IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received March 23, 2020, accepted March 29, 2020, date of publication April 2, 2020, date of current version April 16, 2020.

*Digital Object Identifier 10.1109/ACCESS.2020.2985083*

# Convolutional-Neural-Network-Based Detection Algorithm for Uplink Multiuser Massive MIMO Systems

**LIN LI[1,2], (Member, IEEE), HUIJUN HOU[3], (Member, IEEE), AND WEIXIAO MENG[4], (Senior Member, IEEE)**

[1]Jiangsu Key Laboratory of Broadband Wireless Communication and Internet of Things, Nanjing University of Posts and Telecommunications, Nanjing 210046, China
[2]School of Internet of Things, Nanjing University of Posts and Telecommunications, Nanjing 210046, China
[3]CETC 14th Research Institute, Nanjing 210039, China
[4]Communications Research Center, Harbin Institute of Technology, Harbin 150001, China

Corresponding author: Lin Li (lynn@njupt.edu.cn)

This work was supported by the NUPTSF under Grant NY218127.

**ABSTRACT** In the coming 6th generation (6G) and beyond in wireless communication, an increasing number of ultrascale intelligent factors, including mobile robot users and smart cars, will result in interference exploitation. The management of this exploitation will be a great challenge for detection algorithms in uplink massive multiple-input and multiple-output (MIMO) systems, especially for high-order quadrature amplitude modulation (QAM) signals. Artificial intelligence technology employing machine learning is one of the key approaches among the 6G technical solutions. In this paper, a convolutional-neural-network-based likelihood ascent search (CNNLAS) detection algorithm is proposed on the basis of a graphical detection model for uplink multiuser massive MIMO systems. Compared with other algorithms, the proposed CNNLAS detection algorithm has a stronger robustness against the channel estimation errors, and requires lower average received signal-to-noise ratios to obtain better bit error rate performance and to achieve the theoretical spectral efficiency with a lower polynomial average per symbol computational complexity, both for the graphical low-order and high-order QAM signals in uplink multiuser massive MIMO systems.

**INDEX TERMS** Wireless communication, massive multiple-input and multiple-output (MIMO), convolutional neural network (CNN), detection algorithm, high-order modulation, bit error rate (BER), computational complexity.

## I. INTRODUCTION

The growing number of ultrascale intelligent factors, including mobile robot users and smart cars, create the core requirements for ultrahigh-speed and low-latency communications and innovation in the communication systems architecture, and the 6th generation (6G) and beyond wireless communications are coming [1], [2]. Meanwhile, more aggressive resource sharing and tighter cooperation in these intelligent factors will result in interference exploitation. In uplink massive multiple-input and multiple-output (MIMO) systems, obtaining the optimum bit error rate (BER) performance with a low polynomial computational complexity is a nondeterministic polynomial hard problem [3]–[5]. Interference exploitation and management will be a great challenge for detection algorithms in uplink massive MIMO systems in the future, especially for high-order quadrature amplitude modulation (QAM) signals at an ultrascale. As key technology, the development of a low-complexity detection algorithm for uplink high-order modulated massive MIMO systems is one of the most difficult but urgent issues that needs to be addressed.

Recently, for the low-order modulation signals in massive MIMO systems, including binary phase shift keying and 4-QAM signals, a number of detection algorithms have obtained approximate optimum BER performance using the maximum likelihood algorithm with polynomial computational complexity, including the likelihood ascent

The associate editor coordinating the review of this manuscript and approving it for publication was Adnan M. Abu-Mahfouz.

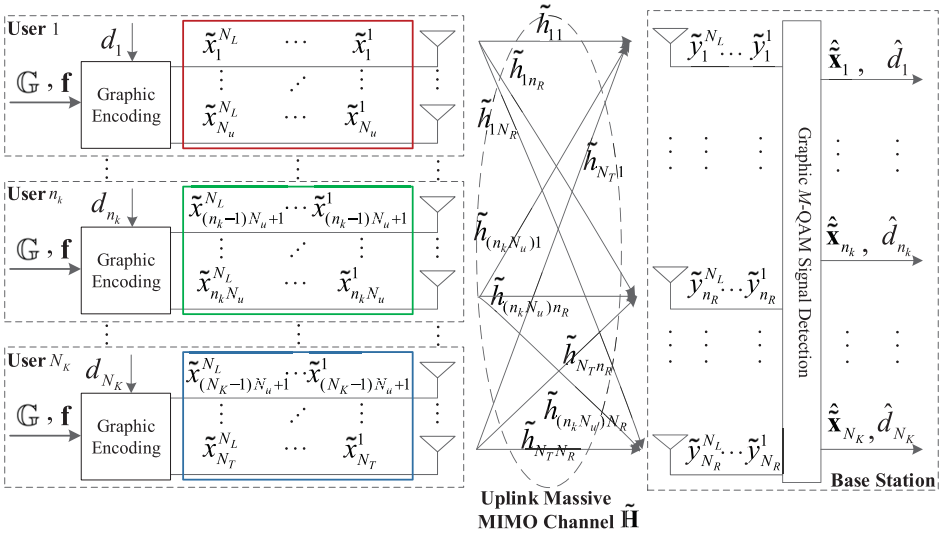


**FIGURE 1.** The graphical detection model for the uplink multiuser massive MIMO system in a cell.

search (LAS) detection algorithm [6]–[8], mixed-Gibbs sampling (MGS) detection method [9], [10] and probabilistic data association detection algorithm [11], [12], belief propagation (BP) detection method [13], [14], and ant colony optimization detection algorithm [15]. However, these methods failed to detect high-order modulation signals due to poor BER performance, including 16-QAM and 64-QAM signals. On the other hand, the semidefinite relaxation decoder (SDR) [16], LLLSDR detection method [17], MGS-MR detection algorithm [18] and hybrid RTS-BP detection algorithm [19] provide solutions for high-order modulation signals in small- or medium-scale MIMO systems. However, the BER performance worsens, and the computational complexity increases with an increase in the antenna number or modulation order. These solutions are not effective for ultrascale massive MIMO systems.

Artificial intelligence (AI) technology employing machine learning is one of the key approaches among 6G technical solutions [1], [2]. In machine learning, deep learning technology has been employed and has achieved many advantages in wireless physical communications [20]–[30], including signal processing and channel estimation. In particular, by means of deep learning technology, the detection network (DetNet) algorithm proposed in [27], [28], orthogonal approximate message passing network (OAMP-Net) detection algorithm in [29], deep-learning-based sphere decoding (DL-SD) algorithm in [26], and multilevel deep neural network (DNN) detection algorithm in [30] provide solutions to the problem of detecting both low-order and high-order modulation signals in small- or medium-scale MIMO systems.

Motivated by this, in this paper, the convolutional neural network (CNN) in deep learning technology is employed for detection in massive MIMO systems. The main contribution of this paper is to propose a CNN-based likelihood ascent search (CNNLAS) detection algorithm on the basis of the presented graphical detection model for uplink multiuser massive MIMO systems. Its effectiveness has been proved through the simulation results. Compared to other algorithms, the proposed CNNLAS algorithm requires a lower average received signal-to-noise ratio (SNR) to obtain a better BER performance and achieve the theoretical spectral efficiency with a lower polynomial computational complexity, both for the graphical low-order and high-order QAM signals in the uplink multiuser massive MIMO systems.

The remainder of this paper is organized as follows. The graphical detection model for the uplink multiuser massive MIMO system is presented in Section II. The proposed CNNLAS detection algorithm is introduced in Section III. Section IV analyzes the computational complexity of the algorithm. Section V presents the simulation results. The conclusion is presented in Section VI.

### Notation

In this paper, the lowercase boldface, uppercase boldface and blackboard bold letter denote a vector, 2-D matrix and 3-D matrix, respectively. The superscripts $(\cdot)^{\mathrm{T}}$ and $(\cdot)^{-1}$, vec$(\cdot)$, unvec$(\cdot)$, $*$ and $\otimes$, $\mathbb{C}$, $\mathbb{R}$ and $\mathbb{Z}$, $\Re(\cdot)$ and $\Im(\cdot)$, $\mathrm{E}\{\cdot\}$, $|\cdot|$, sign$(\cdot)$, $\lfloor\cdot\rceil$, $\mathbf{I}$, $\mathbf{0}_N$, $\mathbf{1}_N$, $\mathbf{1}_{N\times M}$ denote the transpose, inversion, column vectorization operators of a matrix, matrix transformation from a column vector, Hadamard and Kronecker product, complex, real and integer domains, real and imaginary parts of a complex number, statistical expectation, absolute value, sign and round operators, unit diagonal matrix, $N \times 1$ column vector whose entries are all zeros, and $N \times 1$ and $N \times M$ column vector and matrix whose entries are all ones, respectively.

## II. GRAPHICAL DETECTION MODEL FOR UPLINK MULTIUSER MASSIVE MIMO SYSTEM

Fig. 1 shows the graphical detection model for uplink multiuser massive MIMO system in a single cell, which is constructed based on the Vertical Bell Layered Space-Time

(VBLAST) system [4]. In this model, hundreds of receiving antennas are centralized and located at the base station (BS), which receives the signals transmitted from tens of users through the uplink channel. The number of receiving antennas and users in a cell are denoted as $N_R$ and $N_K$, respectively, and their index numbers are denoted as $n_R$ and $n_k$, where $n_R \in \{1, 2, \cdots, N_R\}$ and $n_k \in \{1, 2, \cdots, N_K\}$. The users are randomly distributed in the cell. The transmitting antennas of each user are centralized and are assumed with the same number, which is denoted as $N_{\mathrm{u}}$. The total number of transmitting antennas in a single cell is denoted as $N_T$, where $N_T = N_{\mathrm{u}}N_K$ and $N_T \leq N_R$. The index number of the $n_{\mathrm{u}}$th transmitting antenna from the $n_k$th user is denoted as $n_T = (n_k - 1)N_{\mathrm{u}} + n_{\mathrm{u}}$, where $n_T \in \{1, 2, \cdots, N_T\}$ and $n_{\mathrm{u}} \in \{1, 2, \cdots, N_{\mathrm{u}}\}$. The length of the modulation symbols per frame duration and its index number are denoted as $N_L$ and $n_l$, respectively, where $n_l \in \{1, 2, \cdots, N_L\}$.

In the cell, the graphical signal data center is established with two parts. One part is the 3-D graphical signal matrix and is denoted as $\mathbb{G} = \left[\mathbf{G}^1, \cdots, \mathbf{G}^n, \cdots, \mathbf{G}^N\right] \in \mathbb{A}^{N_{\mathrm{g}} \times N_L \times N}$, where $\mathbf{G}^n \in \mathbb{A}^{N_{\mathrm{g}} \times N_L}$ is the $n$th graphical signal matrix element, $N_{\mathrm{g}} = 2N_{\mathrm{u}}$, $N$ is the total number and $N_K \ll N \leq (\sqrt{M})^{(N_{\mathrm{g}}N_L)}$, and $n \in \{1, 2, \cdots, N\}$ is its index number. Each symbol entry is transformed from the bitstream by $M$-QAM modulation. The modulation constellation is denoted as $\mathbb{S} = \mathbb{A} + j\mathbb{A}$, where $\mathbb{A} = \left\{-(\sqrt{M} - 1), \cdots, -3, -1, 1, 3, \cdots, (\sqrt{M} - 1)\right\}$ represents its real alphabet and $M$ is the modulation order. The other part of the data center is the label vector of the graphical signal, which is denoted as $\mathbf{f} = [f_1, \cdots, f_n, \cdots, f_N] \in \mathbb{R}^{1 \times N}$, where the $n$th entry is $f_n = n$ and labels the $n$th graphical signal. For the BS and users in the cell, each of them is equipped with a graphical signal database, which downloads and stores $\mathbb{G}$ and $\mathbf{f}$ from the graphical signal data center, respectively.

Over the unit frame duration for the $N_K$ users in the cell, each user randomly selects one different graphical signal from the database for graphical encoding at the transmitter. The transmitted graphical signal matrix from the $n_k$th user is denoted as $\tilde{\mathbf{X}}_{n_k} = \left[\tilde{\mathbf{x}}_{n_k}^1, \cdots, \tilde{\mathbf{x}}_{n_k}^{n_l}, \cdots, \tilde{\mathbf{x}}_{n_k}^{N_L}\right] \in \mathbb{S}^{N_{\mathrm{u}} \times N_L}$, where $\tilde{\mathbf{x}}_{n_k}^{n_l} = \left[\tilde{x}_{(n_k-1)N_{\mathrm{u}}+1}^{n_l}, \cdots, \tilde{x}_{(n_k-1)N_{\mathrm{u}}+n_{\mathrm{u}}}^{n_l}, \cdots, \tilde{x}_{n_kN_{\mathrm{u}}}^{n_l}\right]^{\mathrm{T}} \in \mathbb{S}^{N_{\mathrm{u}} \times 1}$ denotes the transmitted graphical signal vector during the $n_l$th symbol time and its entry $\tilde{x}_{(n_k-1)N_{\mathrm{u}}+n_{\mathrm{u}}}^{n_l} \in \mathbb{S}$ denotes the symbol from the $n_{\mathrm{u}}$th transmitting antenna. It is encoded by

$$\tilde{\mathbf{X}}_{n_k} = \Psi \mathbf{G}^{(d_{n_k})} + j\Phi \mathbf{G}^{(d_{n_k})} \tag{1}$$

Herein,

$$\Psi = [1, 0] \otimes \mathbf{I}_{N_{\mathrm{u}}} \in \mathbb{Z}^{N_{\mathrm{u}} \times N_{\mathrm{g}}} \tag{2}$$

and

$$\Phi = [0, 1] \otimes \mathbf{I}_{N_{\mathrm{u}}} \in \mathbb{Z}^{N_{\mathrm{u}} \times N_{\mathrm{g}}} \tag{3}$$

As the $n$th entry of the users' label vector, which is denoted as $\mathbf{d} = \left[d_1, \cdots, d_{n_k}, \cdots, d_{N_K}\right] \in \mathbb{Z}^{1 \times N_K}$, the positive integer $d_{n_k}$ is randomly generated and different in each case, where $1 \leq d_{n_k} \leq N$. Then, the graphical signal transmitted from each user is different from the others and is nonorthogonal.

The graphical signal matrix transmitted from the $N_K$ users is denoted as $\tilde{\mathbf{X}} = \left[\tilde{\mathbf{X}}_1^{\mathrm{T}}, \cdots, \tilde{\mathbf{X}}_{n_k}^{\mathrm{T}}, \cdots, \tilde{\mathbf{X}}_{N_K}^{\mathrm{T}}\right]^{\mathrm{T}} = \left[\tilde{\mathbf{x}}^1, \cdots, \tilde{\mathbf{x}}^{n_l}, \cdots, \tilde{\mathbf{x}}^{N_L}\right] \in \mathbb{S}^{N_T \times N_L}$ under the assumption that the signals are transmitted from each user and received by the BS simultaneously through synchronous processing. The entry $\tilde{\mathbf{x}}^{n_l} = \left[(\tilde{\mathbf{x}}_1^{n_l})^{\mathrm{T}}, \cdots, (\tilde{\mathbf{x}}_{n_k}^{n_l})^{\mathrm{T}}, \cdots, (\tilde{\mathbf{x}}_{N_K}^{n_l})^{\mathrm{T}}\right]^{\mathrm{T}} \in \mathbb{S}^{N_T \times 1}$ denotes the transmitted symbol vector over the $n_l$th symbol duration.

The channel gain matrix is denoted as $\tilde{\mathbf{H}} = [\tilde{\mathbf{h}}_1, \cdots, \tilde{\mathbf{h}}_{n_T}, \cdots, \tilde{\mathbf{h}}_{N_T}] \in \mathbb{C}^{N_R \times N_T}$, where $\tilde{\mathbf{h}}_{n_T} = [\tilde{h}_{1n_T}, \cdots, \tilde{h}_{n_Rn_T}, \cdots, \tilde{h}_{n_Rn_T}]^{\mathrm{T}} \in \mathbb{C}^{N_R \times 1}$, where $\tilde{h}_{n_Rn_T}$ denotes the channel gain from the $n_T$th transmitting antenna to the $n_R$th receiving antenna. Over flat fading channel in the rich-scattering environment, the entries of $\tilde{\mathbf{H}}$ are modeled as independent and identically distributed (iid) complex Gaussian variables with zero mean and unit variance, and are invariant during a frame but change independently from frame to frame. Through the channel estimation procedure, the channel gain matrix has been known at the receiver [4], [5], [31].

Over the unit frame time, the received signal matrix at the BS is denoted as $\tilde{\mathbf{Y}} = \left[\tilde{\mathbf{y}}^1, \cdots, \tilde{\mathbf{y}}^{n_l}, \cdots, \tilde{\mathbf{y}}^{N_L}\right] \in \mathbb{C}^{N_R \times N_L}$, and

$$\tilde{\mathbf{Y}} = \tilde{\mathbf{H}}\tilde{\mathbf{X}} + \tilde{\mathbf{Z}} \tag{4}$$

where $\tilde{\mathbf{y}}^{n_l} = \left[\tilde{y}_1^{n_l}, \cdots, \tilde{y}_{n_R}^{n_l}, \cdots, \tilde{y}_{N_R}^{n_l}\right]^{\mathrm{T}} \in \mathbb{C}^{N_R \times 1}$ denotes the received signal vector during the $n_l$th symbol time. Herein, the entry $\tilde{y}_{n_R}^{n_l}$ is the received symbol at the $n_R$th receiving antenna. $\tilde{\mathbf{Z}} = \left[\tilde{\mathbf{z}}^1, \cdots, \tilde{\mathbf{z}}^{n_l}, \cdots, \tilde{\mathbf{z}}^{N_L}\right] \in \mathbb{C}^{N_R \times N_L}$ denotes the complex additive White Gaussian noise (AWGN) matrix, whose entry $\tilde{\mathbf{z}}^{n_l} \in \mathbb{C}^{N_R \times 1}$ is the received noise vector over the $n_l$th symbol duration. $\mathrm{E}\left\{\tilde{\mathbf{z}}^{n_l}(\tilde{\mathbf{z}}^{n_l})^{\mathrm{H}}\right\} = \sigma^2\mathbf{I}_{N_R}$, where $\sigma^2$ is the noise variance.

At the BS, the convolutional neural network (CNN) is trained by the graphical signal matrix of $\mathbb{G}$ and its label matrix of $\mathbf{B}$ in the graphical signal database. With the trained CNN, a detection algorithm based on the CNN is proposed in this paper. After graphical detection processing at the BS, the estimated transmitted graphical symbol matrix for the $N_K$ users over the unit frame duration is obtained and denoted as $\hat{\tilde{\mathbf{X}}} = \left[\hat{\tilde{\mathbf{X}}}_1^{\mathrm{T}}, \cdots, \hat{\tilde{\mathbf{X}}}_{n_k}^{\mathrm{T}}, \cdots, \hat{\tilde{\mathbf{X}}}_{N_K}^{\mathrm{T}}\right]^{\mathrm{T}} = \left[\hat{\tilde{\mathbf{x}}}^1, \cdots, \hat{\tilde{\mathbf{x}}}^{n_l}, \cdots, \hat{\tilde{\mathbf{x}}}^{N_L}\right] \in \mathbb{S}^{N_T \times N_L}$, where $\hat{\tilde{\mathbf{X}}}_{n_k} \in \mathbb{S}^{N_{\mathrm{u}} \times N_L}$ is the estimated transmitted graphical signal matrix from the $n_k$ user and $\hat{\tilde{\mathbf{x}}}^{n_l} = \left[\hat{\tilde{x}}_1^{n_l}, \cdots, \hat{\tilde{x}}_{n_T}^{n_l}, \cdots, \hat{\tilde{x}}_{N_T}^{n_l}\right]^{\mathrm{T}} \in \mathbb{S}^{N_T \times 1}$ is the estimated transmitted symbol vector over the $n_l$th symbol duration, whose entry $\hat{\tilde{x}}_{n_T}^{n_l}$ denotes the estimated symbol from the $n_T$th transmitting antenna. Meanwhile, the estimated users' label vector is also obtained and represented as $\hat{\mathbf{d}} = \left[\hat{d}_1, \cdots, \hat{d}_{n_k}, \cdots, \hat{d}_{N_K}\right] \in \mathbb{Z}^{1 \times N_K}$. This paper aims to obtain the optimum BER performance over finite polynomial complexity time.

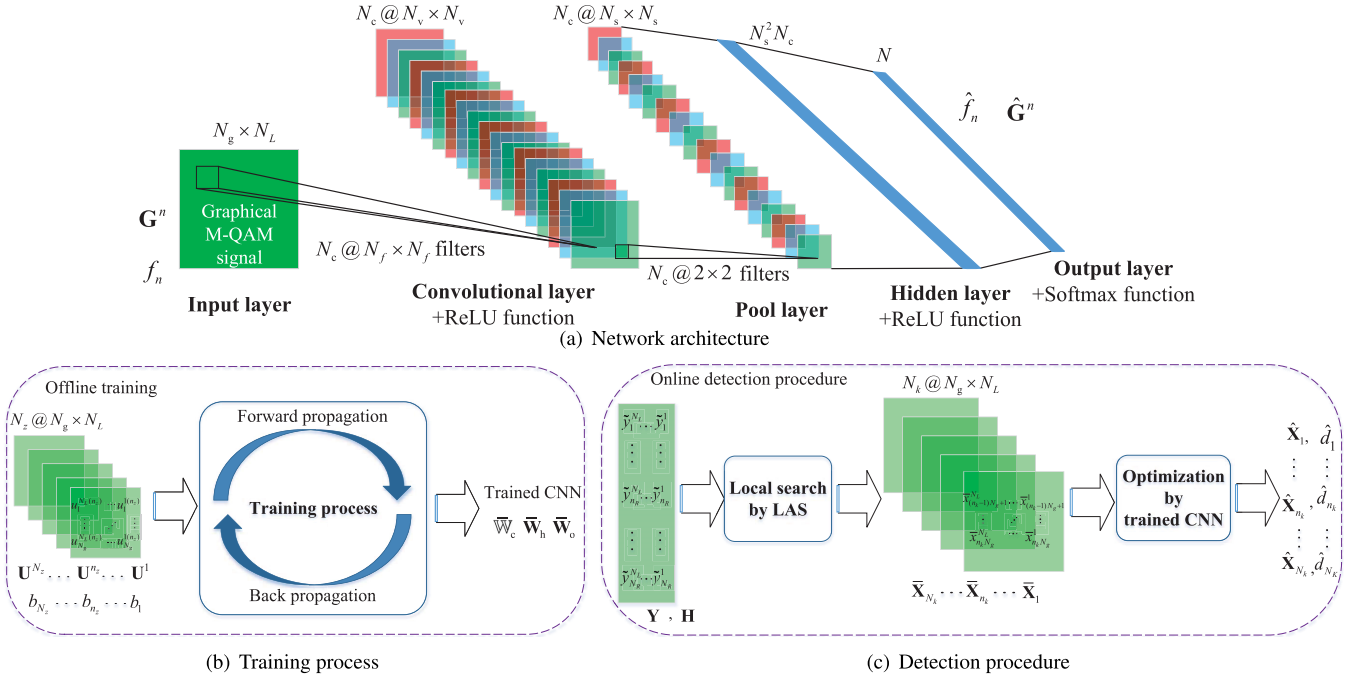


**FIGURE 2.** The network architecture and the diagram for the training process and detection procedure of the proposed CNNLAS detection algorithm.

## III. PROPOSED CONVOLUTIONAL-NEURAL-NETWORK-BASED LIKELIHOOD ASCENT SEARCH DETECTION ALGORITHM

Considering that the algorithm is carried out in the real domain, the complex detection model in (4) is expanded as a real one, that is,

$$\mathbf{Y} = \mathbf{H}\mathbf{X} + \mathbf{Z} \tag{5}$$

Herein,

$$\begin{aligned} \mathbf{H} &\triangleq \begin{bmatrix} \Re(\tilde{\mathbf{H}}) & -\Im(\tilde{\mathbf{H}}) \\ \Im(\tilde{\mathbf{H}}) & \Re(\tilde{\mathbf{H}}) \end{bmatrix}, \quad \mathbf{X} \triangleq \begin{bmatrix} \Re(\tilde{\mathbf{X}}) \\ \Im(\tilde{\mathbf{X}}) \end{bmatrix}, \\ \mathbf{Y} &\triangleq \begin{bmatrix} \Re(\tilde{\mathbf{Y}}) \\ \Im(\tilde{\mathbf{Y}}) \end{bmatrix}, \quad \mathbf{Z} \triangleq \begin{bmatrix} \Re(\tilde{\mathbf{Z}}) \\ \Im(\tilde{\mathbf{Z}}) \end{bmatrix} \end{aligned} \tag{6}$$

where $\mathbf{H} = \left[\mathbf{h}_1, \cdots, \mathbf{h}_{n_t}, \cdots, \mathbf{h}_{N_t}\right] \in \mathbb{R}^{N_r \times N_t}$, $\mathbf{X} = \left[\mathbf{x}^1, \cdots, \mathbf{x}^{n_l}, \cdots, \mathbf{x}^{N_L}\right] \in \mathbb{A}^{N_t \times N_L}$, $\mathbf{Y} = \left[\mathbf{y}^1, \cdots, \mathbf{y}^{n_l}, \cdots, \mathbf{y}^{N_L}\right] \in \mathbb{R}^{N_r \times N_L}$ and $\mathbf{Z} \in \mathbb{R}^{N_r \times N_L}$ denote the expanded real channel gain, transmitted signal, received signal and noise matrix, respectively. Here, $\mathbf{h}_{n_t} = \left[h_{1n_t}, \cdots, h_{n_r n_t}, \cdots, h_{N_r n_t}\right]^{\mathrm{T}} \in \mathbb{R}^{N_r \times 1}$, $\mathbf{y}^{n_l} = \left[y_1^{n_l}, \cdots, y_{n_r}^{n_l}, \cdots, y_{N_r}^{n_l}\right]^{\mathrm{T}} \in \mathbb{R}^{N_r \times 1}$ and $\mathbf{x}^{n_l} = \left[x_1^{n_l}, \cdots, x_{n_t}^{n_l}, \cdots, x_{N_t}^{n_l}\right]^{\mathrm{T}} \in \mathbb{A}^{N_t \times 1}$ represent the expanded real transmitted and received signal vectors over the $n_l$th symbol duration, respectively. Furthermore, $\mathbf{X}_{n_k} = \left[\Re(\tilde{\mathbf{X}}_{n_k}^{\mathrm{T}}), \Im(\tilde{\mathbf{X}}_{n_k}^{\mathrm{T}})\right]^{\mathrm{T}} \in \mathbb{A}^{N_{\mathrm{g}} \times N_L}$, where $N_t = 2N_T$, $N_r = 2N_R$, and $n_t \in \{1, 2, \cdots, N_t\}$, $n_r \in \{1, 2, \cdots, N_r\}$.

### A. TRAINING PROCESS

As shown in Fig. 2(a) and Fig. 2(b), the convolutional neural network (CNN) is trained through $\mathbb{G}$ and $\mathbf{f}$ in the database before the detection procedure.

As shown in Fig. 2(a), considering the computational complexity, the CNN consists of five layers: the input layer, convolutional layer, pool layer, hidden layer and output layer. In the convolutional layer, the weight matrix of the convolutional filter is denoted as $\mathbb{W}_{\mathrm{c}} = \left[\mathbf{W}_{\mathrm{c}}^1, \cdots, \mathbf{W}_{\mathrm{c}}^{n_{\mathrm{c}}}, \cdots, \mathbf{W}_{\mathrm{c}}^{N_{\mathrm{c}}}\right] \in \mathbb{R}^{N_{\mathrm{f}} \times N_{\mathrm{f}} \times N_{\mathrm{c}}}$, where $\mathbf{W}_{\mathrm{c}}^{n_{\mathrm{c}}} \in \mathbb{R}^{N_{\mathrm{f}} \times N_{\mathrm{f}}}$ is the weight matrix of the $n_{\mathrm{c}}$th convolutional filter, $N_{\mathrm{f}}$ denotes its dimension and $N_{\mathrm{c}}$ is the total number of convolutional filters, $n_{\mathrm{c}} \in \{1, 2, \cdots, N_{\mathrm{c}}\}$. The weight matrices between the pool layer and the hidden layer, and between the hidden layer and the output layer are denoted as $\mathbf{W}_{\mathrm{h}} \in \mathbb{R}^{N_{\mathrm{h}} \times N_{\mathrm{d}}}$ and $\mathbf{W}_{\mathrm{o}} \in \mathbb{R}^{N \times N_{\mathrm{h}}}$, where $N_{\mathrm{h}} = \left(\frac{N_L - N_{\mathrm{f}} + 1}{2}\right) N$ and $N_{\mathrm{d}} = \left(\frac{N_L - N_{\mathrm{f}} + 1}{2}\right)^2 N_{\mathrm{c}}$.

As shown in Alg. 1, the training graphical signal matrix inputed into the CNN is given by

$$\mathbb{U} = \mathbf{1}_{\kappa}^{\mathrm{T}} \otimes \mathbb{G} \in \mathbb{A}^{N_L \times N_{\mathrm{u}} \times N_{\mathrm{z}}} \tag{7}$$

Herein, $\mathbb{U} = \left[\mathbf{U}^1, \cdots, \mathbf{U}^{n_{\mathrm{z}}}, \cdots, \mathbf{U}^{N_{\mathrm{z}}}\right]$, whose entry $\mathbf{U}^{n_{\mathrm{z}}} \in \mathbb{A}^{N_L \times N_{\mathrm{u}}}$ denotes the $n_{\mathrm{z}}$th training graphical signal, $N_{\mathrm{z}} = \kappa N$ is its total number, $\kappa$ denotes a positive integer, and $n_{\mathrm{z}} \in \{1, 2, \cdots, N_{\mathrm{z}}\}$. The corresponding label vector of $\mathbb{U}$ is denoted as $\mathbf{B} = \left[\mathbf{b}_1, \cdots, \mathbf{b}_{n_{\mathrm{z}}}, \cdots, \mathbf{b}_{N_{\mathrm{z}}}\right] \in \mathbb{Z}^{N \times N_{\mathrm{z}}}$, whose column vector entry $\mathbf{b}_{n_{\mathrm{z}}} \in \mathbb{Z}^{N \times 1}$ labels the $n_{\mathrm{z}}$th training

graphical signal $\mathbf{U}^{n_z}$, and

$$\mathbf{B} = \mathbf{1}_\kappa^{\mathrm{T}} \otimes \mathbf{F} \tag{8}$$

Herein, $\mathbf{F} \in \mathbb{Z}^{N\times N}$, whose $f_n$th diagonal entry is 1, and the other entries are 0.

**Algorithm 1** Training Process

**Require:** $\mathbb{U} = \mathbf{1}_\kappa^{\mathrm{T}} \otimes \mathbb{G}$, $\mathbf{B} = \mathbf{1}_\kappa^{\mathrm{T}} \otimes \mathbf{F}$, $\mathbb{W}_{\mathrm{c}}^{(0)}$, $\mathbf{W}_{\mathrm{h}}^{(0)}$ and $\mathbf{W}_{\mathrm{o}}^{(0)}$.
1: **for** loop = 1 to $N_{\mathrm{echop}}$ **do**
2: Initialize each entry of $\mathbb{M}_{\mathrm{c}}^{(0)}$, $\mathbf{M}_{\mathrm{h}}^{(0)}$ and $\mathbf{M}_{\mathrm{o}}^{(0)}$ as 0.
3: **for** $n_b = 1$ to $N_b$ **do**
4: Initialize each entry of $\mathbb{L}_{\mathrm{c}}^{(0)}$, $\mathbf{L}_{\mathrm{h}}^{(0)}$ and $\mathbf{L}_{\mathrm{o}}^{(0)}$ as 0.
5: **for** $n_q = n_b$ to $n_b + N_q - 1$ **do**
6: Input $\mathbf{U}^{n_q}$ and $\mathbf{b}_{n_q}$.
7: Calculate $\tilde{\mathbb{V}}_{\mathrm{c}}$, $\mathbb{V}_{\mathrm{c}}$, $\mathbb{V}_{\mathrm{p}}$, $\mathbf{v}_{\mathrm{h}}$, $\tilde{\mathbf{v}}_{\mathrm{h}}$, $\mathbf{v}_{\mathrm{o}}$ successively according from (9) to (15) in the forward propagation.
8: Calculate $\mathbf{e}_{\mathrm{o}}$, $\mathbf{e}_{\mathrm{h}}$, $\mathbb{E}_{\mathrm{p}}$, $\mathbb{E}_{\mathrm{c}}$ successively according from (16) to (21) in the back propagation.
9: Update $\mathbb{L}_{\mathrm{c}}^{(n_q)}$, $\mathbf{L}_{\mathrm{h}}^{(n_q)}$ and $\mathbf{L}_{\mathrm{o}}^{(n_q)}$ as in (22).
10: **end for**
11: Update $\mathbb{M}_{\mathrm{c}}^{(n_b)}$, $\mathbf{M}_{\mathrm{h}}^{(n_b)}$ and $\mathbf{M}_{\mathrm{o}}^{(n_b)}$ as in (23).
12: Update $\mathbb{W}_{\mathrm{c}}^{(n_b)}$, $\mathbf{W}_{\mathrm{h}}^{(n_b)}$ and $\mathbf{W}_{\mathrm{o}}^{(n_b)}$ as in (24).
13: **end for**
14: **end for**
**Ensure:** $\bar{\mathbb{W}}_{\mathrm{c}}$, $\bar{\mathbf{W}}_{\mathrm{h}}$ and $\bar{\mathbf{W}}_{\mathrm{o}}$.

Each entry of $\mathbb{W}_{\mathrm{c}}^{(0)}$, $\mathbf{W}_{\mathrm{h}}^{(0)}$ and $\mathbf{W}_{\mathrm{o}}^{(0)}$ is initialized as a random real number ranging from 0 to $\delta_{\mathrm{c}}$, from $-\delta_{\mathrm{h}}$ to $\delta_{\mathrm{h}}$ and from $-\delta_{\mathrm{o}}$ to $\delta_{\mathrm{o}}$, where $0 \le \delta_{\mathrm{c}} \le 1$, $0 \le \delta_{\mathrm{h}} \le 1$ and $0 \le \delta_{\mathrm{o}} \le 1$, respectively.

During each loop of the training, the CNN is trained by the minibatch method. The number of graphical signals in each batch is $N_{\mathrm{q}} = N_{\mathrm{z}}/N_{\mathrm{b}}$, where $N_{\mathrm{b}}$ denotes the total number of batches. Furthermore, each weight matrix is also updated by the momentum method. As shown in Alg. 1, the superscript "$(n_{\mathrm{b}})$" and "$(n_{\mathrm{q}})$" denote the index number of training iteration, where $n_{\mathrm{b}} \in \{1, 2, \cdots, N_{\mathrm{b}}\}$.

Through the convolutional filters in the convolutional layer, $\mathbf{U}^{n_{\mathrm{q}}}$ is transformed into the feature map matrix, which is denoted as $\tilde{\mathbb{V}}_{\mathrm{c}} = \left[\tilde{\mathbf{V}}_{\mathrm{c}}^1, \cdots, \tilde{\mathbf{V}}_{\mathrm{c}}^{n_{\mathrm{c}}}, \cdots, \tilde{\mathbf{V}}_{\mathrm{c}}^{N_{\mathrm{c}}}\right] \in \mathbb{R}^{N_{\mathrm{v}}\times N_{\mathrm{v}}\times N_{\mathrm{c}}}$, whose $n_{\mathrm{c}}$th entry $\tilde{\mathbf{V}}_{\mathrm{c}}^{n_{\mathrm{c}}} \in \mathbb{R}^{N_{\mathrm{v}}\times N_{\mathrm{v}}}$ denotes the feature map transformed by the $n_{\mathrm{c}}$th convolutional filter, given by

$$\tilde{\mathbf{V}}_{\mathrm{c}(i,j)}^{n_{\mathrm{c}}} = \sum_s \sum_t \mathbf{U}_{(s,t)}^{n_{\mathrm{q}}} \mathbf{W}_{\mathrm{c}(i-s,j-t)}^{n_{\mathrm{c}}(n_{\mathrm{b}}-1)} \tag{9}$$

where the subscript "$(i,j)$" denotes the $(i,j)$th entry of a matrix, and $N_{\mathrm{v}} = (N_L - N_{\mathrm{f}} + 1)$.

Then, the $n_{\mathrm{c}}$th feature map $\tilde{\mathbf{V}}_{\mathrm{c}}^{n_{\mathrm{c}}}$ is activated, given by

$$\mathbf{V}_{\mathrm{c}}^{n_{\mathrm{c}}} = \mathrm{ReLu}\left(\tilde{\mathbf{V}}_{\mathrm{c}}^{n_{\mathrm{c}}}\right) \in \mathbb{R}^{N_{\mathrm{v}}\times N_{\mathrm{v}}} \tag{10}$$

where $\mathrm{ReLu}(x) = \begin{cases} x, x > 0 \\ 0, x \le 0 \end{cases}$ is the activation function with the variable $x$, and $\mathbf{V}_{\mathrm{c}}^{n_{\mathrm{c}}}$ is the $n_{\mathrm{c}}$th entry of the output matrix from the convolutional layer, which is denoted as $\mathbb{V}_{\mathrm{c}} = \left[\mathbf{V}_{\mathrm{c}}^1, \cdots, \mathbf{V}_{\mathrm{c}}^{n_{\mathrm{c}}}, \cdots, \mathbf{V}_{\mathrm{c}}^{N_{\mathrm{c}}}\right] \in \mathbb{R}^{N_{\mathrm{v}}\times N_{\mathrm{v}}\times N_{\mathrm{c}}}$.

Through the $2 \times 2$ mean pooling operation in the pool layer, the $n_{\mathrm{c}}$th activated feature map $\mathbf{V}_{\mathrm{c}}^{n_{\mathrm{c}}}$ is transformed into a smaller map, given by

$$\mathbf{V}_{\mathrm{p}}^{n_{\mathrm{c}}} = \Xi\left(\mathbf{W}_{\mathrm{m}} * \tilde{\mathbf{V}}_{\mathrm{p}}^{n_{\mathrm{c}}}\right)\Xi^{\mathrm{T}} \tag{11}$$

Herein, $\mathbf{V}_{\mathrm{p}}^{n_{\mathrm{c}}} \in \mathbb{R}^{N_{\mathrm{s}}\times N_{\mathrm{s}}}$ is the $n_{\mathrm{c}}$th entry of $\mathbb{V}_{\mathrm{p}} = \left[\mathbf{V}_{\mathrm{p}}^1, \cdots, \mathbf{V}_{\mathrm{p}}^{n_{\mathrm{c}}}, \cdots, \mathbf{V}_{\mathrm{p}}^{N_{\mathrm{c}}}\right] \in \mathbb{R}^{N_{\mathrm{s}}\times N_{\mathrm{s}}\times N_{\mathrm{c}}}$, $\mathbf{W}_{\mathrm{m}} = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} \otimes \mathbf{1}_{N_{\mathrm{s}}\times N_{\mathrm{s}}} \in \mathbb{Z}^{N_{\mathrm{v}}\times N_{\mathrm{v}}}$, $\Xi = \mathbf{I}_{N_{\mathrm{s}}} \otimes \mathbf{1}_2^{\mathrm{T}} \in \mathbb{Z}^{N_{\mathrm{s}}\times N_{\mathrm{v}}}$, $N_{\mathrm{s}} = N_{\mathrm{v}}/2$, and $\tilde{\mathbf{V}}_{\mathrm{p}}^{n_{\mathrm{c}}} \in \mathbb{R}^{N_{\mathrm{v}}\times N_{\mathrm{v}}}$, given by

$$\tilde{\mathbf{V}}_{\mathrm{p}(i,j)}^{n_{\mathrm{c}}} = \sum_s \sum_t \mathbf{V}_{\mathrm{c}(s,t)}^{n_{\mathrm{c}}} \mathbf{W}_{\mathrm{p}(i-s,j-t)} \tag{12}$$

where $\mathbf{W}_{\mathrm{p}} = \frac{1}{4}\mathbf{1}_{2\times 2} \in \mathbb{R}^{2\times 2}$ is the mean pooling matrix.

Then, with the smaller feature map matrix $\mathbb{V}_{\mathrm{p}}$, the output vector from the hidden layer is obtained by

$$\mathbf{v}_{\mathrm{h}} = \mathrm{ReLu}\left(\mathbf{W}_{\mathrm{h}}^{(n_{\mathrm{b}}-1)}\tilde{\mathbf{v}}_{\mathrm{h}}\right) \in \mathbb{R}^{N_{\mathrm{h}}\times 1} \tag{13}$$

where

$$\tilde{\mathbf{v}}_{\mathrm{h}} = \mathrm{vec}\left(\mathbb{V}_{\mathrm{p}}\right) \in \mathbb{R}^{(N_{\mathrm{s}}^2 N_{\mathrm{c}})\times 1} \tag{14}$$

Through the output layer, the output vector is deduced by

$$\mathbf{v}_{\mathrm{o}} = \mathrm{Softmax}\left(\mathbf{W}_{\mathrm{o}}^{(n_{\mathrm{b}}-1)}\mathbf{v}_{\mathrm{h}}\right) \in \mathbb{R}^{N\times 1} \tag{15}$$

Herein, $\mathrm{Softmax}(\chi_n) = \frac{e^{\chi n}}{\sum_{i=1}^{N} \chi_i}$ is the activation function with the variable $\chi_n$, where $\chi_n$ is the entry of $\chi = [\chi_1, \cdots, \chi_n, \cdots, \chi_N]^{\mathrm{T}} \in \mathbb{R}^{N\times 1}$.

With the output vector $\mathbf{v}_{\mathrm{o}}$, the back propagation process is carried out as described below. The error vector from the output layer is obtained by

$$\mathbf{e}_{\mathrm{o}} = \mathbf{b}_{n_{\mathrm{q}}} - \mathbf{v}_{\mathrm{o}} \tag{16}$$

Then, the error vector from the hidden layer is given by

$$\mathbf{e}_{\mathrm{h}} = \left(\frac{\mathrm{sign}\left(\mathbf{v}_{\mathrm{h}}\right) + 1}{2}\right) * \left(\left(\mathbf{W}_{\mathrm{o}}^{(n_{\mathrm{b}}-1)}\right)^{\mathrm{T}} \mathbf{e}_{\mathrm{o}}\right) \tag{17}$$

The error matrix from the pooling layer is denoted as $\mathbb{E}_{\mathrm{p}} = \left[\mathbf{E}_{\mathrm{p}}^1, \cdots, \mathbf{E}_{\mathrm{p}}^{n_{\mathrm{c}}}, \cdots, \mathbf{E}_{\mathrm{p}}^{N_{\mathrm{c}}}\right] \in \mathbb{R}^{N_{\mathrm{v}}\times N_{\mathrm{v}}\times N_{\mathrm{c}}}$, whose $n_{\mathrm{c}}$th entry $\mathbf{E}_{\mathrm{p}}^{n_{\mathrm{c}}} \in \mathbb{R}^{N_{\mathrm{v}}\times N_{\mathrm{v}}}$ is given by

$$\mathbf{E}_{\mathrm{p}}^{n_{\mathrm{c}}} = \left(\tilde{\mathbf{E}}_{\mathrm{p}}^{n_{\mathrm{c}}} \otimes \mathbf{1}_{2\times 2}\right) * \left(\frac{1}{4}\mathbf{1}_{N_{\mathrm{v}}\times N_{\mathrm{v}}}\right) \tag{18}$$

where $\tilde{\mathbf{E}}_{\mathrm{p}}^{n_{\mathrm{c}}} \in \mathbb{R}^{N_{\mathrm{s}}\times N_{\mathrm{s}}}$ is the $n_{\mathrm{c}}$th matrix entry of $\tilde{\mathbb{E}}_{\mathrm{p}} = \left[\tilde{\mathbf{E}}_{\mathrm{p}}^1, \cdots, \tilde{\mathbf{E}}_{\mathrm{p}}^{n_{\mathrm{c}}}, \cdots, \tilde{\mathbf{E}}_{\mathrm{p}}^{N_{\mathrm{c}}}\right] \in \mathbb{R}^{N_{\mathrm{s}}\times N_{\mathrm{s}}\times N_{\mathrm{c}}}$ and $\tilde{\mathbb{E}}_{\mathrm{p}}$ is given by

$$\tilde{\mathbb{E}}_{\mathrm{p}} = \mathrm{unvec}_{N_{\mathrm{s}}\times N_{\mathrm{s}}\times N_{\mathrm{c}}}\left(\left(\mathbf{W}_{\mathrm{h}}^{(n_{\mathrm{b}}-1)}\right)^{\mathrm{T}} \mathbf{e}_{\mathrm{h}}\right) \tag{19}$$

The error matrix from the convolutional layer is denoted as $\mathbb{E}_{\mathrm{c}} = \left[\mathbf{E}_{\mathrm{c}}^{1}, \cdots, \mathbf{E}_{\mathrm{c}}^{n_{\mathrm{c}}}, \cdots, \mathbf{E}_{\mathrm{c}}^{N_{\mathrm{c}}}\right] \in \mathbb{R}^{N_{\mathrm{f}} \times N_{\mathrm{f}} \times N_{\mathrm{c}}}$, whose $n_{\mathrm{c}}$th matrix entry $\mathbf{E}_{\mathrm{c}}^{n_{\mathrm{c}}} \in \mathbb{R}^{N_{\mathrm{f}} \times N_{\mathrm{f}}}$ is given by

$$\mathbf{E}_{\mathrm{c}(i,j)}^{n_{\mathrm{c}}} = \sum_{s} \sum_{t} \mathbf{U}_{(s,t)}^{n_{\mathrm{q}}} \tilde{\mathbf{E}}_{\mathrm{c}(i-s,j-t)}^{n_{\mathrm{c}}} \tag{20}$$

where $\tilde{\mathbf{E}}_{\mathrm{c}}^{n_{\mathrm{c}}} \in \mathbb{R}^{N_{\mathrm{v}} \times N_{\mathrm{v}}}$ is the $n_{\mathrm{c}}$th matrix entry of $\tilde{\mathbb{E}}_{\mathrm{c}} = \left[\tilde{\mathbf{E}}_{\mathrm{c}}^{1}, \cdots, \tilde{\mathbf{E}}_{\mathrm{c}}^{n_{\mathrm{c}}}, \cdots, \tilde{\mathbf{E}}_{\mathrm{c}}^{N_{\mathrm{c}}}\right] \in \mathbb{R}^{N_{\mathrm{v}} \times N_{\mathrm{v}} \times N_{\mathrm{c}}}$ and $\tilde{\mathbb{E}}_{\mathrm{c}}$ is given by

$$\tilde{\mathbb{E}}_{\mathrm{c}} = \left(\frac{\mathrm{sign}\left(\mathbb{V}_{\mathrm{c}}\right) + 1}{2}\right) * \mathbb{E}_{\mathrm{p}} \tag{21}$$

After the $n_{\mathrm{q}}$th training iteration, the delta weight matrices $\mathbb{L}_{\mathrm{c}}^{(n_{\mathrm{q}})} \in \mathbb{R}^{N_{\mathrm{f}} \times N_{\mathrm{f}} \times N_{\mathrm{c}}}$, $\mathbf{L}_{\mathrm{h}}^{(n_{\mathrm{q}})} \in \mathbb{R}^{N_{\mathrm{h}} \times N_{\mathrm{d}}}$ and $\mathbf{L}_{\mathrm{o}}^{(n_{\mathrm{q}})} \in \mathbb{R}^{N \times N_{\mathrm{h}}}$ are updated by

$$\begin{cases} \mathbb{L}_{\mathrm{c}}^{(n_{\mathrm{q}})} = \mathbb{L}_{\mathrm{c}}^{(n_{\mathrm{q}}-1)} + \mathbb{E}_{\mathrm{c}} \\ \mathbf{L}_{\mathrm{h}}^{(n_{\mathrm{q}})} = \mathbf{L}_{\mathrm{h}}^{(n_{\mathrm{q}}-1)} + \mathbf{e}_{\mathrm{h}} \tilde{\mathbf{v}}_{\mathrm{h}}^{\mathrm{T}} \\ \mathbf{L}_{\mathrm{o}}^{(n_{\mathrm{q}})} = \mathbf{L}_{\mathrm{o}}^{(n_{\mathrm{q}}-1)} + \mathbf{e}_{\mathrm{o}} \mathbf{v}_{\mathrm{h}}^{\mathrm{T}}. \end{cases} \tag{22}$$

After the training iteration of the $n_{\mathrm{b}}$th batch, the momentum weight matrices $\mathbb{M}_{\mathrm{c}}^{(n_{\mathrm{b}})} \in \mathbb{R}^{N_{\mathrm{f}} \times N_{\mathrm{f}} \times N_{\mathrm{c}}}$, $\mathbf{M}_{\mathrm{h}}^{(n_{\mathrm{b}})} \in \mathbb{R}^{N_{\mathrm{h}} \times N_{\mathrm{d}}}$ and $\mathbf{M}_{\mathrm{o}}^{(n_{\mathrm{b}})} \in \mathbb{R}^{N \times N_{\mathrm{h}}}$ are updated, given by

$$\begin{cases} \mathbb{M}_{\mathrm{c}}^{(n_{\mathrm{b}})} = \alpha \left(\dfrac{\mathbb{L}_{\mathrm{c}}^{(n_{\mathrm{b}}+N_{\mathrm{q}}-1)}}{N_{\mathrm{q}}}\right) + \beta \mathbb{M}_{\mathrm{c}}^{(n_{\mathrm{b}}-1)} \\ \mathbf{M}_{\mathrm{h}}^{(n_{\mathrm{b}})} = \alpha \left(\dfrac{\mathbf{L}_{\mathrm{h}}^{(n_{\mathrm{b}}+N_{\mathrm{q}}-1)}}{N_{\mathrm{q}}}\right) + \beta \mathbf{M}_{\mathrm{h}}^{(n_{\mathrm{b}}-1)} \\ \mathbf{M}_{\mathrm{o}}^{(n_{\mathrm{b}})} = \alpha \left(\dfrac{\mathbf{L}_{\mathrm{o}}^{(n_{\mathrm{b}}+N_{\mathrm{q}}-1)}}{N_{\mathrm{q}}}\right) + \beta \mathbf{M}_{\mathrm{o}}^{(n_{\mathrm{b}}-1)}. \end{cases} \tag{23}$$

where $\alpha$ is the learning rate and $\beta$ is the momentum parameter, with $0 < \alpha < 1$ and $0 < \beta < 1$.

Then, the weight matrices $\mathbb{W}_{\mathrm{c}}^{(n_{\mathrm{b}})}$, $\mathbf{W}_{\mathrm{h}}^{(n_{\mathrm{b}})}$ and $\mathbf{W}_{\mathrm{o}}^{(n_{\mathrm{b}})}$ are updated accordingly, that is,

$$\begin{cases} \mathbb{W}_{\mathrm{c}}^{(n_{\mathrm{b}})} = \mathbb{W}_{\mathrm{c}}^{(n_{\mathrm{b}}-1)} + \mathbb{M}_{\mathrm{c}}^{(n_{\mathrm{b}})} \\ \mathbf{W}_{\mathrm{h}}^{(n_{\mathrm{b}})} = \mathbf{W}_{\mathrm{h}}^{(n_{\mathrm{b}}-1)} + \mathbf{M}_{\mathrm{h}}^{(n_{\mathrm{b}})} \\ \mathbf{W}_{\mathrm{o}}^{(n_{\mathrm{b}})} = \mathbf{W}_{\mathrm{o}}^{(n_{\mathrm{b}}-1)} + \mathbf{M}_{\mathrm{o}}^{(n_{\mathrm{b}})}. \end{cases} \tag{24}$$

After the training loops by $N_{\mathrm{echop}}$ times, the modified weight matrices of the trained CNN are obtained and denoted as $\bar{\mathbb{W}}_{\mathrm{c}} = \left[\bar{\mathbf{W}}_{\mathrm{c}}^{1}, \cdots, \bar{\mathbf{W}}_{\mathrm{c}}^{n_{\mathrm{c}}}, \cdots, \bar{\mathbf{W}}_{\mathrm{c}}^{N_{\mathrm{c}}}\right]$, $\bar{\mathbf{W}}_{\mathrm{h}}$ and $\bar{\mathbf{W}}_{\mathrm{o}}$.

### B. DETECTION PROCEDURE

As shown in Fig. 2(c) and Alg. 2, through the above trained CNN, the detection procedure is carried out as described below.

Over the $n_l$th symbol duration, through the minimum mean square error (MMSE) detection method, the initial vectors for the local search are denoted as $\mathbf{x}^{n_l(0)}$ and $\tilde{\mathbf{x}}^{n_l(0)}$, given by

$$\begin{cases} \mathbf{x}^{n_l(0)} = \mathcal{Q}\left(\Omega^{-1} \bar{\mathbf{H}}^{\mathrm{T}} \bar{\mathbf{y}}^{n_l}\right) \\ \tilde{\mathbf{x}}^{n_l(0)} = \bar{\mathbf{H}}^{\mathrm{T}} \bar{\mathbf{y}}^{n_l} - \Omega \mathbf{x}_0^{n_l}. \end{cases} \tag{25}$$

**Algorithm 2** Detection procedure

**Require:** $\mathbf{Y}$, $\mathbf{H}$, $\mathbb{G}$, $\mathbf{f}$, $\bar{\mathbb{W}}_{\mathrm{c}}$, $\bar{\mathbf{W}}_{\mathrm{h}}$ and $\bar{\mathbf{W}}_{\mathrm{o}}$.

**for** $n_l = 1$ to $N_L$ **do**
  Initialize $\mathbf{x}^{n_l(0)}$ and $\tilde{\mathbf{x}}^{n_l(0)}$ as in (25) and (26), $\tau = 0$.
  **repeat**
    **for** $q = 1$ to $N_t$ **do**
      Calculate and modified $\rho_q^{n_l(\tau)}$ as in (28) and (29).
      Compute $\Delta F_q^{n_l(\tau+1)}(\rho_q^{n_l(\tau)})$ as in (27).
    **end for**
    Calculate $q'$ as in (31).
    **if** $\Delta F_{q'}^{n_l(\tau+1)}(\rho_{q'}^{n_l(\tau)}) < 0$ **then**
      Update $\mathbf{x}^{n_l(\tau+1)}$ and $\tilde{\mathbf{x}}^{n_l(\tau+1)}$ as in (30).
      $\tau = \tau + 1$.
    **end if**
  **until** $\Delta F_{q'}^{n_l(\tau+1)}(\rho_{q'}^{n_l(\tau)}) \geq 0$
  $\bar{\mathbf{x}}^{n_l} = \mathbf{x}^{n_l(\tau)}$
**end for**
**for** $n_k = 1$ to $N_k$ **do**
  Input $\bar{\mathbf{X}}_{n_k}$ into the trained CNN.
  Calculate $\hat{\mathbb{V}}_c$, $\hat{\mathbb{V}}_p$, $\hat{\mathbf{v}}_h$ and $\hat{\mathbf{v}}_o$ successively according from (32) to (37) through the trained CNN.
  Estimate $\hat{d}_{n_k}$ as in (38).
  Calculate $\hat{\mathbf{X}}_{n_k}$ as in (39).
**end for**

**Ensure:** $\hat{\mathbf{X}}$ and $\hat{\mathbf{d}}$.

Herein,

$$\Omega = \bar{\mathbf{H}}^{\mathrm{T}} \bar{\mathbf{H}} \tag{26}$$

where $\bar{\mathbf{H}} = \begin{bmatrix} \mathbf{H} \\ \sqrt{\frac{\sigma^2}{2}} \mathbf{I}_{N_t} \end{bmatrix}$ and $\bar{\mathbf{y}}^{n_l} = \begin{bmatrix} \mathbf{y}^{n_l} \\ \mathbf{0}_{N_t} \end{bmatrix}$. $\mathcal{Q}(\cdot)$ is the hard decision operator according to $\mathbb{A}$. With the initial value, the local search iterates by likelihood ascent searching proceed as described below.

During the $(\tau + 1)$th iteration, the $q$th difference cost function is obtained by

$$\Delta F_q^{n_l(\tau+1)}(\rho_q^{n_l(\tau)}) = (\rho_q^{n_l(\tau)})^2 r_{q,q} - 2\rho_q^{n_l(\tau)} \left|\tilde{x}_q^{n_l(\tau)}\right| \tag{27}$$

where $r_{q,q}$ is the $q$th diagonal entry of $\Omega$, $\tilde{x}_q^{n_l(\tau)}$ denotes the $q$th entry of $\tilde{\mathbf{x}}^{n_l(\tau)}$, and

$$\rho_q^{n_l(\tau)} = 2\left\lfloor \left|\tilde{x}_q^{n_l(\tau)}\right| \Big/ (2r_{q,q}) \right\rfloor \tag{28}$$

Herein, when $x_q'^{n_l(\tau+1)} = x_q^{n_l(\tau)} + \rho_q^{n_l(\tau)} \mathrm{sign}(v_q^{n_l(\tau)})$ is out of the boundary of $\mathbb{A}$, $\rho_q^{n_l(\tau)}$ is also further modified, given by

$$\rho_q^{n_l(\tau)} = \frac{\mathrm{sign}\left(x_q'^{n_l(\tau+1)}\right)\left(\sqrt{M} - 1\right) - x_q^{n_l(\tau)}}{\mathrm{sign}(\tilde{x}_q^{n_l(\tau)})} \tag{29}$$

where $x_q^{n_l(\tau)}$ denotes the $q$th entry of $\mathbf{x}^{n_l(\tau)}$. $\tau = 0, 1, \cdots$ and $q \in \{1, 2, \cdots, N_t\}$.

When $\Delta F_{q'}^{n_l(\tau+1)}(\rho_{q'}^{n_l(\tau)}) < 0$,

$$\begin{cases} \mathbf{x}^{n_l(\tau+1)} = \mathbf{x}^{n_l(\tau)} + \rho_{q'}^{n_l(\tau)} \text{sign}(\tilde{x}_{q'}^{n_l(\tau)}) \mathfrak{e}_{q'} \\ \tilde{\mathbf{x}}^{n_l(\tau+1)} = \tilde{\mathbf{x}}^{n_l(\tau)} - \rho_{q'}^{n_l(\tau)} \text{sign}(\tilde{x}_{q'}^{n_l(\tau)}) \mathbf{r}_{q'}. \end{cases} \tag{30}$$

Herein,

$$q' = \arg\min_{q} \Delta F_{q}^{n_l(t+1)}(\rho_{q}^{n_l(t)}) \tag{31}$$

$\mathbf{r}_{q'} \in \mathbb{R}^{N_t \times 1}$ is the $q'$th column vector entry of $\Omega$ and $\mathfrak{e}_{q'} \in \mathbb{Z}^{N_t \times 1}$ denotes the vector whose $q'$th entry is 1 and the other entries are 0.

The above iterations are looped until $\Delta F_{q'}^{n_l(\tau+1)}(\rho_{q'}^{n_l(\tau)}) \geq 0$, and the optimum solution vector during the unit symbol time by the local search is obtained, which is denoted as $\bar{\mathbf{x}}^{n_l} = \mathbf{x}^{n_l(\tau)} \in \mathbb{A}^{N_t \times 1}$.

After the frame duration, the estimation of the transmitted signal matrix from the $N_K$ users by the local search is obtained, which is denoted as $\bar{\mathbf{X}} = \left[\bar{\mathbf{x}}^1, \cdots, \bar{\mathbf{x}}^{n_l}, \cdots, \bar{\mathbf{x}}^{N_l}\right] = \left[\left(\bar{\mathbf{X}}_1^{\mathrm{T}}\right)^{\mathrm{T}}, \cdots, \left(\bar{\mathbf{X}}_{n_k}^{\mathrm{T}}\right)^{\mathrm{T}}, \cdots, \left(\bar{\mathbf{X}}_{N_k}^{\mathrm{T}}\right)^{\mathrm{T}}\right] \in \mathbb{A}^{N_t \times N_L}$, whose entry $\bar{\mathbf{X}}_{n_k} \in \mathbb{A}^{N_g \times N_L}$ is the corresponding estimation of the graphical signal from the $n_k$th user.

Then, for a further optimization, $\bar{\mathbf{X}}_{n_k}$ is inputted to the trained CNN from $n_k = 1$ to $n_k = N_K$.

In the trained CNN, through the transformation from $\bar{\mathbf{X}}_{n_k}$ by the convolutional filters, the output activated feature map matrix from the convolutional layer is denoted as $\hat{\mathbb{V}}_{\mathrm{c}} = \left[\hat{\mathbf{V}}_{\mathrm{c}}^1, \cdots, \hat{\mathbf{V}}_{\mathrm{c}}^{n_{\mathrm{c}}}, \cdots, \hat{\mathbf{V}}_{\mathrm{c}}^{N_{\mathrm{c}}}\right] \in \mathbb{R}^{N_{\mathrm{v}} \times N_{\mathrm{v}} \times N_{\mathrm{c}}}$, whose $n_{\mathrm{c}}$th activated feature map matrix entry $\hat{\mathbf{V}}_{\mathrm{c}}^{n_{\mathrm{c}}}$ is given by

$$\hat{\mathbf{V}}_{\mathrm{c}}^{n_{\mathrm{c}}} = \text{ReLu}\left(\hat{\bar{\mathbf{V}}}_{\mathrm{c}}^{n_{\mathrm{c}}}\right) \in \mathbb{R}^{N_{\mathrm{v}} \times N_{\mathrm{v}}} \tag{32}$$

where $\hat{\bar{\mathbf{V}}}_{\mathrm{c}}^{n_{\mathrm{c}}} \in \mathbb{R}^{N_{\mathrm{v}} \times N_{\mathrm{v}}}$ is the $n_{\mathrm{c}}$th entry of $\hat{\bar{\mathbb{V}}}_{\mathrm{c}} \in \mathbb{R}^{N_{\mathrm{v}} \times N_{\mathrm{v}} \times N_{\mathrm{c}}}$, given by

$$\hat{\bar{\mathbf{V}}}_{\mathrm{c}(i,j)}^{n_{\mathrm{c}}} = \sum_{s} \sum_{t} \bar{\mathbf{X}}_{n_k(s,t)} \bar{\mathbf{W}}_{\mathrm{c}(i-s,j-t)}^{n_{\mathrm{c}}} \tag{33}$$

Through the pool layer, the output smaller feature map matrix is obtained, which is denoted as $\hat{\mathbb{V}}_{\mathrm{p}} = \left[\hat{\mathbf{V}}_{\mathrm{p}}^1, \cdots, \hat{\mathbf{V}}_{\mathrm{p}}^{n_{\mathrm{c}}}, \cdots, \hat{\mathbf{V}}_{\mathrm{p}}^{N_{\mathrm{c}}}\right] \in \mathbb{R}^{N_{\mathrm{s}} \times N_{\mathrm{s}} \times N_{\mathrm{c}}}$, whose $n_{\mathrm{c}}$th matrix entry $\hat{\mathbf{V}}_{\mathrm{p}}^{n_{\mathrm{c}}} \in \mathbb{R}^{N_{\mathrm{s}} \times N_{\mathrm{s}}}$ is given by

$$\hat{\mathbf{V}}_{\mathrm{p}}^{n_{\mathrm{c}}} = \Xi\left(\mathbf{W}_{\mathrm{m}} * \hat{\bar{\mathbf{V}}}_{\mathrm{p}}^{n_{\mathrm{c}}}\right) \Xi^{\mathrm{T}} \tag{34}$$

where $\hat{\bar{\mathbf{V}}}_{\mathrm{p}}^{n_{\mathrm{c}}} \in \mathbb{R}^{N_{\mathrm{v}} \times N_{\mathrm{v}}}$ is the $n_{\mathrm{c}}$th entry of $\hat{\bar{\mathbb{V}}}_{\mathrm{p}} \in \mathbb{R}^{N_{\mathrm{v}} \times N_{\mathrm{v}} \times N_{\mathrm{c}}}$, given by

$$\hat{\bar{\mathbf{V}}}_{\mathrm{p}(i,j)}^{n_{\mathrm{c}}} = \sum_{s} \sum_{t} \hat{\mathbf{V}}_{\mathrm{c}(s,t)}^{n_{\mathrm{c}}} \mathbf{W}_{\mathrm{p}(i-s,j-t)} \tag{35}$$

The output vector from the hidden layer is denoted as $\hat{\mathbf{v}}_{\mathrm{h}} \in \mathbb{R}^{N_{\mathrm{h}} \times 1}$, given by

$$\hat{\mathbf{v}}_{\mathrm{h}} = \text{ReLu}\left(\bar{\mathbf{W}}_{\mathrm{h}}\left(\text{vec}\left(\hat{\mathbb{V}}_{\mathrm{p}}\right)\right)\right) \tag{36}$$

Through the output layer, the output vector of the trained CNN is denoted as $\hat{\mathbf{v}}_{\mathrm{o}} \in \mathbb{R}^{N \times 1}$, that is,

$$\hat{\mathbf{v}}_{\mathrm{o}} = \text{Softmax}\left(\bar{\mathbf{W}}_{\mathrm{o}} \hat{\mathbf{v}}_{\mathrm{h}}\right) \tag{37}$$

The estimated label number of the $n_k$th user of $\hat{d}_{n_k}$ is deduced by

$$\hat{d}_{n_k} = \arg\max_{n} \hat{\mathbf{v}}_{\mathrm{o}} \tag{38}$$

where $n \in \{1, 2, \cdots, N\}$.

After the optimization by the trained CNN, the optimum solution of the $n_k$th graphical signal matrix by the proposed CNNLAS detection algorithm is obtained by

$$\hat{\mathbf{X}}_{n_k} = \Psi \mathbf{G}^{(\hat{d}_{n_k})} + j\Phi \mathbf{G}^{(\hat{d}_{n_k})} \tag{39}$$

From $n_k = 1$ to $n_k = N_K$, the optimum solution of the graphical signal matrix of $\hat{\mathbf{X}}$ from the $N_K$ users and the optimum estimation of its label vector of $\hat{\mathbf{d}}$ are obtained by the CNNLAS detection algorithm.

## IV. COMPUTATIONAL COMPLEXITY ANALYSIS

The computational complexity is evaluated by the order of magnitude, denoted as $\mathrm{O}(\cdot)$, of the number of float point number operations (flops) [32].

Considering the calculation for $\mathbf{AB}$, where $\mathbf{A} \in \mathbb{R}^{a \times b}$ and $\mathbf{B} \in \mathbb{R}^{b \times c}$, the computational complexity is $(2b-1)ac \approx \mathrm{O}(abc)$ [32].

Considering one convolutional layer, the computational complexity is $\mathrm{O}(p^2q^2st)$, where $p^2$ is the scale of each feature map output from one convolutional filter, $q^2$ is the scale of each convolutional filter, $s$ is the number of input channels for each filter, and $t$ is the number of convolutional filters in the layer [33].

As shown in Tab. 1, the computational complexity for the proposed algorithm arises from two processes. The first process is the convolutional neural network training, and the second process is the detection procedure.

### A. THE COMPUTATIONAL COMPLEXITY OF THE TRAINING PROCESS

The computational complexity of the convolutional layer from (9) and (10) is $\mathrm{O}(N_v^2 N_f^2 N_c)$. The computational complexity of the pool layer from (11) and (12) is $\mathrm{O}(N_v^2 N_c^2)$. The computational complexity from the hidden layer from (13) and (14) is $\mathrm{O}(N_h N_d)$. The computational complexity of the output layer from (17) is $\mathrm{O}(N_h N)$. Thus, the computational complexity during the forward propagation process is $\mathrm{O}(N_v^2 N_f^2 N_c) + \mathrm{O}(N_v^2 N_c^2) + \mathrm{O}(N_h N_d) + \mathrm{O}(N_h N) = \mathrm{O}(4N_s^2 N_f^2 N_c) + \mathrm{O}(4N_s^2 N_c^2) + \mathrm{O}(N_s^3 N_c N) + \mathrm{O}(N_s N^2)$.

The computational complexity of the procedure of the back propagation from (16) to (21) is on the same order as that of the forward propagation process, that is, $\mathrm{O}(4N_s^2 N_f^2 N_c) + \mathrm{O}(4N_s^2 N_c^2) + \mathrm{O}(N_s^3 N_c N) + \mathrm{O}(N_s N^2)$. The number of batches is $N_b$, and the number of graphical signals updated during each batch is $N_q$.

**TABLE 1.** The computational complexity of the proposed CNNLAS detection algorithm.

| Main calculation part | CNN training | Detection procedure | | |
|---|---|---|---|---|
| | | Initial solution calculation | Local search of LAS | Optimization by the trained CNN |
| Computational complexity | $\mathrm{O}\left(N_z(N_c^2+N^2)\right)$ | $\mathrm{O}\left(N_T^3\right)$ | $\mathrm{O}\left(N_T^3\right)$ | $\mathrm{O}\left(N_T(N_c^2+N^2)\right)$ |

Thus, considering that $N \gg N_f$ and $N \gg N_s$, the total computational complexity of the training process during each loop is

$$2N_bN_q\begin{pmatrix}\mathrm{O}(4N_s^2N_f^2N_c)+\mathrm{O}(4N_s^2N_c^2)\\+\mathrm{O}(N_s^3N_cN)+\mathrm{O}(N_sN^2)\end{pmatrix}\approx \mathrm{O}\left(N_z(N_c^2+N^2)\right) \quad (40)$$

### B. THE COMPUTATIONAL COMPLEXITY OF THE DETECTION PROCEDURE

As shown in Tab. 1, the computational complexity of the detection procedure arises from three processes.

First, the computational complexity of calculating $\Omega$ as in (26) is $\mathrm{O}\left(N_T^2N_R\right)$. When $N_T = N_R$, this complexity is approximated as $\mathrm{O}\left(N_T^3\right)$. Then, the computational complexity of the initial solution calculation as in (25) is $\mathrm{O}\left(N_T^3\right)+\mathrm{O}\left(N_T^3\right)+\mathrm{O}\left(N_T^3\right)+\mathrm{O}\left(N_T^2\right)+\mathrm{O}\left(N_T^2\right)+\mathrm{O}\left(N_T^2\right) = \mathrm{O}\left(3N_T^3+3N_T^2\right)$. When $N_T \geq 100$, this complexity is approximated as $\mathrm{O}\left(N_T^3\right)$.

Second, the computational complexity of the local search of the LAS from (27) to (31) mainly arises from the calculation of $\Omega$ as in (26), which is $\mathrm{O}\left(N_T^3\right)$.

Third, the computational complexity of the further optimization by the trained CNN from (32) to (39) is the same as that of the forward procedure for the $N_K$ users, that is, $N_K(\mathrm{O}(4N_s^2N_f^2N_c) + \mathrm{O}(4N_s^2N_c^2) + \mathrm{O}(N_s^3N_cN) + \mathrm{O}(N_sN^2)) = N_T(\mathrm{O}((4N_s^2N_f^2)/N_uN_c) + \mathrm{O}((4N_s^2/N_u)N_c^2) + \mathrm{O}((N_s^3/N_u)N_cN) + \mathrm{O}((N_s/N_u)N^2))$. Considering $N_T >> N_s$ and $N_T >> N_f$, the complexity is approximated as $\mathrm{O}\left(N_T(N_c^2+N^2)\right)$.

Thus, the total average computational complexity per symbol is deduced as

$$\frac{\mathrm{O}\left(N_T^3\right)+\mathrm{O}\left(N_T^3\right)+\mathrm{O}\left(N_T(N_c^2+N^2)\right)}{N_T}\approx \mathrm{O}\left(N_T^2+N_c^2+N^2\right) \quad (41)$$

## V. SIMULATION RESULTS

The numbers of receiving and transmitting antennas in the massive MIMO system are considered to be $N_T = N_R = 288$, $N_T = 120$ and $N_R = 288$. For each user, the number of transmitting antennas is $N_\mathrm{u} = 8$. The number of users in the unit cell is $N_K = N_T/N_\mathrm{u}$ and ranges from 15 to 36. The number of graphical signals $\mathbb{G}$ in the database is $N = 100$, and its dimension is $N_\mathrm{g} \times N_L$, where $N_\mathrm{g} = 16$ and $N_L = 16$. Each symbol of $\mathbf{G}^{(n)}$ is modulated by $M$-QAM from the randomly generated bitstream, and the label index of $\mathbf{G}^{(n)}$ is $f_n = n$, where $M$ ranges from 4 to 64 and $n \in \{1, 2, \cdots, N\}$. Over unit duration, the graphical signal transmitted from the $n_k$th user $\tilde{\mathbf{X}}_{n_k}$ is generated as in (1), and its label index $d_{n_k}$ is a random integer ranging from 1 to $N$ and different from each other, where $n_k \in \{1, 2, \ldots, N_K\}$. Detailed in Section II, the channel is flat Rayleigh fading. The average received SNR ranges from 0 dB to 46 dB.

Detailed in Section III, the hyper parameters of the CNN are presented as below. The dimension of each filter in the convolutional layer is $N_\mathrm{f} \times N_\mathrm{f}$, where $N_\mathrm{f}$ changes from 3 to 11. The total number is $N_\mathrm{c}$ and changes from 20 to 100. For the initial value of the weight matrix in the hidden layer and the output layer, $\delta_\mathrm{h} = \sqrt{\frac{6}{4N_\mathrm{f}N+N_\mathrm{s}^2N_\mathrm{c}}}$ and $\delta_\mathrm{o} = \sqrt{\frac{6}{N_\mathrm{f}N}}$ [34]. The learning rate is $\alpha = 0.01$ and the momentum parameter is $\beta = 0.95$. During each loop, the training graphical signal inputted into the CNN $\mathbf{U}^{n_z}$ and its label vector $\mathbf{b}_{n_z}$ are generated as in (7) and (8), where $n_z \in \{1, 2, \cdots, N_z\}$. Its total number is $N_\mathrm{z} = N_\mathrm{b}N_\mathrm{q} = 100000$, where the total number of training graphical signals in each batch is $N_\mathrm{q} = 200$ and the number of batches is $N_\mathrm{b} = 500$. The number of training loops is $N_\mathrm{echop} = 3$.

In this section, implemented by programming through MATLAB 2017a, the BER performance, spectral efficiency and computational complexity of the proposed CNNLAS algorithm are simulated and compared with those of the MMSE detection algorithm in [4], MMSELAS detection algorithm in [6], SDR algorithm in [16], LLLSDR algorithm in [17], DetNet detection algorithm in [28], where the number of layers for the DetNet algorithm is $L_D = 90$.

### A. BER PERFORMANCE

#### 1) BER PERFORMANCE OF DETECTING DIFFERENT GRAPHICAL $M$-QAM SIGNALS

Fig. 3 shows the BER performance of the proposed CNNLAS detection algorithm for the graphical 4-QAM signals in the uplink multiuser MIMO system, where $N_T = 32$ and $N_R = 48$. At an average received SNR of 3 dB, the BER of the proposed CNNLAS detection algorithm in the case of $N_f \times N_f = 3\times3$ and $N_c = 20$ is $10^{-5}$, which is much lower than the BER of $10^{-1}$ of the counterpart algorithms. To obtain a BER of $10^{-5}$, the required average received SNR for the proposed CNNLAS algorithm increases by 1 dB when $N_f \times N_f = 3\times3$ increases to $N_f \times N_f = 7 \times 7$ as a result of the generated overfitting but decreases to 2 dB when $N_c = 20$ increases to $N_c = 60$. This average received SNR is nearly 9 dB

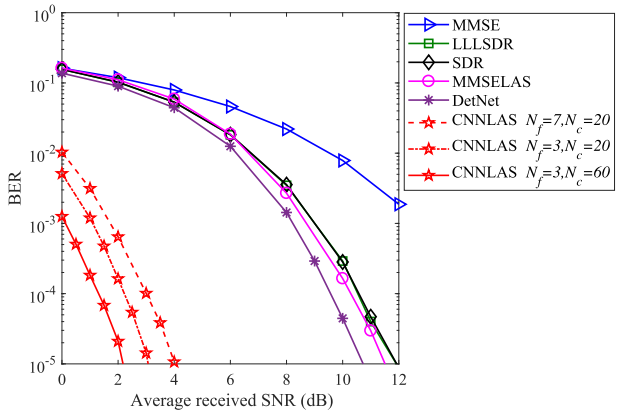


**FIGURE 3.** BER performance of the proposed CNNLAS detection algorithm for graphical 4-QAM signals in uplink multiuser MIMO system, where $N_T = 32$ and $N_R = 48$.

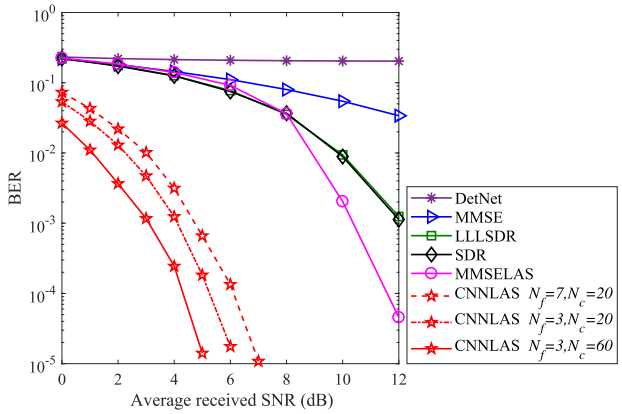


**FIGURE 4.** BER performance of the proposed CNNLAS detection algorithm for graphical 4-QAM signals in uplink multiuser massive MIMO system, where $N_T = N_R = 288$.

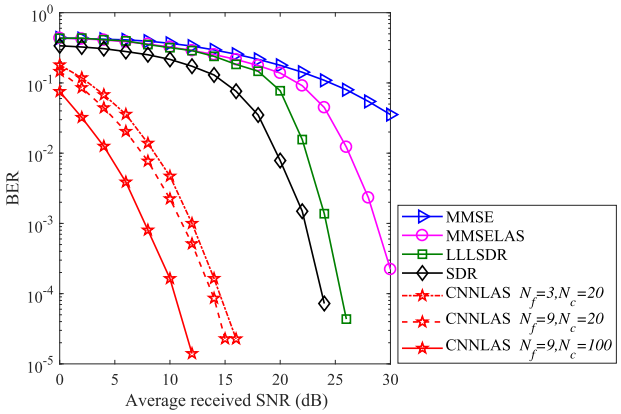


**FIGURE 5.** BER performance of the proposed CNNLAS detection algorithm for graphical 16-QAM signals in uplink multiuser massive MIMO system, where $N_T = N_R = 288$.

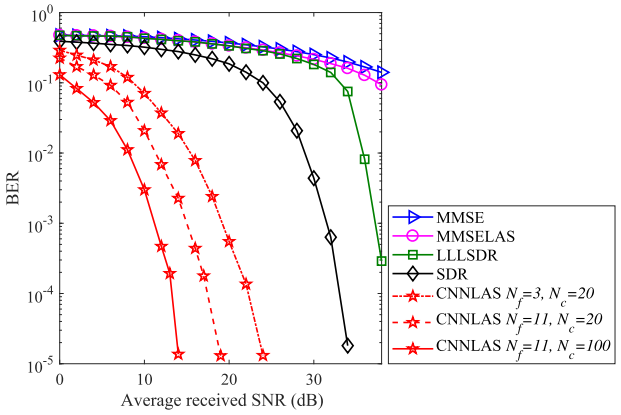


**FIGURE 6.** BER performance of the proposed CNNLAS detection algorithm for graphical 64-QAM signals in uplink multiuser massive MIMO system, where $N_T = N_R = 288$.

lower than that required for the DetNet detection algorithm, 10 dB lower than that required for the MMSELAS, SDR and LLLSDR algorithms, and much lower than that required for the MMSE detection algorithm. Compared to the counterpart algorithms, the proposed CNNLAS detection algorithm obtains a much better BER performance in medium-small MIMO system.

Fig. 4 shows the BER performance of the proposed CNNLAS detection algorithm for the graphical 4-QAM signals in the uplink multiuser massive MIMO system, where $N_T = N_R = 288$. The DetNet detection algorithm obtains a quite worse BER performance of $10^{-1}$ in this case. At an average received SNR of 6 dB, the BER of the proposed CNNLAS detection algorithm in the case of $N_f \times N_f = 3 \times 3$ and $N_c = 20$ is $10^{-5}$, which is much lower than the BER of $10^{-1}$ of the counterpart algorithms. To obtain a BER of $10^{-5}$, the required average received SNR for the proposed CNNLAS algorithm increases by 1 dB when $N_f \times N_f = 3 \times 3$ increases to $N_f \times N_f = 7 \times 7$ as a result of the generated overfitting but decreases to 5 dB when $N_c = 20$ increases to $N_c = 60$. This average received SNR is nearly 7 dB lower than that required for the MMSELAS method and much lower than that required for the other detection algorithms.

Fig. 5 shows the BER performance of the proposed CNNLAS detection algorithm for the graphical 16-QAM signals in the uplink multiuser massive MIMO system, where $N_T = N_R = 288$. At an average received SNR of 16 dB, the BER of the proposed CNNLAS algorithm in the case of $N_f \times N_f = 3 \times 3$ and $N_c = 20$ is $10^{-5}$, which is lower than the BER of $10^{-1}$ of the counterpart detection algorithms. To obtain a BER of $10^{-5}$, the required average received SNR for the proposed CNNLAS detection algorithm decreases by 1 dB when $N_f \times N_f = 3 \times 3$ increases to $N_f \times N_f = 9 \times 9$ and decreases to 12 dB when $N_c = 20$ increases to $N_c = 100$. This average received SNR is 12 dB lower than that required for the SDR method, and 14 dB lower than that for the LLSDR algorithm, more than 18 dB lower than that for the MMSELAS detection algorithm, and much lower than that required for the MMSE algorithm.

Fig. 6 shows the BER performance of the proposed CNNLAS detection algorithm for the graphical 64-QAM signals in the uplink multiuser massive MIMO system, where $N_T = N_R = 288$. At a fixed average received SNR, the BER

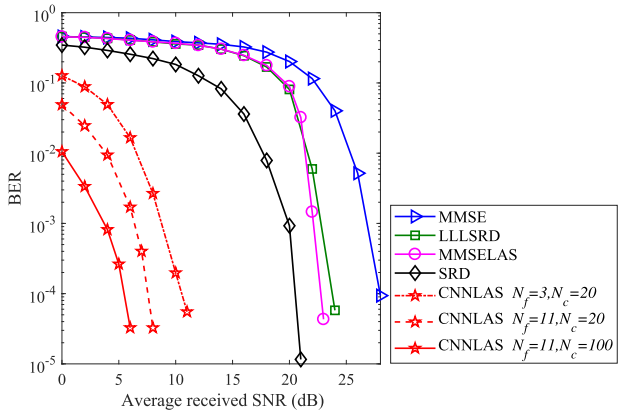


**FIGURE 7.** BER performance of the proposed CNNLAS detection algorithm for graphical 64-QAM signals in uplink multiuser massive MIMO system, where $N_T = 120$ and $N_R = 288$.

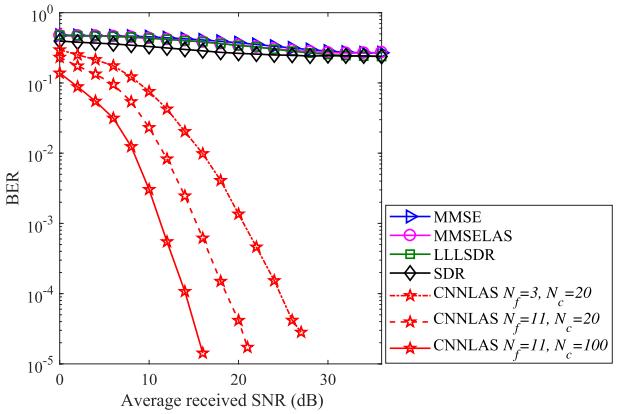


**FIGURE 8.** BER performance of the proposed CNNLAS detection algorithm in presence of channel estimation errors for graphical 64-QAM signals in uplink multiuser massive MIMO system, where $N_T = N_R = 288$ and $\sigma_\varepsilon^2 = 0.1$.

of the proposed CNNLAS detection algorithm is lower than those of the counterpart detection algorithms. To obtain the BER of $10^{-5}$, the required average received SNR for the proposed CNNLAS detection algorithm is 24 dB in the case of $N_f \times N_f = 3 \times 3$ and $N_c = 20$, decreases by 5 dB when $N_f \times N_f = 3 \times 3$ increases to $N_f \times N_f = 11 \times 11$ and further decreases to 14 dB when $N_c = 20$ increases to $N_c = 100$. This average received SNR is 20 dB lower than that required for the SDR method, more than 24 dB lower than that required for the LLLSDR algorithm, and much lower than that required for the MMSELAS and MMSE detection algorithms.

The above-mentioned simulation results illustrate that the proposed CNNLAS detection algorithm is both applicable for graphical low-order and high-order modulation signals in the uplink multiuser medium-small or massive MIMO systems. Under medium-low average received SNR cases, the proposed CNNLAS detection algorithm obtains a much better BER performance than the counterpart algorithms in massive MIMO system when $N_T = N_R$.

### 2) BER PERFORMANCE IN THE UPLINK MULTIUSER MASSIVE MIMO SYSTEMS WHERE $N_T < N_R$

Fig. 7 shows the BER performance of the proposed CNNLAS detection algorithm for the graphical 64-QAM signals in the uplink multiuser massive MIMO system, where $N_T = 120$ and $N_R = 288$. At an average received SNR of 11 dB, the BER of the proposed CNNLAS detection algorithm in the case of $N_f \times N_f = 3 \times 3$ and $N_c = 20$ is $10^{-5}$, which is lower than the BER of $10^{-1}$ of the counterpart algorithms. To achieve a BER of $10^{-5}$, the required average received SNR for the proposed CNNLAS decreases by 3 dB when $N_f \times N_f = 3 \times 3$ increases to $N_f \times N_f = 11 \times 11$ and decreases to 6 dB when $N_c = 20$ increases to $N_c = 100$. This average received SNR is 15 dB lower than that required for the SDR method, more than 18 dB lower than that required for the LLLSDR and MMSELAS algorithms, and much lower than that required for the MMSE algorithm.

At medium low average received SNR, when $N_T < N_R$, the proposed CNNLAS detection algorithm obtains better BER performance than the counterpart detection algorithms for the uplink multiuser massive MIMO system.

### 3) BER PERFORMANCE IN PRESENCE OF CHANNEL ESTIMATION ERRORS

In presence of channel estimation errors, the performance of the proposed CNNLAS algorithm in the uplink multiuser massive MIMO system is investigated. The estimated channel gain matrix is given by [35]

$$\hat{\tilde{\mathbf{H}}} = \tilde{\mathbf{H}} + \Delta\tilde{\mathbf{H}} \in \mathbb{C}^{N_R \times N_T} \tag{42}$$

where $\Delta\tilde{\mathbf{H}} \in \mathbb{C}^{N_R \times N_T}$ is the error matrix with iid complex Gaussian entries with zero mean and $\sigma_\varepsilon^2$ variance.

Fig. 8 illustrates the BER performance of the proposed CNNLAS detection algorithm for the graphical 64-QAM signals in the presence of channel estimation errors as those in (42), where $N_T = N_R = 288$ and $\sigma_\varepsilon^2 = 0.1$, and all other conditions remain the same as in Fig. 6. It can be seen that, to achieve a BER of $10^{-5}$, the required average received SNR for the proposed CNNLAS detection algorithm decreases by 6 dB when $N_f \times N_f = 3 \times 3$ increases to $N_f \times N_f = 11 \times 11$ and decreases to 16 dB when $N_c = 20$ increases to $N_c = 100$. In the presence of channel estimation errors, to achieve the same BER performance, the average received SNRs are respectively 3 dB, 2 dB and 2 dB higher than those required in the cases without channel estimation errors. The BER performance of the proposed CNNLAS algorithm decreases sharply versus the average received SNR, but it is much better than those of the counterpart compared algorithms which obtain a quite worse BER performance of $10^{-1}$.

In the presence of channel estimation errors, the counterpart detection algorithms present poor BER performance, whereas the proposed CNNLAS detection algorithm still obtains good BER performance. The proposed CNNLAS detection algorithm shows a much stronger robustness against channel estimation errors in the uplink multiuser massive MIMO system.

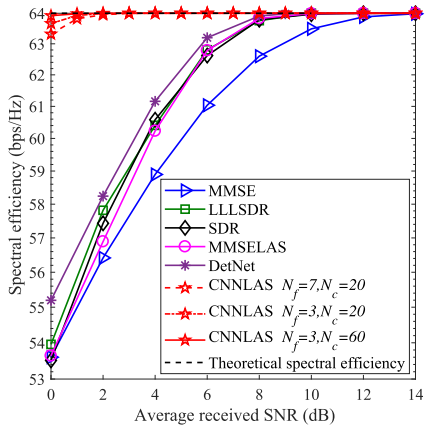


**FIGURE 9.** Spectral efficiency of the proposed CNNLAS detection algorithm for graphical 4-QAM signals in uplink multiuser MIMO system, where $N_T = 32$ and $N_R = 48$.

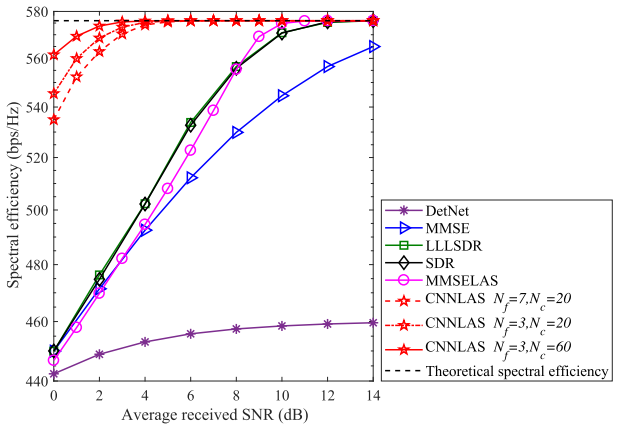


**FIGURE 10.** Spectral efficiency of the proposed CNNLAS detection algorithm for graphical 4-QAM signals in uplink multiuser massive MIMO system, where $N_T = N_R = 288$.

### B. SPECTRAL EFFICIENCY

According to Section II, the graphical detection model for the uplink multiuser massive MIMO system is constructed on the VBLAST system, whose theoretical spectral efficiency for $M$-QAM signals is [36]

$$\mathrm{SE}_{\mathrm{theory}} = N_T \log_2(M) \tag{43}$$

The spectral efficiency of the proposed CNNLAS detection algorithm is also compared with the theoretical value.

#### 1) SPECTRAL EFFICIENCY OF DETECTING DIFFERENT GRAPHICAL $M$-QAM SIGNALS

Fig. 9 shows the spectral efficiency of the proposed CNNLAS detection algorithm for the graphical 4-QAM signals in the uplink multiuser MIMO system, where $N_T = 32$ and $N_R = 48$. The proposed CNNLAS algorithm obtains a much higher spectral efficiency than the counterpart detection algorithms. To obtain the theoretical spectral efficiency of 64 bps/Hz, the required average received SNR for the proposed CNNLAS algorithm in the case of $N_f \times N_f = 3 \times 3$ and $N_c = 20$ is 3 dB, increases by 1 dB when $N_f \times N_f = 3 \times 3$ grows to $N_f \times N_f = 7 \times 7$ as a result of the generated overfitting but decreases to 1 dB when $N_c = 20$ increases to $N_c = 60$. This average received SNR is 8 dB lower than that required for the DetNet detection algorithm, 9 dB lower than that required for the MMSELAS, SDR, and LLLSDR algorithms, and 13 dB lower than that required for the MMSE algorithm. Compared to the counterpart detection algorithms, the proposed CNNLAS detection algorithm requires a lower average received SNR to achieve the theoretical spectral efficiency in the medium-small MIMO systems.

Fig. 10 shows the spectral efficiency of the proposed CNNLAS detection algorithm for the graphical 4-QAM signals in the uplink multiuser massive MIMO system, where $N_T = N_R = 288$. The DetNet detection algorithm obtains a much lower spectral efficiency of 460 bps/Hz than the theoretical value in this case. The proposed CNNLAS algorithm obtains the much higher spectral efficiency than the counterpart detection algorithms. To obtain the theoretical spectral efficiency of 576 bps/Hz, the required average received SNR for the proposed CNNLAS algorithm in the case of $N_f \times N_f = 3 \times 3$ and $N_c = 20$ is 5 dB, increases by 1 dB when $N_f \times N_f = 3 \times 3$ grows to $N_f \times N_f = 7 \times 7$ as a result of the generated overfitting but decreases to 4 dB when $N_c = 20$ increases to $N_c = 60$. This average received SNR is 7 dB lower than that required for the MMSELAS algorithm, 8 dB lower than that required for the SDR and LLLSDR algorithms, and more than 16 dB lower than that required for the MMSE algorithm.

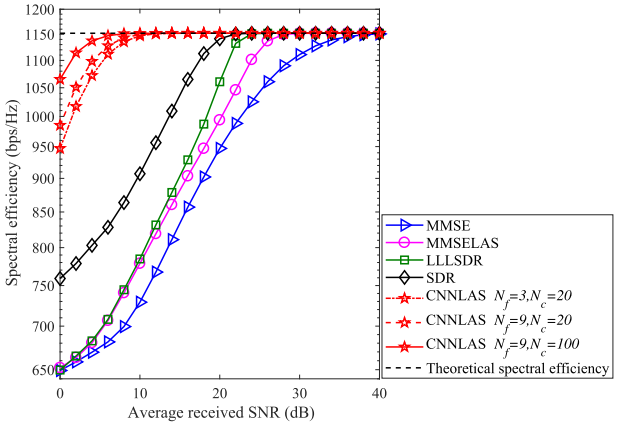


**FIGURE 11.** Spectral efficiency of the proposed CNNLAS detection algorithm for graphical 16-QAM signals in uplink multiuser massive MIMO system, where $N_T = N_R = 288$.

Fig. 11 shows the spectral efficiency of the proposed CNNLAS detection algorithm for the graphical 16-QAM signals in the uplink multiuser massive MIMO system, where $N_T = N_R = 288$. The proposed CNNLAS algorithm obtains a much higher spectral efficiency than that of the counterpart algorithms. To obtain the theoretical spectral efficiency of 1152 bps/Hz, the required average received SNR for the proposed CNNLAS algorithm in the case of $N_f \times N_f = 3 \times 3$ and $N_c = 20$ is 12 dB, decreases by 2 dB when

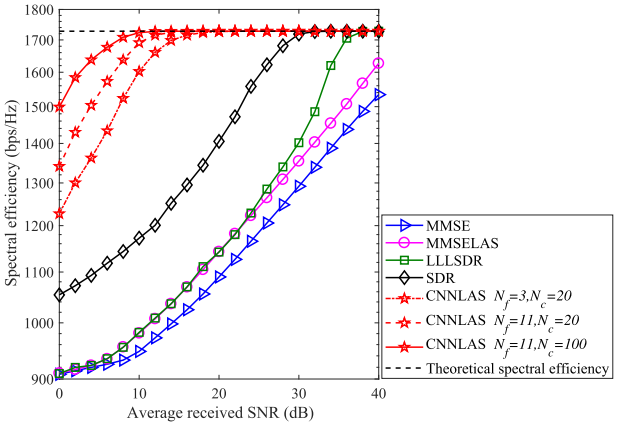


**FIGURE 12.** Spectral efficiency of the proposed CNNLAS detection algorithm for graphical 64-QAM signals in uplink multiuser massive MIMO system, where $N_T = N_R = 288$.

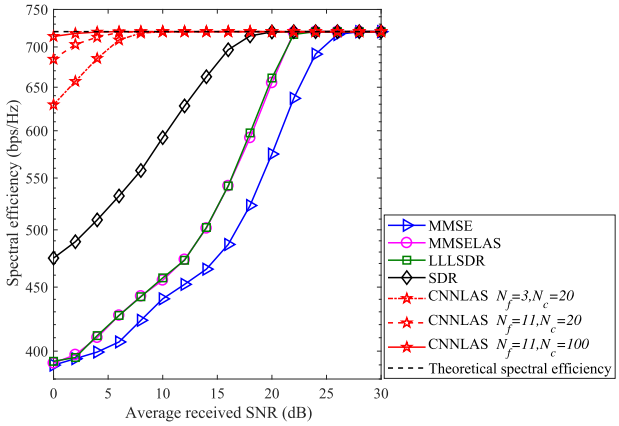


**FIGURE 13.** Spectral efficiency of the proposed CNNLAS detection algorithm for graphical 64-QAM signals in uplink multiuser massive MIMO system, where $N_T = 120$ and $N_R = 288$.

$N_f \times N_f = 3 \times 3$ increases to $N_f \times N_f = 9 \times 9$ and decreases to 8 dB when $N_c = 20$ increases to $N_c = 100$. This average received SNR is 14 dB lower than that required for the SDR algorithm, and 16 dB lower than that required for the LLLSDR method, 20 dB lower than that for the MMSELAS algorithm, and 32 dB lower than that required for the MMSE detection algorithm.

Fig. 12 shows the spectral efficiency of the proposed CNNLAS detection algorithm for the graphical 64-QAM signals in the uplink multiuser massive MIMO system, where $N_T = N_R = 288$. Along with the increase of the average received SNR, the spectral efficiency of the detection algorithms increases until converging to the theoretical value of 1728 bps/Hz. The proposed spectral efficiency of the proposed CNNLAS algorithm is much higher than that of the counterpart algorithms. In order to obtain the theoretical spectral efficiency, the required average received SNR for the proposed CNNLAS detection algorithm in the case of $N_f \times N_f = 3 \times 3$ and $N_c = 20$ is 20 dB, decreases by 4 dB when $N_f \times N_f = 3 \times 3$ increases to $N_f \times N_f = 11 \times 11$, and decreases to 12 dB when $N_c = 20$ increases to $N_c = 100$. This average received SNR is 20 dB lower than that required for the SDR method, 26 dB lower than that required for the LLLSDR method, and more than 34 dB lower than that required for the MMSELAS and MMSE detection algorithms.

The before-mentioned simulation results show that, compared with the counterpart detection algorithms, the proposed CNNLAS detection algorithm requires a lower average received SNR to achieve the theoretical spectral efficiency for uplink multiuser medium-small or massive MIMO system in terms of detecting the graphical low-order and high-order modulation signals.

### 2) SPECTRAL EFFICIENCY IN THE UPLINK MULTIUSER MASSIVE MIMO SYSTEMS WHERE $N_T < N_R$

Fig. 13 shows the spectral efficiency of the proposed CNNLAS detection algorithm for the graphical 64-QAM signals in the uplink multiuser massive MIMO system, where $N_T = 120$ and $N_R = 288$. The proposed spectral efficiency of the proposed CNNLAS algorithm is much higher than that of the counterpart algorithms. To obtain the theoretical spectral efficiency, the required average received SNR for the proposed CNNLAS detection algorithm in the case of $N_f \times N_f = 3 \times 3$ and $N_c = 20$ is 8 dB, decreases by 2 dB when $N_f \times N_f = 3 \times 3$ increases to $N_f \times N_f = 11 \times 11$ and decreases to 4 dB when $N_c = 20$ increases to $N_c = 100$. This average received SNR is 16 dB lower than that required for the SDR method, and 20 dB lower than that for the LLLSDR and MMSELAS algorithms, and more than 24 dB lower than that required for the MMSE detection algorithm.

In the case of $N_T < N_R$, the proposed CNNLAS detection algorithm requires a much lower average received SNR to reach the theoretical spectral efficiency when compared with the counterpart detection algorithms in the uplink multiuser massive MIMO system.

### 3) SPECTRAL EFFICIENCY IN PRESENCE OF CHANNEL ESTIMATION ERRORS

Fig. 14 presents the spectral efficiency of the proposed CNNLAS detection algorithm for the graphical 64-QAM signals in the presence of channel estimation errors as in (42), where $N_T = N_R = 288$ and $\sigma_\varepsilon^2 = 0.1$, and all other conditions remain the same as those in Fig. 12. The results show that the spectral efficiency of the SDR, LLLDR, MMSELAS and MMSE detection algorithms increase along with the average received SNR and converge to 1316 bps/Hz or 1265 bps/Hz, which are much lower than the theoretical value 1728 bps/Hz. To reach the theoretical spectral efficiency, the average received SNR required for the proposed CNNLAS algorithm with channel estimation errors in the case of $N_f \times N_f = 3 \times 3$ and $N_c = 20$ is 22 dB, which is 2 dB higher than that required for the same case of the proposed CNNLAS algorithm without the channel estimation errors. This SNR decreases by 6 dB when $N_f \times N_f = 3 \times 3$ increases to $N_f \times N_f = 11 \times 11$, and decreases to 12 dB

**TABLE 2.** The average per symbol computational complexity of detection algorithms.

| Detection algorithm | SDR | LLLSDR | MMSE | MMSELAS | DetNet | CNNLAS (proposed) |
|---|---|---|---|---|---|---|
| Average per symbol computational complexity | $\mathrm{O}(N_T^{3.5}M + N_T^{2.5}M^{3.5})$ | $\mathrm{O}(N_T^{3.5}(\log_2\sqrt{M})^2 + N_T^{2.5}(\log_2\sqrt{M})^3)$ | $\mathrm{O}(N_T^2)$ | $\mathrm{O}(N_T^2)$ | $\mathrm{O}(ML_DN_T^2)$ | $\mathrm{O}(N_T^2+N^2+N_c^2)$ |

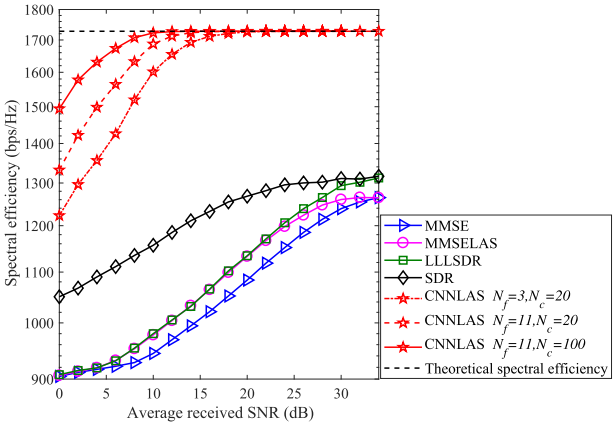


**FIGURE 14.** Spectral efficiency of the proposed CNNLAS detection algorithm in presence of channel estimation errors for graphical 64-QAM signals in uplink multiuser massive MIMO system, where $N_T = N_R = 288$ and $\sigma_e^2 = 0.1$.

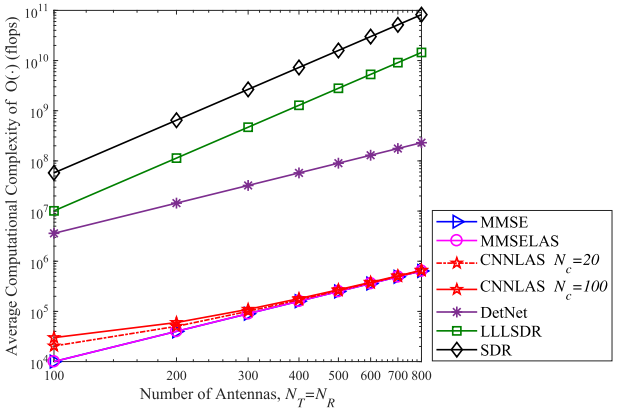


**FIGURE 15.** Average per symbol computational complexity of the proposed CNNLAS detection algorithm versus the number of antennas for graphical 4-QAM signals in uplink multiuser massive MIMO system.

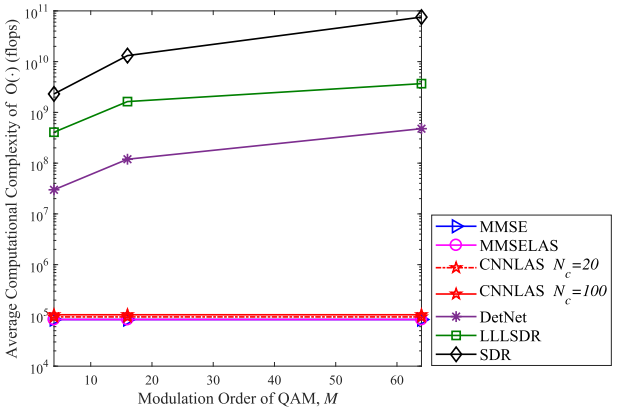


**FIGURE 16.** Average per symbol computational complexity of the proposed CNNLAS detection algorithm versus the modulation order for graphical $M$-QAM signals in uplink multiuser massive MIMO system, where $N_T = N_R = 288$.

when $N_c = 20$ increases to $N_c = 100$, which are the same as those of the proposed CNNLAS algorithm without channel estimation errors.

In the presence of channel estimation errors, the counterpart algorithms are not applicable to the massive MIMO systems since their spectral efficiency is much lower than the theoretical value. The proposed CNNLAS detection algorithm reaches the theoretical spectral efficiency at medium-low average received SNR, and shows strong robustness against the channel estimation errors in the uplink multiuser massive MIMO system.

### C. COMPUTATIONAL COMPLEXITY

The average per symbol computational complexity of the detection algorithms in the uplink multiuser massive MIMO system in terms of O(·) with the number of flops is shown in Tab.2, including the values for the proposed CNNLAS detection algorithm in this paper and the above mentioned counterpart detection algorithms.

#### 1) COMPUTATIONAL COMPLEXITY VERSUS DIFFERENT ANTENNAS

Fig. 15 shows the average per symbol computational complexity of the proposed CNNLAS detection algorithm versus the number of antennas for graphical 4-QAM signals in the uplink multiuser massive MIMO system. The computational complexity of the proposed CNNLAS detection algorithm increase polynomially with the number of antennas. For $N_T = N_R = 100$, the average computational complexity of the CNNLAS algorithm in the case of $N_c = 20$ is $2.0 \times 10^4$, which increases by a small amount but maintains a similar order of magnitude of $3.0 \times 10^4$ when $N_c = 20$ increases to $N_c = 100$. For $N_T = N_R = 800$, the computational complexities of the CNNLAS detection algorithm in the two cases are $6.5 \times 10^5$ and $6.6 \times 10^5$, which are on a similar order of magnitude to the complexities of $6.4 \times 10^5$ of the MMSE and MMSELAS algorithms, but much lower than the complexity of $2.3 \times 10^8$ of the DetNet detection algorithm, the complexity of $1.5 \times 10^{10}$ of the LLLSDR method and the complexity of $8.2 \times 10^{10}$ of the SDR algorithm.

#### 2) COMPUTATIONAL COMPLEXITY OF DETECTING DIFFERENT GRAPHICAL $M$-QAM SIGNALS

Fig. 16 shows the average per symbol computational complexity of the proposed CNNLAS detection algorithm versus the modulation order $M$ for graphical $M$-QAM signals in the uplink multiuser massive MIMO system, where $N_T = N_R = 288$. The computational complexities of the DetNet,

**TABLE 3.** The comparison of the computational complexity and the performance including the BER and spectral efficiency of the proposed CNNLAS detection algorithm with those of other detection algorithms for graphical $M$-QAM signals in the uplink multiuser massive MIMO system where $N_T = N_R = 288$.

| | Detection algorithm | CNNLAS (proposed) | MMSE | MMSELAS | LLLSDR | SDR | DetNet |
|---|---|---|---|---|---|---|---|
| Graphical 4-QAM signals | Average received SNR required to obtain the BER of $10^{-5}$ | 5 dB | 26 dB | 12 dB | 14 dB | 14 dB | $>40$ dB |
| | Average received SNR required to achieve the theoretical spectral efficiency of 576 bps/Hz | 4 dB | 20 dB | 11 dB | 12 dB | 12 dB | $>40$ dB |
| | Average per symbol computational complexity of $\mathrm{O}(\cdot)$ (flops) | $9.7\times10^4$ | $8.3\times10^4$ | $8.3\times10^4$ | $4.1\times10^8$ | $2.3\times10^9$ | $3.0\times10^7$ |
| Graphical 16-QAM signals | Average received SNR required to obtain the BER of $10^{-5}$ | 12 dB | $>40$ dB | 32 dB | 26 dB | 24 dB | – |
| | Average received SNR required to achieve the theoretical spectral efficiency of 1152 bps/Hz | 8 dB | 40 dB | 28 dB | 24 dB | 22 dB | |
| | Average per symbol computational complexity of $\mathrm{O}(\cdot)$ (flops) | $1.0\times10^5$ | $8.3\times10^4$ | $8.3\times10^4$ | $1.6\times10^9$ | $1.3\times10^{10}$ | |
| Graphical 64-QAM signals | Average received SNR required to obtain the BER of $10^{-5}$ | 14 dB | $>40$ dB | $>40$ dB | 39 dB | 34 dB | – |
| | Average received SNR required to achieve the theoretical spectral efficiency of 1728 bps/Hz | 12 dB | $>40$ dB | $>40$ dB | 38 dB | 32 dB | |
| | Average per symbol computational complexity of $\mathrm{O}(\cdot)$ (flops) | $1.0\times10^5$ | $8.3\times10^4$ | $8.3\times10^4$ | $3.7\times10^9$ | $7.5\times10^{10}$ | |

LLLSDR and the SDR detection algorithms grow with an increase in the modulation order $M$, while the complexity of the proposed CNNLAS algorithm remains unchanged at a lower order of magnitude of $9.3\times10^4$ with a slight increase to $1.0\times10^5$ when $N_c = 20$ increases to $N_c = 100$. For the graphical 64-QAM signals, the complexity of the proposed CNNLAS algorithm is on a similar order of magnitude to the complexities of $8.3\times10^4$ of the MMSE and MMSELAS algorithms, but much lower than the complexity of $4.8\times10^8$ of the DetNet detection algorithm, the complexity of $3.7\times10^9$ of the LLLSDR algorithm and the complexity of $7.5\times10^{10}$ of the SDR algorithm.

Similar to the MMSE and MMSELAS algorithms, the computational complexity of the proposed CNNLAS detection algorithm polynomially increases along with the number of antennas and remains invariable against the modulation order. Besides, especially for the graphical high-order modulation signals, the computational complexity of the proposed CNNLAS detection algorithm is much lower than those of the DetNet, LLLSDR and SDR algorithms in the uplink multiuser massive MIMO systems.

Eventually, as summarized in Tab. 3, the computational complexity and the performance including the BER and spectral efficiency of the above mentioned counterpart detection algorithms are compared with those of the proposed CNNLAS detection algorithm in the case of $N_f \times N_f = 11\times11$ and $N_c = 100$ for the graphical 64-QAM signals, in the case of $N_f \times N_f = 9\times9$ and $N_c = 100$ for the graphical 16-QAM signals, in the case of $N_f \times N_f = 3\times3$ and $N_c = 60$ for the graphical 4-QAM signals in the uplink multiuser massive MIMO system where $N_T = N_R = 288$. The DetNet detection algorithm has the lowest spectral efficiency and the worst BER performance with high computational complexity in this case. Although the computational complexity of the MMSE detection algorithm is the lowest, the BER and spectral efficiency performance of it are poor. With the same complexity, the BER and spectral efficiency performance of the MMSELAS algorithm are poor for the graphical high-order modulation signals, including 16-QAM and 64-QAM. The LLLSDR and SDR algorithms have improvements on the BER and spectral efficiency performance for the graphical high-order modulation signals, but their computational complexities are too high, and sharply increase versus $N_T$ and $M$. From the comparison, with a medium low polynomial computational complexity of $10^5$, the proposed CNNLAS detection algorithm required much lower average received SNRs to obtain the BER performance of $10^{-5}$ and to achieve the theoretical spectral efficiency both for the graphical low-order and high-order modulation signals in the uplink multiuser massive MIMO system.

## VI. CONCLUSION

In this paper, a CNNLAS detection algorithm is proposed on the basis of a graphical detection model for the uplink multiuser massive MIMO system. The proposed CNNLAS detection algorithm is both applicable for detecting the graphical low-order and high-order QAM signals in the uplink multiuser medium-small or massive MIMO systems. The performance of the proposed CNNLAS algorithm is evaluated in the cases of $N_T = N_R$ and $N_T < N_R$. The polynomial

average per symbol computational complexity of the proposed CNNLAS algorithm is $O(N_T^2 + N^2 + N_c^2)$. Compared with the counterpart algorithms, the proposed CNNLAS detection algorithm shows a stronger robustness against channel estimation errors, and it respectively obtains a BER of $10^{-5}$ for graphical 4-QAM, 16-QAM and 64-QAM signals at average received SNRs of 5 dB, 12 dB and 14 dB when $N_T = N_R = 288$. Meanwhile, it respectively requires much lower average received SNRs of 4 dB, 8 dB and 12 dB to reach the theoretical spectral efficiency.

## REFERENCES

[1] B. Zong, C. Fan, X. Wang, X. Duan, B. Wang, and J. Wang, "6G technologies: Key drivers, core requirements, system architectures, and enabling technologies," *IEEE Veh. Technol. Mag.*, vol. 14, no. 3, pp. 18–27, Sep. 2019.

[2] K. B. Letaief, W. Chen, Y. Shi, J. Zhang, and Y.-J.-A. Zhang, "The roadmap to 6G: AI empowered wireless networks," *IEEE Commun. Mag.*, vol. 57, no. 8, pp. 84–90, Aug. 2019.

[3] L. Lu, G. Y. Li, A. L. Swindlehurst, A. Ashikhmin, and R. Zhang, "An overview of massive MIMO: Benefits and challenges," *IEEE J. Sel. Topics Signal Process.*, vol. 8, no. 5, pp. 742–758, Oct. 2014.

[4] S. Yang and L. Hanzo, "Fifty years of MIMO detection: The road to large-scale MIMOs," *IEEE Commun. Surveys Tuts.*, vol. 17, no. 4, pp. 1941–1988, Sep. 2015.

[5] Y. Yapici, I. Guvenc, and Y. Kakishima, "A MAP-based layered detection algorithm and outage analysis over MIMO channels," *IEEE Trans. Wireless Commun.*, vol. 17, no. 7, pp. 4256–4269, Jul. 2018.

[6] K. Vishnu Vardhan, S. K. Mohammed, A. Chockalingam, and B. Sundar Rajan, "A low-complexity detector for large MIMO systems and multicarrier CDMA systems," *IEEE J. Sel. Areas Commun.*, vol. 26, no. 3, pp. 473–485, Apr. 2008.

[7] A. Elghariani and M. Zoltowski, "Low complexity detection algorithms in large-scale MIMO systems," *IEEE Trans. Wireless Commun.*, vol. 15, no. 3, pp. 1689–1702, Mar. 2016.

[8] A. K. Sah and A. K. Chaturvedi, "An unconstrained likelihood ascent based detection algorithm for large MIMO systems," *IEEE Trans. Wireless Commun.*, vol. 16, no. 4, pp. 2262–2273, Apr. 2017.

[9] J. C. Hedstrom, C. H. Yuen, R.-R. Chen, and B. Farhang-Boroujeny, "Achieving near MAP performance with an excited Markov chain Monte Carlo MIMO detector," *IEEE Trans. Wireless Commun.*, vol. 16, no. 12, pp. 7718–7732, Dec. 2017.

[10] J. Choi, "An MCM–MIMO detector as a stochastic linear system solver using successive overrelexation," *IEEE Trans. Wireless Commun.*, vol. 15, no. 2, pp. 1445–1455, Feb. 2016.

[11] S. Yang, X. Xu, D. Alanis, S. Xin Ng, and L. Hanzo, "Is the low-complexity mobile-relay-aided FFR-DAS capable of outperforming the high-complexity CoMP?" *IEEE Trans. Veh. Technol.*, vol. 65, no. 4, pp. 2154–2169, Apr. 2016.

[12] S. Yang, T. Lv, R. G. Maunder, and L. Hanzo, "From nominal to true a posteriori probabilities: An exact Bayesian theorem based probabilistic data association approach for iterative MIMO detection and decoding," *IEEE Trans. Commun.*, vol. 61, no. 7, pp. 2782–2793, Jul. 2013.

[13] P. Sen and A. O. Yilmaz, "A low-complexity graph-based LMMSE receiver for MIMO ISI channels with $M$ -QAM modulation," *IEEE Trans. Wireless Commun.*, vol. 16, no. 2, pp. 1185–1195, Feb. 2017.

[14] S. Yoon and C.-B. Chae, "Low-complexity MIMO detection based on belief propagation over pairwise graphs," *IEEE Trans. Veh. Technol.*, vol. 63, no. 5, pp. 2363–2377, Jun. 2014.

[15] M. Mandloi and V. Bhatia, "Multiple stage ant colony optimization algorithm for near-OPTD large-MIMO detection," in *Proc. 23rd Eur. Signal Process. Conf. (EUSIPCO)*, Nice, France, Aug. 2015, pp. 914–918.

[16] Z.-Q. Luo, W.-K. Ma, A. So, Y. Ye, and S. Zhang, "Semidefinite relaxation of quadratic optimization problems," *IEEE Signal Process. Mag.*, vol. 27, no. 3, pp. 20–34, May 2010.

[17] A. Mobasher, M. Taherzadeh, R. Sotirov, and A. K. Khandani, "A near-maximum-likelihood decoding algorithm for MIMO systems based on semi-definite programming," *IEEE Trans. Inf. Theory*, vol. 53, no. 11, pp. 3869–3886, Nov. 2007.

[18] T. Datta, N. A. Kumar, A. Chockalingam, and B. S. Rajan, "A novel Monte-Carlo-sampling-based receiver for large-scale uplink multiuser MIMO systems," *IEEE Trans. Veh. Technol.*, vol. 62, no. 7, pp. 3019–3038, Sep. 2013.

[19] P. Som, T. Datta, N. Srinidhi, A. Chockalingam, and B. S. Rajan, "Low-complexity detection in large-dimension MIMO-ISI channels using graphical models," *IEEE J. Sel. Topics Signal Process.*, vol. 5, no. 8, pp. 1497–1511, Dec. 2011.

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

[21] Y. Liao, H. Yao, Y. Hua, and C. Li, "CSI feedback based on deep learning for massive MIMO systems," *IEEE Access*, vol. 7, pp. 86810–86820, Jun. 2019.

[22] Y. Yang, Y. Li, K. Li, S. Zhao, R. Chen, J. Wang, and S. Ci, "DECCO: Deep-learning enabled coverage and capacity optimization for massive MIMO systems," *IEEE Access*, vol. 6, pp. 23361–23371, Apr. 2018.

[23] H. Ye, G. Y. Li, and B.-H. Juang, "Power of deep learning for channel estimation and signal detection in OFDM systems," *IEEE Wireless Commun. Lett.*, vol. 7, no. 1, pp. 114–117, Feb. 2018.

[24] E. Nachmani, E. Marciano, L. Lugosch, W. J. Gross, D. Burshtein, and Y. Be'ery, "Deep learning methods for improved decoding of linear codes," *IEEE J. Sel. Topics Signal Process.*, vol. 12, no. 1, pp. 119–131, Feb. 2018.

[25] M. Kim, N.-I. Kim, W. Lee, and D.-H. Cho, "Deep learning-aided SCMA," *IEEE Commun. Lett.*, vol. 22, no. 4, pp. 720–723, Apr. 2018.

[26] M. Mohammadkarimi, M. Mehrabi, M. Ardakani, and Y. Jing, "Deep learning-based sphere decoding," *IEEE Trans. Wireless Commun.*, vol. 18, no. 9, pp. 4368–4378, Sep. 2019.

[27] N. Samuel, T. Diskin, and A. Wiesel, "Deep MIMO detection," in *Proc. IEEE 18th Int. Workshop Signal Process. Adv. Wireless Commun. (SPAWC)*, Sapporo, Jul. 2017, pp. 1–5.

[28] N. Samuel, T. Diskin, and A. Wiesel, "Learning to detect," *IEEE Trans. Signal Process.*, vol. 67, no. 10, pp. 2554–2564, May 2019.

[29] H. He, C.-K. Wen, S. Jin, and G. Y. Li, "A model-driven deep learning network for MIMO detection," in *Proc. IEEE Global Conf. Signal Inf. Process. (GlobalSIP)*, Anaheim, CA, USA, Nov. 2018, pp. 584–588.

[30] V. Corlay, J. J. Boutros, P. Ciblat, and L. Brunel, "Multilevel MIMO detection with deep learning," in *Proc. 52nd Asilomar Conf. Signals, Syst., Comput.*, Pacific Grove, CA, USA, Oct. 2018, pp. 1805–1809.

[31] H. Wu, Y. Liu, and K. Wang, "Analysis of DFT-based channel estimation for uplink massive MIMO systems," *IEEE Commun. Lett.*, vol. 22, no. 2, pp. 328–331, Feb. 2018.

[32] G. H. Golub and C. F. Van Loan, *Matrix Computations*, vol. 3. Baltimore, MD, USA: The Johns Hopkins Univ. Press, 2012.

[33] K. He and J. Sun, "Convolutional neural networks at constrained time cost," 2014, *arXiv:1412.1710*. [Online]. Available: http://arxiv.org/abs/1412.1710

[34] A. Krizhevsky, I. Sutskever, and G. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, Jan. 2012, vol. 25, no. 2, pp. 1097–1105.

[35] H. Artes, D. Seethaler, and F. Hlawatsch, "Efficient detection algorithms for MIMO channels: A geometrical approach to approximate ml detection," *IEEE Trans. Signal Process.*, vol. 51, no. 11, pp. 2808–2820, Nov. 2003.

[36] R. Y. Mesleh, H. Haas, S. Sinanovic, C. W. Ahn, and S. Yun, "Spatial modulation," *IEEE Trans. Veh. Technol.*, vol. 57, no. 4, pp. 2228–2241, Jul. 2008.

**LIN LI** (Member, IEEE) received the B.S., M.S., and Ph.D. degrees in information and communication engineering from the Harbin Institute of Technology, Harbin, China, in 2010, 2012, and 2017, respectively. She is currently a Lecturer with the Jiangsu Key Laboratory of Broadband Wireless Communication and Internet of Things and the School of Internet of Things, Nanjing University of Posts and Telecommunications. Her main research interests include wireless communications, massive MIMO systems, signal processing, and machine learning. She was a recipient of the ChinaCom 2016 Best Paper Award and the IEEE ICSIDP 2019 Excellent Paper Award.

**HUIJUN HOU** (Member, IEEE) received the B.S., M.S., and Ph.D. degrees in information and communication engineering from the Harbin Institute of Technology, Harbin, China, in 2010, 2012, and 2017, respectively. He is currently an Engineer with the CETC 14th Research Institute. His research interests include statistical and array signal processing and machine learning.

**WEIXIAO MENG** (Senior Member, IEEE) received the B.Eng., M.Eng., and Ph.D. degrees from the Harbin Institute of Technology (HIT), Harbin, China, in 1990, 1995, and 2000, respectively. From 1998 to 1999, he worked at NTT DoCoMo on adaptive array antenna and dynamic resource allocation for beyond 3G as a Senior Visiting Researcher. He is currently a Full Professor and the Vice Dean of the School of Electronics and Information Engineering, HIT. He has published three books and over 220 articles on journals and international conferences. His research interests include broadband wireless communications and networking, MIMO, GNSS receiver, and wireless localization technologies. He is a Senior Member of the IEEE ComSoc. He is also the Chair of the IEEE Communications Society Harbin Chapter, the China Institute of Electronics, and the China Institute of Communication. He has been an Editorial Board Member for Wiley's WCMC Journal, since 2010, an Area Editor of PHYCOM Journal, since 2014, an Editorial Board of the IEEE COMMUNICATIONS SURVEYS, AND TUTORIALS, since 2014, and the IEEE WIRELESS COMMUNICATIONS, since 2015. He acted as leading TPC Co-Chair of ChinaCom2011 and ChinaCom2016, a leading Services and Applications Track Co-Chair of the IEEE WCNC2013, the Awards Co-Chair of the IEEE ICC2015, and the Wireless Networking Symposia Co-Chair of Globecom 2015. In 2005, he was honored provincial excellent returnee and selected into New Century Excellent Talents (NCET) plan by Ministry of Education, China, in 2008, and the Distinguished Academic Leadership of Harbin.