V. Numerical Results

We consider the downlink of an NB-IoT system, where the subcarrier interval is $f = 15$ kHz, the number of subcarriers is $K = 12$, and the modulation scheme is quadrature phase shift keying [3]. We utilize $p = 4$ equal pilot sequences, each of which occupies 12 subcarriers. Hence, the dimensions of the original feature sample are 96×2. The pilot sequence is randomly generated to improve the generalization performance [29], [30]. The six-path channel model suggested by COST 207 is selected, of which the propagation delays are $[0, 0.1, 0.2, 0.3, 0.4, 0.5]$ μs, the power attenuations are $[0, -4, -8, -12, -16, -20]$ dB, and the user speed is 1.5 m/s [46]. The impairment parameters for the training samples are randomly selected from $\theta \in [-0.75, 0.75]$,

TABLE I
LAYER SETTING FOR THE SHARED CNN

Type of layer	Output size of the single MTL	Output size of the proposal
Feature Extraction Block 1		
Convolutional	96 × 128, kernel = 1	48 × 96, kernel = 1
Convolutional	96 × 128, kernel = 16	48 × 96, kernel = 8
Normalization	96 × 128	48 × 96
ReLU	96 × 128	48 × 96
Addition	96 × 128	48 × 96
Pooling	48 × 128	24 × 96
Feature Extraction Block 2		
Convolutional	48 × 64, kernel = 1	24 × 64, kernel = 1
Convolutional	48 × 64, kernel = 8	24 × 64, kernel = 8
Normalization	48 × 64	24 × 64
ReLU	48 × 64	24 × 64
Addition	48 × 64	24 × 64
Pooling	24 × 64	24 × 64
Feature Extraction Block 3		
Convolutional	24 × 64, kernel = 1	24 × 48, kernel = 1
Convolutional	24 × 64, kernel = 8	24 × 48, kernel = 4
Normalization	24 × 64	24 × 48
ReLU	24 × 64	24 × 48
Addition	24 × 64	24 × 48
Pooling	12 × 64	12 × 48
Feature Extraction Block 4		
Convolutional	12 × 64, kernel = 1	12 × 24, kernel = 1
Convolutional	12 × 64, kernel = 4	12 × 24, kernel = 4
Normalization	12 × 64	12 × 24
ReLU	12 × 64	12 × 24
Addition	12 × 64	12 × 24
Flatten	256	256
Dense	400	400

$\epsilon_t/\epsilon_r \in [0.8, 1.2]$, and $\phi_t/\phi_r \in [-0.35, 0.35]$ to represent the hardware imperfection of typical IoT devices [6].

We generate $N_s = 2 \times 10^5$, $N_v = 1 \times 10^3$, and $N_t = 1 \times 10^3$ original samples for training, validation, and test data sets, respectively. The length of the sliding window is $L_w = 48$ and the sliding step is $L_s = 1$. Thus, original input can generate $T = 49$ parallel inputs for the MTL subnetworks. The shared CNN and FC networks consist of $N_C = 4$ feature extraction blocks and $N_D = 5$ dense layers, respectively. We adopt the Adam optimizer, where the mini-batch size, the learning rate, and the exponential decay rates are $N_b = 128$, $\eta = 1 \times 10^{-4}$, and $(0.999, 0.9)$ for the initial training, respectively. For the finetuning, the learning rate is set as $\eta = 8 \times 10^{-5}$. The hyperparameter in the loss function is set as $\lambda = 0.1$.

We compare the proposed parallel MTL estimator with the existing methods in terms of estimation accuracy, ability to address coupling effect, symbol error rate (SER) performance, and average computation time. Specifically, we consider the serial estimator [21], the CRE method [29], the isolated transmitter (IsoTx) method [30] as the baselines. In addition, we show the performance of a single MTL to validate the effectiveness of the proposed parallel MTL structure, of which the network parameters are detailed in Tables I and II. According to these settings, the total FLOPs of the single MTL method and the proposed parallel MTL method are 30 million and 5 million, respectively. Besides, the proposed parallel MTL estimator achieves faster convergency than the single MTL method, of which the convergence epochs are around 200 and 1500, respectively.

TABLE II
LAYER SETTING FOR FC NETWORKS

Type of layer	Output size of the single MTL	Output size of the proposal	Activation
Subnetwork $\mathcal{D}_1$			
Dense	100	100	ReLU
Dense	100	100	ReLU
Dense	25	25	ReLU
Dense	2	2	ReLU
Dense	1	1	Linear
Subnetwork $\mathcal{D}_2$ and $\mathcal{D}_3$			
Dense	100	100	ReLU
Dense	100	100	ReLU
Dense	25	25	ReLU
Dense	2	2	ReLU
Dense	2	2	Linear

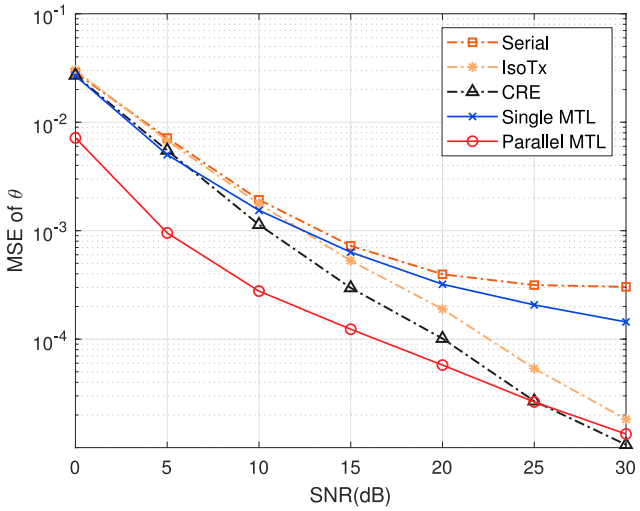
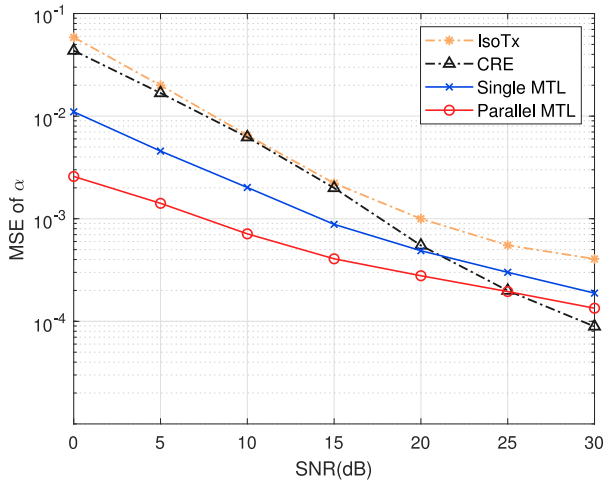


Fig. 3. MSE of CFO parameter θ against different SNRs.

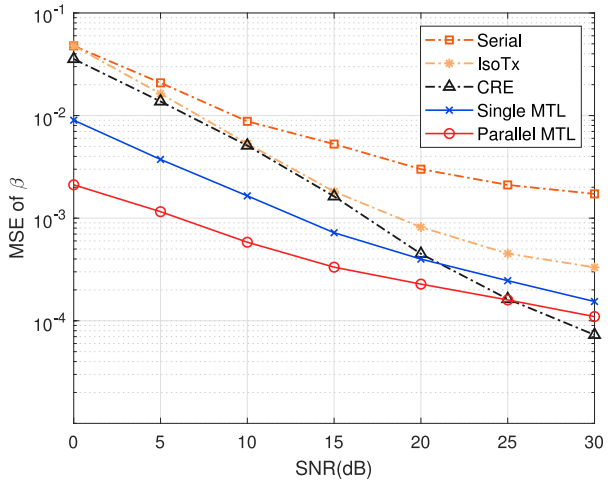
A. Estimation Accuracy

In Fig. 3, we show the MSE of CFO parameter θ against different SNRs, where the impairment parameters are set as $\theta = 0.2$, $\epsilon_t = \epsilon_r = 1.1$, and $\phi_t = \phi_r = 0.18$, and the SNR varies from 0 to 30 dB. As we can see, the proposed parallel MTL estimator achieves the lowest MSE as compared to other methods in the low and medium SNR region of $[0, 20]$ dB. As the proposed estimator applies the weighted loss function to average the task-specific noise patterns, the noise impact is relieved and thus the estimation accuracy is improved. While the conventional methods neglect the noise impact, which leads to severe performance degradation in the low and medium SNR regions.

In the high-SNR region of $[20, 30]$ dB, the MSE of the serial method flattens as it neglects the impact of IQ imbalance on CFO estimation and thus cannot address the coupling effect. The MSEs of the IsoTx and the CRE methods keep decreasing as they can address the coupling effect by using the alternative optimization. The MSEs of the single MTL method and the proposed multiple MTL estimator also keep decreasing as they can alleviate the coupling effect by using the feature extraction blocks to extract the correlated features between CFO and IQ imbalance. Besides, the proposed parallel MTL estimator achieves lower MSEs than the single MTL



(a)



(b)

Fig. 4. MSE of IQ imbalance parameters against different SNRs: (a) MSE of the transmitter IQ imbalance parameter α and (b) MSE of the receiver IQ imbalance parameter β.

in all SNR regions as it utilizes prior knowledge of CFO and IQ imbalance to reduce the input dimension, which further improves the training efficiency and the estimation accuracy.

In Fig. 4(a), we show the MSE of the transmitter IQ imbalance parameter α against different SNRs for all methods except for the serial method, as it only considers the CFO and the receiver IQ imbalance. The hardware impairment parameter settings are the same as in Fig. 3. As we can see, the proposed parallel MTL estimator achieves the lowest MSE as compared to other methods in the low and medium SNR region of $[0, 20]$ dB. As the proposed estimator applies the weighted loss function to average the task-specific noise patterns, the noise impact is relieved and thus the estimation accuracy is improved. Besides, the proposed parallel MTL estimator achieves lower MSE than the single MTL method in all SNR regions as it utilizes prior knowledge of CFO and IQ imbalance to reduce the input dimension, which further improves the estimation accuracy.

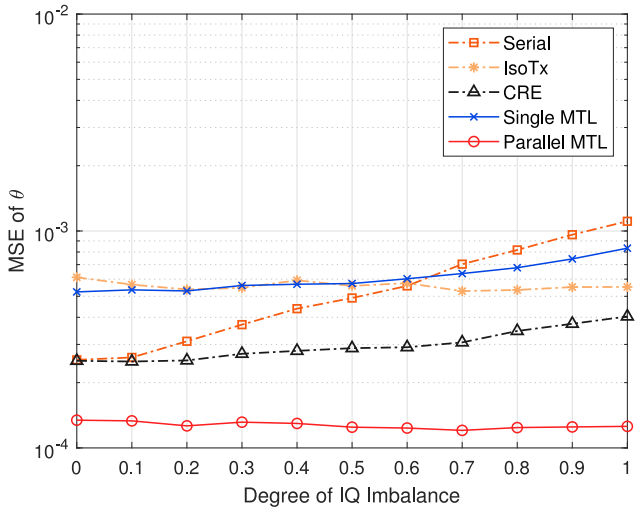


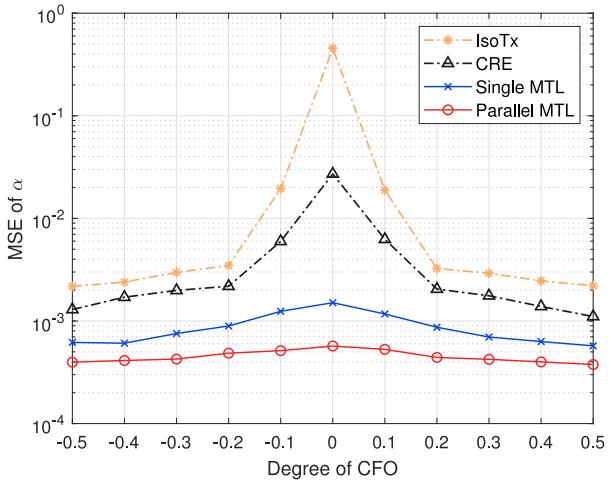
Fig. 5. MSE of CFO parameter θ against different IQ imbalance degrees.

In Fig. 4(b), we show the MSE of the receiver IQ imbalance parameter β against different SNRs for all methods, where the hardware impairment parameter settings are the same as in Fig. 3. As we can see, the proposed parallel MTL estimator achieves the lowest MSE as compared to other methods in the low and medium SNR region of $[0, 20]$ dB and outperforms the single MTL method in all SNR regions for the similar reason given in Fig. 4(a). In the high-SNR region of $[20, 30]$ dB, the MSE of the serial method flattens while the MSEs of the other methods keep decreasing for the similar reasons given in Fig. 3.

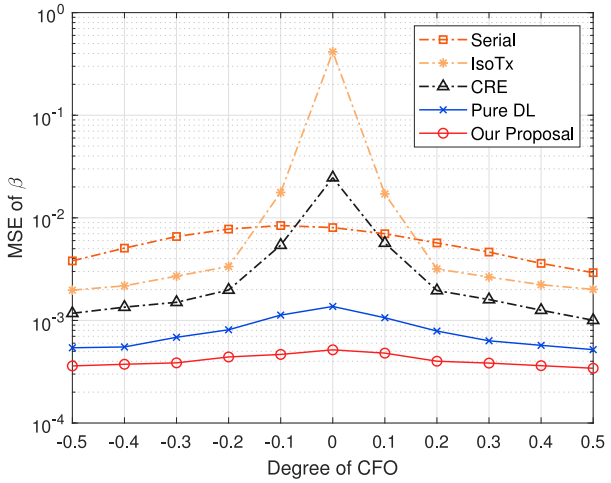
B. Coupling Effect

In Fig. 5, we show the impact of the coupling effect by presenting the MSE of CFO parameter θ as a function of IQ imbalance. The hardware parameters are set as $\theta = 0.2$, $\epsilon_t = \epsilon_r = 1 + 0.15\delta$, and $\phi_t = \phi_r = 0.15\delta$, where $\delta \in [0, 1]$ stands for the degree of IQ imbalance. The SNR is fixed at 15 dB. As we can see, the MSE of the serial method increases dramatically with IQ imbalance degrees since it cannot address the coupling effect. The MSE of the CRE method increases slightly with the increase of the IQ imbalance as it can address the coupling effect by using the alternative optimization. The IsoTx method shows better performance than the CRE method and the MSE keeps unchanged when the IQ imbalance increases since it executes more iterations to minimize the coupling effect. The proposed parallel MTL estimator and the single MTL method can also address the coupling effect as they can utilize the shared feature extraction blocks to extract the correlated features between CFO and IQ imbalance. Hence, they provide similar performance as iterative methods. In addition, the MSE of the proposed parallel MTL estimator is flatter than that of the single MTL method as it applies a parallel structure to average all outputs of the multiple MTL subnetworks, which further decreases the estimation bias.

In Fig. 6(a), we show the MSE of the transmitter IQ imbalance parameter α against different CFOs for all methods except for the serial method since it only considers CFO and



(a)



(b)

Fig. 6. MSE of IQ imbalance parameters against different CFO degrees: (a) MSE of the transmitter IQ imbalance parameter α and (b) MSE of the receiver IQ imbalance parameter β.

the receiver IQ imbalance. The impairment parameters are set as $\theta \in [-0.5, 0.5]$, $\epsilon_t = \epsilon_r = 1.1$, and $\phi_t = \phi_r = 0.18$, while SNR is fixed at 15 dB. As we can see, the MSEs of the IsoTx and the CRE methods are much higher than the single MTL method and the proposed parallel MTL estimator when CFO is small. For the IsoTx and the CRE methods, it is difficult to estimate θ accurately or eliminate the residual CFO when the impact of CFO is relatively small as compared to the IQ imbalance, which deteriorates the estimation accuracy of α. Thus, their MSEs of α decrease with CFO degrees. While the single MTL method and the proposed parallel MTL estimator achieve low MSE of α for any level of θ for the similar reason given in Fig. 3. In addition, the MSEs of our proposal and the single MTL method are lower and flatter, and our proposal outperforms the single MTL method for the similar reason given in Fig. 5.

In Fig. 6(b), we show the MSE of the receiver IQ imbalance parameter β against different CFO degrees. For the serial method, the MSE remains high for all CFOs since it cannot

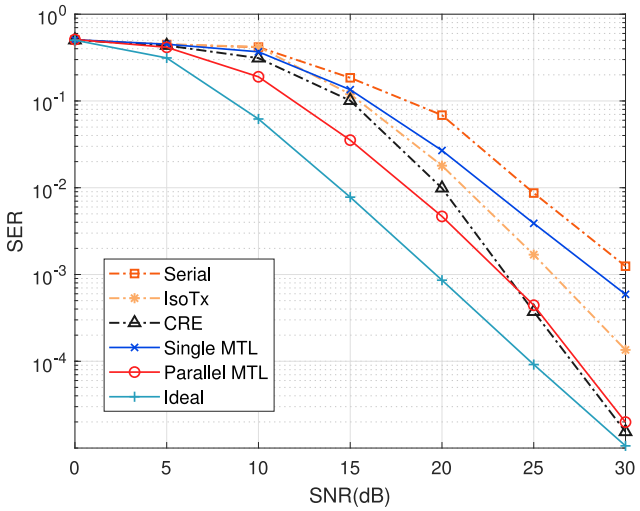


Fig. 7. SER against different SNRs.

effectively address the coupling effect due to the same reason as that of Fig. 4. For the IsoTx and the CRE methods, the MSE of β decreases dramatically when $|\theta|$ varies from 0 to 0.2 and slowly when $|\theta| > 0.2$ for the similar reason given in Fig. 6(a). For the single MTL method and the proposed parallel MTL estimator, they achieve low MSE of β due to the same reason in Fig. 3, and the latter shows better performance than the former for the similar reason given in Fig. 5.

C. Symbol Error Rate

In Fig. 7, we show the SER against different SNRs, where the hardware impairment parameter settings are the same as in Fig. 3. The ideal case with perfect estimation and compensation is also included as the baseline. For the considered estimation methods, we employ the LS algorithm and the zero-forcing algorithm for channel estimation and signal detection, respectively [47]. As we can see, the proposed parallel MTL estimator obtains the lowest SER as compared to other methods in low and medium SNR regions $[0, 20]$ dB since the impairment parameter estimation accuracy increases with SNR, as shown in Figs. 3 and 4. In the high-SNR region of $[20, 30]$ dB, the parallel MTL estimator and the CRE method have similar performance, which are still significantly better than the other methods.

D. Computation Time

In Fig. 8, we compare the computation time using the Windows server with the Intel Core i7 central processing unit (CPU). All results are averaged by 1000 Monte Carlo simulations. The average computation time for the serial, the CRE, the IsoTx, the single MTL, and the proposed methods are 25 μs, 400 μs, 200 μs, 190 μs, and 55 μs, respectively. The serial method is a one-shot method, which only performs several simple algebraic operations with low-computational complexity and thus obtains the lowest computation time. The CRE and the IsoTx methods are iterative methods, which require several iterations for convergence and thus lead to large processing delays. Our proposal achieves a short computation

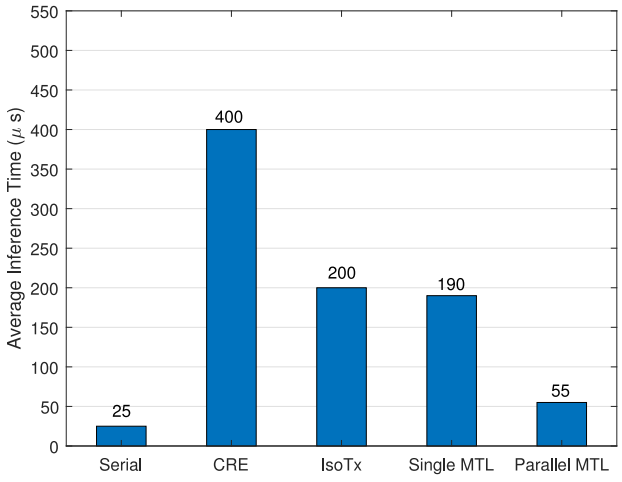


Fig. 8. Average computation time of different algorithms.

time since the data dimension reduction module is utilized to reduce the computational complexity, and thus the online computation time is also reduced, which is better than the IsoTx, the CRE, and the single MTL methods.

Moreover, the proposed parallel MTL estimator can be further accelerated by using parallel processors, such as graphic processor units (GPUs), owing to its natural parallel architecture. Note that the conventional methods cannot be accelerated by the parallel computation capability of GPUs since they adopt either a serial or an alternative framework. When the estimator is deployed on the Nvidia RTX 2060 GPU, the computation time can be reduced to 10 μs.

VI. Conclusion

In this article, we have proposed a parallel MTL estimator to jointly estimate CFO and IQ imbalance in NB-IoT systems. Specifically, the specific structure of the MTL can utilize the correlation between CFO estimation and IQ imbalance estimation, which highly reduces the coupling effect and noise impact. In addition, the parallel structure and the sliding window scheme can effectively reduce the network complexity and decrease the estimation bias. Numerical results show that the proposed parallel MTL estimator can address the coupling effect with short pilot sequences and achieves higher estimation accuracy than the conventional methods in typical SNR regions of IoT devices. Also, the proposed parallel MTL estimator can reduce the computation time as compared to the iterative methods and the single MTL method.

Acknowledgment

The authors would like to thank the editors and the anonymous reviewers, whose invaluable comments helped improve the presentation of this article substantially.

References

[1] J. Xu, J. Yao, L. Wang, Z. Ming, K. Wu, and L. Chen, "Narrowband Internet of Things: Evolutions, technologies, and open issues," *IEEE Internet Things J.*, vol. 5, no. 3, pp. 1449–1462, Jun. 2018.

[2] L. Chettri and R. Bera, "A comprehensive survey on Internet of Things (IoT) toward 5G wireless systems," *IEEE Internet Things J.*, vol. 7, no. 1, pp. 16–32, Jan. 2020.

[3] M. H. Jespersen, M. Pajovic, T. Koike-Akino, Y. Wang, P. Popovski, and P. V. Orlik, "Deep learning for synchronization and channel estimation in NB-IoT random access channel," in *Proc. IEEE GLOBECOM*, Waikoloa, HI, USA, Dec. 2019, pp. 1–7.

[4] A. Chakrapani, "NB-IoT uplink receiver design and performance study," *IEEE Internet Things J.*, vol. 7, no. 3, pp. 2469–2482, Mar. 2020.

[5] Y. Meng, W. Zhang, G. L. Stüber, and W. Wang, "Blind fast CFO estimation and performance analysis for OFDM," *IEEE Trans. Veh. Technol.*, vol. 69, no. 10, pp. 11501–11514, Oct. 2020.

[6] J. Zhang, R. Woods, M. Sandell, M. Valkama, A. Marshall, and J. Cavallaro, "Radio frequency fingerprint identification for narrowband systems, modelling and classification," *IEEE Trans. Inf. Forensics Security*, vol. 16, pp. 3974–3987, 2021.

[7] G. J. González, F. H. Gregorio, and J. Cousseau, "Interference analysis in the LTE and NB-IoT uplink multiple access with RF impairments," in *Proc. IEEE DSP*, Shanghai, China, Feb. 2018, pp. 1–4.

[8] W. Wu, S. Hu, D. Lin, and Z. Liu, "DSLN: Securing Internet of Things through RF fingerprint recognition in low-SNR settings," *IEEE Internet Things J.*, vol. 9, no. 5, pp. 3838–3849, Mar. 2022.

[9] 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, Sep. 2021.

[10] P. H. Moose, "A technique for orthogonal frequency division multiplexing frequency offset correction," *IEEE Trans. Commun.*, vol. 42, no. 10, pp. 2908–2914, Oct. 1994.

[11] J. Li, G. Liu, and G. B. Giannakis, "Carrier frequency offset estimation for OFDM-based WLANs," *IEEE Signal Process. Lett.*, vol. 8, no. 3, pp. 80–82, Mar. 2001.

[12] L. He, S. Ma, Y.-C. Wu, Y. Zhou, T.-S. Ng, and H. V. Poor, "Pilot-aided IQ imbalance compensation for OFDM systems operating over doubly selective channels," *IEEE Trans. Signal Process.*, vol. 59, no. 5, pp. 2223–2233, May 2011.

[13] F. Yan, W.-P. Zhu, and M. O. Ahmad, "Carrier frequency offset estimation and I/Q imbalance compensation for OFDM systems," *EURASIP J. Adv. Signal Process.*, vol. 2007, no. 1, Dec. 2007, Art. no. 45364.

[14] L. Wu, X.-D. Zhang, P.-S. Li, and Y.-T. Su, "A closed-form blind CFO estimator based on frequency analysis for OFDM systems," *IEEE Trans. Commun.*, vol. 57, no. 6, pp. 1634–1637, Jun. 2009.

[15] T. Liu and H. Li, "Joint estimation of carrier frequency offset, DC offset and I/Q imbalance for OFDM systems," *Signal Process.*, vol. 91, no. 5, pp. 1329–1333, Dec. 2011.

[16] C.-W. Chang, Y.-H. Chung, S.-M. Phoong, and Y.-P. Lin, "Joint estimation of CFO and receiver I/Q imbalance using virtual subcarriers for OFDM systems," in *Proc. IEEE PIMRC*, Sydney, NSW, Australia, Sep. 2012, pp. 2297–2302.

[17] Y. Meng, W. Zhang, W. Wang, and H. Lin, "Joint CFO and I/Q imbalance estimation for OFDM systems exploiting constant modulus subcarriers," *IEEE Trans. Veh. Technol.*, vol. 67, no. 10, pp. 10076–10080, Oct. 2018.

[18] T.-C. Lin and S.-M. Phoong, "A low-cost blind estimation of I/Q imbalance in OFDM systems in the presence of CFO," in *Proc. IEEE PIMRC*, Hong Kong, China, Aug. 2015, pp. 72–76.

[19] W. Xu, Y. Wang, and X. Hu, "Blind joint estimation of carrier frequency offset and I/Q imbalance in OFDM systems," *Signal Process.*, vol. 108, pp. 46–55, Mar. 2015.

[20] S. D. Rore, E. Lopez-Estraviz, F. Horlin, and L. V. Der Perre, "Joint estimation of carrier frequency offset and IQ imbalance for 4G mobile wireless systems," in *Proc. IEEE ICC*, Istanbul, Turkey, Jun. 2006, pp. 2066–2071.

[21] J. Tubbax et al., "Joint compensation of IQ imbalance and frequency offset in OFDM systems," in *Proc. IEEE GLOBECOM*, San Francisco, CA, USA, Dec. 2003, pp. 2365–2369.

[22] H. Lin, X. Zhu, and K. Yamashita, "Low-complexity pilot-aided compensation for carrier frequency offset and I/Q imbalance," *IEEE Trans. Commun.*, vol. 58, no. 2, pp. 448–452, Feb. 2010.

[23] J.-R. Liang and C.-H. Kuo, "LS-based joint estimation of carrier frequency offset and IQ imbalance in OFDM systems," in *Proc. Int. Symp. Next Gener. Electron.*, Kaohsiung, Taiwan, Nov. 2010, pp. 52–55.

[24] J.-H. Deng and K.-T. Feng, "Time-frequency multiplex estimator design of joint Tx IQ imbalance, CFO, channel estimation, and compensation for OFDM systems," in *Proc. IEEE TSP*, Prague, Czech Republic, Jul. 2015, pp. 1–5.

[25] Y.-C. Pan and S.-M. Phoong, "A time-domain joint estimation algorithm for CFO and I/Q imbalance in wideband direct-conversion receivers," *IEEE Trans. Wireless Commun.*, vol. 11, no. 7, pp. 2353–2361, Jul. 2012.
[26] F. Wu, Y. Li, and M. Zhao, "Estimation of TX I/Q imbalance at the RX side with RX I/Q imbalance and carrier frequency offset for OFDM systems," in *Proc. IEEE GC Wkshps*, Austin, TX, USA, Dec. 2014, pp. 960–965.
[27] Q. Zou, A. Tarighat, K. Y. Kim, and A. H. Sayed, "OFDM channel estimation in the presence of frequency offset, IQ imbalance and phase noise," in *Proc. IEEE ICASSP*, Honolulu, HI, USA, Apr. 2007, pp. III-273–III-276.
[28] F. Horlin, A. Bourdoux, and E. Lopez-Estraviz, "Low-complexity EM-based joint acquisition of the carrier frequency offset and IQ imbalance," *IEEE Trans. Wireless Commun.*, vol. 7, no. 6, pp. 2212–2220, Jun. 2008.
[29] 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.
[30] 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.
[31] 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.
[32] 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.
[33] 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.
[34] X. Gao, S. Jin, C.-K. Wen, and G. Ye Li, "ComNet: Combination of deep learning and expert knowledge in OFDM receivers," *IEEE Commun. Lett.*, vol. 22, no. 12, pp. 2627–2630, Dec. 2018.
[35] A. Felix, S. Cammerer, S. Dörner, J. Hoydis, and S. Ten Brink, "OFDM-Autoencoder for end-to-end learning of communications systems," in *Proc. IEEE SPAWC Wkshps*, Kalamata, Greece, Jun. 2018, pp. 1–5.
[36] S. Zheng, S. Chen, and X. Yang, "DeepReceiver: A deep learning-based intelligent receiver for wireless communications in the physical layer," *IEEE Trans. Cogn. Commun. Netw.*, vol. 7, no. 1, pp. 5–20, Mar. 2021.
[37] 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*, Xiamen, China, Nov. 2021, pp. 793–798.
[38] 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.
[39] 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, Jul. 2021.
[40] Y. Zhang and Q. Yang, "A survey on multi-task learning," *IEEE Trans. Knowl. Data Eng.*, vol. 34, no. 12, pp. 5586–5609, Dec. 2022.
[41] R. Sebastian, "An overview of multi-task learning in deep neural networks," Jun. 2017, *arXiv:1706.05098*.
[42] H. M. Alonso and P. Barbara, "When is multitask learning effective? Semantic sequence prediction under varying data conditions," in *Proc. EACL*, Valencia, Spain, Apr. 2017, pp. 44–53.
[43] K. Crammer and Y. Mansour, "Learning multiple tasks using shared hypotheses," in *Proc. NIPS*, Dec. 2012, pp. 1–9.
[44] K. Thung and C. Wee, "A brief review on multi-task learning," *Multimed Tools Appl.*, vol. 77, pp. 29705–29725, Aug. 2018.
[45] J. Bingel and A. Søgaard, "Identifying beneficial task relations for multi-task learning in deep neural networks," in *Proc. EACL*, Valencia, Spain, Apr. 2017, pp. 1–6.
[46] M. Failli, "COST 207: Digital and mobile radio communications," Eur. Comm., Luxembourg City, Luxembourg, Rep., Dec. 1989.
[47] F. Michelinakis, A. S. Al-Selwi, M. Capuzzo, A. Zanella, K. Mahmood, and A. Elmokashfi, "Dissecting energy consumption of NB-IoT devices empirically," *IEEE Internet Things J.*, vol. 8, no. 2, pp. 1224–1242, Jan. 2021.

Siqi Liu (Student Member, IEEE) received the B.S. degree from Beijing University of Posts and Telecommunications, Beijing, China, in 2015, and the M.S. degree from the University of Manchester, Manchester, U.K., in 2016. He is currently pursuing the Ph.D. degree with the School of Electronic Science and Engineering, Nanjing University, Nanjing, China.

His current research interests include radio-frequency hardware impairments and machine learning in wireless networks.

Tianyu Wang (Member, IEEE) received the B.S. degrees in physics and a double major in computer software and the Ph.D. degree from the School of Electronics Engineering and Computer Science, Peking University, Beijing, China, in 2011 and 2016, respectively.

He is currently an Assistant Professor with the School of Electronic Science and Engineering, Nanjing University, Nanjing, China. His current research interest focuses on network slicing and machine learning in wireless networks.

Shaowei Wang (Senior Member, IEEE) received the Ph.D. degree from Wuhan University, Wuhan, China, in 2006.

In 2006, he joined the School of Electronic Science and Engineering, Nanjing University, Nanjing, China, as a Faculty Member, where he is currently a Full Professor. From 2012 to 2013, he was a Visiting Scholar/Professor with Stanford University, Stanford, CA, USA, and The University of British Columbia, Vancouver, BC, Canada. His research interests include communications and networking, operations research and machine learning.