PAPER

Cascaded Deep Neural Network for Off-Grid Direction-of-Arrival Estimation*

Huafei WANG[†a)], Xianpeng WANG[†b)], Xiang LAN[†c)], *and* Ting SU[†d)], *Nonmembers*

SUMMARY Using deep learning (DL) to achieve direction-of-arrival (DOA) estimation is an open and meaningful exploration. Existing DL-based methods achieve DOA estimation by spectrum regression or multi-label classification task. While, both of them face the problem of off-grid errors. In this paper, we proposed a cascaded deep neural network (DNN) framework named as off-grid network (OGNet) to provide accurate DOA estimation in the case of off-grid. The OGNet is composed of an autoencoder consisted by fully connected (FC) layers and a deep convolutional neural network (CNN) with 2-dimensional convolutional layers. In the proposed OGNet, the off-grid error is modeled into labels to achieve off-grid DOA estimation based on its sparsity. As compared to the state-of-the-art grid-based methods, the OGNet shows advantages in terms of precision and resolution. The effectiveness and superiority of the OGNet are demonstrated by extensive simulation experiments in different experimental conditions.

key words: *off-grid, direction-of-arrival estimation, deep learning, autoencoder, convolutional neural networks*

1. Introduction

Direction-of-arrival (DOA) estimation has been a hot topic of research for decades, since it plays a crucial role in the field of wireless communication and target sensing [1], [2]. In order to realize accurate DOA estimation, plenty of methods have been proposed over the past few decades. The most well-known methods are the subspace-based methods, which include the multiple signal classification (MUSIC) [3], estimation of signal parameters via rotational invariance techniques (ESPRIT) [4] and their variants [5]–[9]. The basic principle of MUSIC method [3] is that a spatial spectrum is first constructed based on the orthogonal relationship between the signal and noise subspaces, and then the peak search is performed over the spatial spectrum with a specific step size to achieve DOA estimation. While, the ESPRIT method [4], [9] capitalizes the rotational invariance of signal subspace and does not require spectrum search. However, the performance of these subspace-based methods depend on the accuracy of covariance, which depends on the number of snapshots and signal-to-noise ratio (SNR), therefore their performance may suffer significantly degradation at insufficient number of snapshots or low SNRs.

In recent decade, compressed sensing (CS) technique has attracted much attentions [10], [11], which has been successfully applied in DOA estimation [12], [13]. The CS-based methods mainly capture the sparsity of sources in spatial domain, and adopt different strategies of sparse minimization to achieve DOA estimation. Since the sparsity of source signals comes from the discrete grid in spatial domain, the CS-based methods can be classified into three main categories: on-grid, off-grid, and gridless methods [14]. On-grid methods [15], [16] can accurately estimate the angles coincided with the fixed grid points in spatial domain. However, their performance suffers from the off-grid error when angles mismatch with the grid points, especially under a coarse grid condition. Apparently, the problem of off-grid errors faced by the on-grid methods can be alleviated by recursive grid refinement or increasing the degree of spatial discretization. However, it may result in a significant increase in computational complexity. Off-grid methods [17], [18] and gridless methods [19] achieve a balance between estimation accuracy and computational complexity, which can realize high precision DOA estimation under coarse grid conditions with low computational cost. However, since all above CS-based methods are model-driven, they rely on the pre-established mathematical model, and share a common shortcoming that each estimation needs a complete optimization process that need appropriate parameter initialization. The inappropriate initial parameters may cause the performance degradation and even method failure.

Most recently, DOA estimation using deep learning (DL) technique [20], [21] has raised much attentions, which is completely data-driven method. The DL-based methods use the powerful nonlinear mapping capability of neural networks (NN) to learn features in the array data to achieve DOA estimation. As compare to the CS-based approaches, DL-based methods can achieve DOA estimation based on simple multiplications and additions in trained networks with no optimization process, and the training is to be done off-line once and for all. As a current research hotspot, a stream of DL-based methods have emerged for DOA estimation and its application [22]–[24]. Specifically, in [25] and

Manuscript received January 9, 2024.
Manuscript revised March 24, 2024.
Manuscript publicized May 14, 2024.

[†] School of Information and Communication Engineering and the State Key Laboratory of Marine Resource Utilization in South China Sea, Hainan University, 570228 China.

[*] This work is supported in part by the Hainan Province Science and Technology Special Fund under Grant ZDYF2023GXJS168; in part by the National University of Defense Technology Independent Innovation Fund under Grant 23-ZZCX-JDZ-45; in part by the Important Science and Technology of Haikou City under Grant 2021-004; and in part by the National Natural Science Foundation of China under Grant 62101088.

a) E-mail: wong9525@163.com
b) E-mail: wxpeng1986@126.com (Corresponding author)
c) E-mail: xlan@hainanu.edu.cn
d) E-mail: suting4190@hainanu.edu.cn

DOI: 10.23919/transcom.2024EBP3006

Table 1 Glossary of notations throughout the paper.

Notations	Definitions
Italic (e.g., x or X):	Scalars
Lowercase bold italic (e.g., $\boldsymbol{x}$):	Vectors
Capital bold italic (e.g., $\boldsymbol{X}$):	Matrices
$\boldsymbol{I}_i$:	$(i \times i)$-dimensional unit matrix
$\mathbf{0}_i$:	$(i \times i)$-dimensional zero matrix
$\boldsymbol{X}^{-1}$:	Inverse of $\boldsymbol{X}$
$\boldsymbol{X}^{T}$:	Transpose of $\boldsymbol{X}$
$\boldsymbol{X}^{H}$:	Hermitian transpose of $\boldsymbol{X}$
$\lVert \boldsymbol{X} \rVert_F$:	Frobenius-norm of $\boldsymbol{X}$
$\lvert \boldsymbol{x} \rvert_2$:	2-norm of vector $\boldsymbol{x}$
$\lvert x \rvert$:	Modulus of scalar x
$\mathbb{E}\{\cdot\}$:	Calculate expectation
$\mathbb{R}^{i\times i}$:	Set of $(i \times i)$-dimensional real value matrix
$\mathbb{C}^{i\times i}$:	Set of $(i \times i)$-dimensional complex value matrix
C_N^n:	Generate all possible combinations of n numbers from N numbers
$utv\{\boldsymbol{X}\}$:	Vectorize the upper triangular part of $\boldsymbol{X}$ by column
$mat\{\boldsymbol{x}\}$:	The reverse operator of $utv\{\boldsymbol{X}\}$

[26], the authors applied the denoising autoencoder to denoise the array covariance of uniform linear array and sparse linear array, respectively. Then implement DOA estimation based on the denoised covariance using MUSIC-based methods, e.g., root MUSIC and spatial smooth MUSIC. Similarly, A. Barthelme et al. used neural network to reconstruct the covariance matrix from sample covariance matrix, then achieve DOA estimation by applying the MUSIC estimator to the reconstructed covariance matrix [27]. On the other hand, a deep convolutional neural network (CNN) was presented in [28] to reconstruct the noiseless covariance matrix by using its Toeplitz structure, then root MUSIC method is applied to realized gridless DOA estimation. However, the methods in [25]–[28] are essentially semi-DL methods with the techniques for final DOA estimation still model-driven, which results in their inability to achieve end-to-end DOA estimation. For purely DL-based methods, the authors in [29], [30] proposed the deep neural networks (DNN) for robust DOA estimation in non-ideal situations such as array imperfections and color noise. However, multiple parallel networks are adopted both in [29] and [30], which leads to large network structures that require large amounts of accurately labeled data for training, and such volume of labeled data for non-ideal situations are very difficult to collect in practice. On the other hand, a deep CNN was developed in [31] with utilizing the sparsity prior. However, since 1-dimensional (1D) convolution is utilized, the method do not exhibit significant performance improvements. Further, G. K. Papageorgiou et al. designed a deep network in [32] for DOA estimation in low SNR, where the 2-dimensional (2D) convolutions are used and the DOA estimation is modeled as a multi-label classification task by inputting 3-channel covariance. Nevertheless, such a network suffers from the similar problem that CS-based methods suffer, i.e., the off-grid errors. Coarse labeling of covariance leads to that the network can only accurately classify the angles been labeled, the off-labeled DOAs (similar to the off-grid DOAs) can only be classified as the nearest angle to it, which cause off-grid errors. While, the dense labeling will undoubtedly require a large amount of labeled data, the collection of such amounts of labeled data is a challenge in practical applications.

Above all, most of the existing DL-based DOA estimation methods either do not provide end-to-end DOA estimation or the estimation precision is restricted by off-grid errors. Thus, this paper try to fill in this gap by proposing a cascaded DNN. The proposed neural network is referred to as off-grid network (OGNet) which is composed of an autoencoder (AE) and a deep CNN (DCNN). The AE behaves like a filter, which takes the upper triangular part of the sampling unitary covariance as input to reduce the divergences between sampling and theoretical unitary covariance. Afterward, the reconstructed unitary covariance by AE is used as the input of the DCNN to predict the off-grid error vector hence to realize DOA estimation based on its sparsity. The major contributions of this paper are summarized as:

- A neural network architecture is proposed for the DOA estimation in off-grid scenarios. The proposed architecture include an AE and a deep CNN.
- In the proposed neural network, the AE behaves like a pre-processor to reduce divergences between the sampling and theoretical unitary covariance. And the deep CNN is designed for off-grid DOA estimation by modeling the off-grid error into labels, which enables it can achieve off-grid DOA estimation based on its sparsity and without a priori information of on-grid angles.
- The proposed neural network architecture achieves more accurate off-grid DOA estimation as compare to the state-of-the-art grid-based methods include traditional methods and DL-based methods.

The remaining part of this paper is structured as follows: A briefly description of the problem formulation of DOA estimation is given in Sect. 2. In Sect. 3, the architecture of the proposed cascaded DNN for off-grid DOA estimation is presented. The network training strategies corresponding to AE and DCNN are introduced in Sect. 4. Simulation experiment results are shown in Sect. 5 to evaluate the effectiveness and superiority of the proposed network. Finally, the conclusions

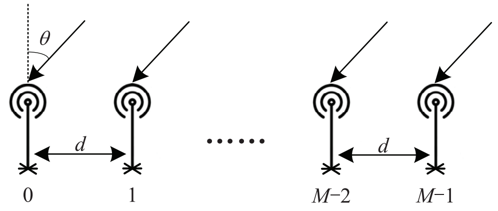


Fig. 1 Uniform linear array with M antennas and inter-antenna distance $d = \lambda/2$.

of this paper are given in Sect. 6. The glossary of notations throughout this paper is given in Table 1 for convince.

2. Problem Formulation

For DOA estimation, there are several array geometries to be applied, such as the linear, circular and planar. In this work, a uniform linear array (ULA) is considered. As shown in Fig. 1, suppose a ULA equipped with M antennas is configured at a inter-antenna distance of $d = \lambda/2$, where λ is the wavelength of signals. With P independent far-field narrow-band signals impinging on the ULA from different directions of $\theta_p (p = 1, 2, \ldots, P)$, the data received by ULA at l-th sampling snapshot is [3]

$$\boldsymbol{y}(l) = \boldsymbol{A}\boldsymbol{s}(l) + \boldsymbol{n}(l),\ l = 1, 2, \cdots, L, \tag{1}$$

where $\boldsymbol{s}(l) = [s_1(l), \cdots, s_P(l)]^T \in \mathbb{C}^{P\times 1}$ denotes the signal vector and $\boldsymbol{n}(l)$ denotes the additive Gaussian white noise vector at l-th sampling snapshot. $\boldsymbol{A} = [\boldsymbol{a}(\theta_1), \boldsymbol{a}(\theta_2), \cdots, \boldsymbol{a}(\theta_P)]$ denotes the $M \times P$ array steering matrix with

$$\boldsymbol{a}(\theta_p) = [1, e^{j(2\pi d/\lambda)\sin\theta_p}, \cdots, e^{j(2\pi d/\lambda)(M-1)\sin\theta_p}]^T, \tag{2}$$

where j is the imaginary unit. By collecting L snapshots, the multi-snapshot data is expressed as

$$\boldsymbol{Y} = \boldsymbol{A}\boldsymbol{S} + \boldsymbol{N}, \tag{3}$$

with $\boldsymbol{Y} = [\boldsymbol{y}(1), \cdots, \boldsymbol{y}(L)]$, $\boldsymbol{S} = [\boldsymbol{s}(1), \cdots, \boldsymbol{s}(L)]$ and $\boldsymbol{N} = [\boldsymbol{n}(1), \cdots, \boldsymbol{n}(L)]$.

Based on Eq. (1), with collecting infinite snapshots, the theoretical covariance of the receiving data can be expressed as

$$\boldsymbol{R} = \mathbb{E}\{\boldsymbol{y}(l)\boldsymbol{y}(l)^H\} = \boldsymbol{A}\boldsymbol{R}_s\boldsymbol{A}^H + \boldsymbol{R}_n, \tag{4}$$

where $\boldsymbol{R}_s$ and $\boldsymbol{R}_n$ denote the covariance of incident signals and noise, respectively. However, the theoretical covariance in Eq. (4) is hard to obtain and unknown in practice, hence it is usually replaced by the sampling covariance, which is

$$\bar{\boldsymbol{R}} = \frac{1}{L}\sum_{l=1}^{L}\boldsymbol{y}(l)\boldsymbol{y}(l)^H. \tag{5}$$

Since the theoretical covariance $\boldsymbol{R}$ is a centro-Hermitian matrix, it can be transformed into a real-valued matrix [33], [34], which is called theoretical unitary covariance (TUC), by [8]

$$\boldsymbol{R}_u = (\boldsymbol{U}_M)^H\boldsymbol{R}\boldsymbol{U}_M \tag{6}$$

with

$$\begin{cases} \boldsymbol{U}_{M=even} = \frac{\sqrt{2}}{2}\begin{bmatrix} \boldsymbol{I}_{\frac{M}{2}} & j\boldsymbol{I}_{\frac{M}{2}} \\ \boldsymbol{\Pi}_{\frac{M}{2}} & -j\boldsymbol{\Pi}_{\frac{M}{2}} \end{bmatrix}, \\ \boldsymbol{U}_{M=odd} = \frac{\sqrt{2}}{2}\begin{bmatrix} \boldsymbol{I}_{\frac{M-1}{2}} & \boldsymbol{0}_{\frac{M-1}{2}\times 1} & j\boldsymbol{I}_{\frac{M-1}{2}} \\ \boldsymbol{0}_{1\times\frac{M-1}{2}} & \sqrt{2} & \boldsymbol{0}_{1\times\frac{M-1}{2}} \\ \boldsymbol{\Pi}_{\frac{M-1}{2}} & \boldsymbol{0}_{\frac{M-1}{2}\times 1} & -j\boldsymbol{\Pi}_{\frac{M-1}{2}} \end{bmatrix}, \end{cases} \tag{7}$$

where $\boldsymbol{\Pi}_i$ denotes $(i \times i)$-dimensional matrix with the anti-diagonal elements being 1 others all 0. While, although the sampling covariance $\bar{\boldsymbol{R}}$ in Eq. (5) is a Hermitian matrix, it's not centro-Hermitian, hence it cannot directly be transformed into real-valued by Eq. (6). Fortunately, the forward-backward technique can be applied first to turn $\bar{\boldsymbol{R}}$ into a centro-Hermitian matrix, which is

$$\bar{\boldsymbol{R}}_{fb} = \frac{1}{2}(\bar{\boldsymbol{R}} + \boldsymbol{\Pi}_M\bar{\boldsymbol{R}}^*\boldsymbol{\Pi}_M). \tag{8}$$

Then, the sampling unitary covariance (SUC) based on $\bar{\boldsymbol{R}}$ is

$$\bar{\boldsymbol{R}}_u = (\boldsymbol{U}_M)^H\bar{\boldsymbol{R}}_{fb}\boldsymbol{U}_M. \tag{9}$$

Based on Eq. (9), we are interested in estimating unknown DOAs from $\bar{\boldsymbol{R}}_u$. Hence, as considering the powerful nonlinear mapping capability of NN, a cascaded DNN is designed in the following section to predict the off-grid error vector by using $\bar{\boldsymbol{R}}_u$ as input, then achieve off-grid DOA estimation using the sparsity of the predicted off-grid error vector.

3. Proposed Cascaded DNN for Off-Grid DOA Estimation

The overall architecture of the proposed cascaded DNN, named as OGNet, is shown in Fig. 2, which is composed of two components. The first component is a neural network called AE, which is consisted by fully connected (FC) layers. And the second component is the DCNN mainly composed of 2-dimensional (2D) convolutional (Conv.) layers and FC layers. The former is to reduce the divergences between $\bar{\boldsymbol{R}}_u$ and $\boldsymbol{R}_u$, while the latter is to predict the off-grid error vector by using the unitary covariance predicted by AE as input. Finally, the DOA estimation is realized based on the sparsity of the predicted off-grid error vector. The detailed architecture of the components within OGNet is introduced as following.

3.1 The Architecture of AE

From Sect. 2, it is known that the $\boldsymbol{R}_u$ is obtained based on

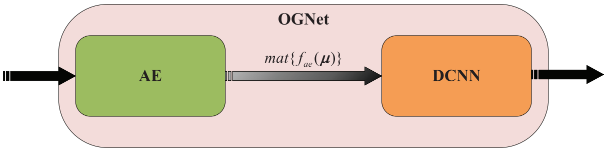


Fig. 2 The overall architecture of the proposed cascaded deep neural network, i.e., the OGNet.

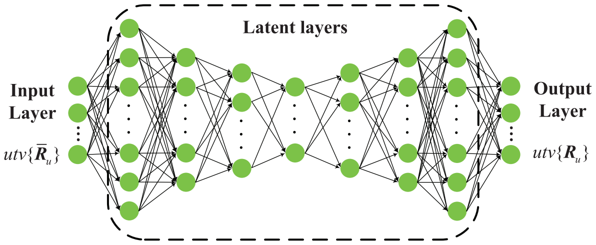


Fig. 3 The architecture of autoencoder within OGNet.

infinite snapshots, which is practically unrealistic, while the $\bar{\boldsymbol{R}}_u$ is calculated with L snapshots. Therefore, there must exist certain difference between $\bar{\boldsymbol{R}}_u$ and $\boldsymbol{R}_u$, i.e.

$$\boldsymbol{R}_u = \bar{\boldsymbol{R}}_u + \Delta\boldsymbol{R}_u, \tag{10}$$

where $\Delta\boldsymbol{R}_u$ represents the divergence matrix between $\bar{\boldsymbol{R}}_u$ and $\boldsymbol{R}_u$. The AE within the proposed OGNet is designed to reduce $\Delta\boldsymbol{R}_u$. The architecture of AE within OGNet is displayed in Fig. 3, which is consisted of 9 FC layers include 1 input layer, 1 output layer and 7 latent layers (i.e., FC layers). Each latent layer is followed by a ReLU activation layer except for the input and output layers to avoid the gradient disappearing. The specific configurations of each layer in AE is given in Table 2.

Note that $\bar{\boldsymbol{R}}_u$ and $\boldsymbol{R}_u$ are all $M \times M$-dimensional Hermitian matrices, therefore the training data pair for AE is respectively denoted as vector consisted of the elements of upper triangular parts of $\bar{\boldsymbol{R}}_u$ and $\boldsymbol{R}_u$ by columns, i.e.,

$$\begin{cases} \boldsymbol{\mu} = utv\{\bar{\boldsymbol{R}}_u\}, \\ \boldsymbol{u} = utv\{\boldsymbol{R}_u\}. \end{cases} \tag{11}$$

where $\boldsymbol{\mu} \in \mathbb{R}^{(\frac{M(M-1)}{2}+M)\times 1}$ represents the input of AE, and $\boldsymbol{u} \in \mathbb{R}^{(\frac{M(M-1)}{2}+M)\times 1}$ represents the output label of AE. Then, the nonlinear mapping procedure of AE can be parameterized as

$$f_{ae}(\boldsymbol{\mu}) = f_{aout}(f_{a7}(\ldots(f_{a1}(f_{ain}(\boldsymbol{\mu}))))) = \boldsymbol{\omega}, \tag{12}$$

where $\boldsymbol{\omega} \in \mathbb{R}^{(\frac{M(M-1)}{2}+M)\times 1}$ is the predicted output of AE during training; $f_{ae}\{\cdot\}$ represents the nonlinear mapping function of the whole AE, $f_{ain}\{\cdot\}$ and $f_{aout}\{\cdot\}$ respectively denote the mapping function of input layer and output layer, and $f_{ai}\{\cdot\}$ with $i = 1, 2, \cdots, 7$ denotes the mapping function of i-th latent layer.

Since the AE is modeled for a regression task and completely composed of FC layers, it is potential to over-fitting in the case of small training data. In order to prevent overfitting, the mean-square-error (MSE) with L_2 regularization is chosen as the loss function of AE to optimize the trainable weights and biases set $\boldsymbol{\Theta}_a$ during training phase, that is

Table 2 Specific configurations of autoencoder.

Layers	Number of neurons	Activation function
Input	$M(M-1)/2 + M$	-
FC #1	300	ReLU
FC #2	200	ReLU
FC #3	100	ReLU
FC #4	50	ReLU
FC #5	100	ReLU
FC #6	200	ReLU
FC #7	300	ReLU
Output	$M(M-1)/2 + M$	Linear

$$\boldsymbol{\Theta}_a^\star = \arg\min_{\boldsymbol{\Theta}_a} \frac{1}{D_a}\left\{\sum_{d=1}^{D_a} \mathcal{L}(\boldsymbol{\omega}^{(d)}, \boldsymbol{u}^{(d)}) + \frac{\lambda}{2}\sum_{n=1}^{N_a} \|\boldsymbol{\mathcal{W}}_a^{(n)}\|_F^2\right\}, \tag{13}$$

where $\mathcal{L}(\boldsymbol{\omega}^{(d)}, \boldsymbol{u}^{(d)}) = \{\frac{1}{Q}\sum_{i=1}^{Q} |\boldsymbol{\omega}_i^{(d)} - \boldsymbol{u}_i^{(d)}|^2\}$ is the MSE loss with $Q = \frac{M(M-1)}{2} + M$; D_a represents the total number of training data for AE; $\boldsymbol{\omega}^{(d)}$ and $\boldsymbol{u}^{(d)}$ respectively denote the predicted output and the output label of AE when inputting d-th data; $\boldsymbol{\omega}_i^{(d)}$ and $\boldsymbol{u}_i^{(d)}$ represent the i-th entries of $\boldsymbol{\omega}^{(d)}$ and $\boldsymbol{u}^{(d)}$, respectively; $\boldsymbol{\Theta}_a = \{\boldsymbol{\mathcal{W}}_a, \boldsymbol{b}_a\}$ is the set of trainable weights and biases in AE; λ is the regularization parameter for the weights in AE, which is $\lambda = 10^{-4}$ in this paper; N_a is the total number of hidden layers in AE and $\boldsymbol{\mathcal{W}}_a^{(n)}$ represents the weights of n-th hidden layer of AE.

3.2 The Architecture of DCNN

After obtaining a prediction of $\boldsymbol{u}$ from AE, the predicted unitary covariance can be obtained based on its Hermitian property, i.e.,

$$\hat{\boldsymbol{R}}_u = mat\{\boldsymbol{\omega}\}, \tag{14}$$

which is taken as the input of DCNN to predict the sparse off-grid error vector. During training phase, $\boldsymbol{R}_u$ is chosen as the training input. The architecture of the DCNN is displayed in Fig. 4. The DCNN contains 1 input layer, 1 output layer, 10 2D Conv. layers, 1 flatten layer and 4 FC layers. Each Conv. layer has 64 channels and is followed by a batch normalization (BN) layer [35]. The kernel size of all Conv. layers is $\kappa \times \kappa$ with $\kappa = 3$ and stride $\delta = 1$ and same padding. Similarly, to prevent the gradient disappearing, the activation function used in each Conv. and FC layer of DCNN is ReLU. The specific configurations of each layer of DCNN is given in Table 3.

The output label of DCNN is designed as an G-dimensional P-sparse vector $\boldsymbol{\xi}$, where G depends on the number of discrete grid points in spatial domain. For instance, if the spatial domain from $-\phi$ to ϕ is discretized by the grid interval of α, then $G = 2\phi/\alpha + 1$, and the spatial discrete grid can be obtained as $\boldsymbol{\Psi} = \{\psi_1, \psi_2, \cdots, \psi_G\}$ with $\alpha = \psi_{g+1} - \psi_g$ $(g = 1, 2, \cdots, G-1)$. Suppose that the true

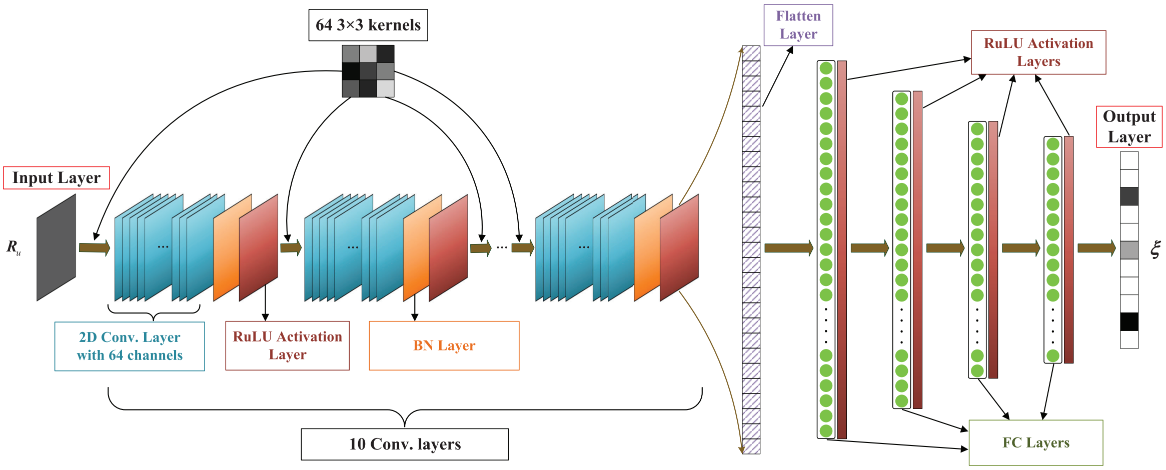


Fig. 4 The architecture of DCNN within OGNet.

Table 3 Specific configurations of DCNN.

Layers	Size/neurons	Activation function
Input	$M \times M \times 1$	-
Conv. #1 with BN	$64 \times M \times M \times 1$	ReLU
Conv. #2 with BN	$64 \times M \times M \times 1$	ReLU
⋮	⋮	⋮
Conv. #9 with BN	$64 \times M \times M \times 1$	ReLU
Conv. #10 with BN	$64 \times M \times M \times 1$	ReLU
Flatten	9216	-
FC #1	2048	ReLU
FC #2	1024	ReLU
FC #3	512	ReLU
FC #4	256	ReLU
Output	G	Linear

DOAs of P sources are $\boldsymbol{\theta} = \{\theta_1, \theta_2, \cdots, \theta_P\}$, then the sparse off-grid error vector is expressed as

$$\boldsymbol{e} = [e_1, e_2, \cdots, e_G]^T \in \mathbb{R}^{G\times 1}, \tag{15}$$

where $e_g \subset (-\alpha/2, \alpha/2]$ with $g = 1, 2, \cdots, G$, and the entries of $\boldsymbol{e}$ are all 0 except for the g_p-th entry being $e_{g_p} = \theta_p - \psi_{g_p}$ with $p = 1, 2, \cdots, P$ and $g_p = 1, 2, \cdots, G$. ψ_{g_p} denotes the angle in $\boldsymbol{\Psi}$ nearest to θ_p. Note that e_{g_p} could be negative or positive and could be very small when the targets are very close to the discrete grids. In order to enhance the sparsity of $\boldsymbol{e}$ and convert it into a positive vector for better learning by neural networks, a linear transformation is introduced as

$$\xi_{g_p} = e_{g_p} \times c + \frac{\alpha}{2} \times c, \tag{16}$$

where $g_p = 1, 2, \cdots, G$, c is a constant which is set to $c = 10$ in this paper. Then the output label of DCNN $\boldsymbol{\xi}$ is expressed as

$$\boldsymbol{\xi} = [\xi_1, \xi_2, \cdots, \xi_G]^T \in \mathbb{R}^{G\times 1}, \tag{17}$$

which shares the same sparsity with $\boldsymbol{e}$.

Remark 1. *We set $e_g \subset (-\alpha/2, \alpha/2]$ intentionally, because if $-\alpha/2$ and $\alpha/2$ both are included, there will exist conflict in the network labels, leading to problems during training. Let us take a simple example: suppose the true DOA is $-59°$ and $\alpha = 2°$, then we can labeled the outputs as that $e_1 = \alpha/2 = 1°$ or $e_2 = -\alpha/2 = -1°$, which are actually pointing at the same DOA. Hence, if $-\alpha/2$ and $\alpha/2$ are both included, the labels are conflict for training. On the other hand, we set $e_g \subset (-\alpha/2, \alpha/2]$ instead of $e_g \subset [-\alpha/2, \alpha/2)$. This is because a linear transformation is made in Eq. (16) to maintain the sparsity of the labels, if we set $e_g \subset [-\alpha/2, \alpha/2)$, the sparsity will be vanished when $e_g = -\alpha/2$, which will also lead to problems during training.*

Similarly, the nonlinear mapping procedure of DCNN can be parameterized as

$$f_{cnn}(\boldsymbol{R}_u) = f_{cout}(f_{c14}(f_{c13}(\ldots(f_{c2}(f_{c1}(f_{cin}(\boldsymbol{R}_u))))))) = \boldsymbol{\zeta}, \tag{18}$$

where $\boldsymbol{\zeta} \in \mathbb{R}^{G\times 1}$ represents the predicted output of DCNN during training; $f_{cnn}\{\cdot\}$ denotes the nonlinear mapping function of the entire DCNN, $f_{cin}\{\cdot\}$ and $f_{cout}\{\cdot\}$ respectively stand for the mapping function of input layer and output layer of DCNN; $f_{ci}\{\cdot\}$ with $i = 1, 2, \ldots, 10$ is the mapping function of i-th Conv. layer, and $f_{ci}\{\cdot\}$ with $i = 11, 12, \ldots, 14$ is the mapping function of $(i-10)$-th FC layer.

Likewise, the DCNN is modeled to complete a regression task. Therefore, MSE is chosen as the loss function of the DCNN for the optimization of the trainable weights and biases set $\boldsymbol{\Theta}_c$ in DCNN, i.e.,

$$\boldsymbol{\Theta}_c^\star = \arg\min_{\boldsymbol{\Theta}_c} \frac{1}{D_c}\left\{\sum_{d=1}^{D_c}\left\{\frac{1}{G}\sum_{i=1}^{G}|\zeta_i^{(d)} - \xi_i^{(d)}|^2\right\}\right\}, \tag{19}$$

where D_c represents the total number of training data for DCNN; $\boldsymbol{\zeta}^{(d)}$ and $\boldsymbol{\xi}^{(d)}$ respectively denote the predicted output and the training label of DCNN by input d-th data during training; $\zeta_i^{(d)}$ and $\xi_i^{(d)}$ represent the i-th entry of $\boldsymbol{\zeta}^{(d)}$ and $\boldsymbol{\xi}^{(d)}$, respectively. It should be noted that there are other candidate loss functions for regression task, such as mean-absolute-error (MAE) and smooth MAE. As compare to MAE, the gradient of MSE is dynamic (as the error decreases, so does the gradient), which can accelerate the convergence of the

Table 4 The procedure of off-grid DOA estimation using OGNet.

```
Algorithm 1: Off-grid DOA estimation using OGNet.
Train AE with proper training data and strategies (off-line);
Train DCNN with proper training data and strategies (off-line);
ULA receives data $\boldsymbol{Y}$;
Calculate the sampling covariance $\bar{\boldsymbol{R}}$;
Transform $\bar{\boldsymbol{R}}$ to unitary covariance $\bar{\boldsymbol{R}}_u$ by Eq. (8) and (9);
Get the upper triangular parts of $\bar{\boldsymbol{R}}_u$ by $\boldsymbol{\mu} = utv\{\bar{\boldsymbol{R}}_u\}$;
Input $\boldsymbol{\mu}$ into the trained AE to predict $\boldsymbol{\omega}$;
Obtain $\hat{\boldsymbol{R}}_u$ by Eq. (14);
Input $\hat{\boldsymbol{R}}_u$ into the trained DCNN to predict $\boldsymbol{\zeta}$;
Obtain $P$ spikes in $\boldsymbol{\zeta}$ and corresponding indices by peak searching;
Realize high-precision off-grid DOA estimation according to Eq. (20).
```

function and make the network training faster. Hence, the MSE rather than MAE is chosen as the loss function of DCNN, because the training of CNN values training speed, especially the training of deep CNN.

3.3 Off-Grid DOA Estimation

The training of the networks within OGNet are performed off-line with proper strategies to obtain the trained OGNet. Once the training is completed, the off-grid errors corresponding to sources can be predicted by feed a sampling covariance into the trained OGNet, while the exact DOAs are not estimated yet. The off-grid DOA estimation is realized by a post processing based on the sparsity of $\boldsymbol{\zeta}$, and without a priori on-grid angles estimation. Since $\boldsymbol{\zeta}$ is P-sparse, and the index of its each value corresponds to that of on-grid angle on the spatial grid $\boldsymbol{\Psi} = \{\psi_1, \psi_2, \cdots, \psi_G\}$, where there are spikes there're sources impinging from those angles. On the other hand, according to Eq. (16), each value of the spikes in $\boldsymbol{\zeta}$ contains the off-grid error information of sources. Hence, the on-grid angles and off-grid errors of sources can be obtained simultaneously by performing peak searching on $\boldsymbol{\zeta}$ to find P spikes. Then, the off-grid DOA estimation can be realized by

$$\bar{\boldsymbol{\theta}} = \boldsymbol{\Psi}_\iota + \left(\frac{\zeta_\iota}{c} - \frac{\alpha}{2}\right), \tag{20}$$

where $\iota \in 1, 2, \cdots, G$ denotes the indices corresponding to the P spikes in $\boldsymbol{\zeta}$, $\boldsymbol{\Psi}_\iota$ and $\boldsymbol{\zeta}_\iota$ denote the ι-th entry of $\boldsymbol{\Psi}$ and $\boldsymbol{\zeta}$, respectively. The procedure of off-grid DOA estimation using OGNet is summarized as in Table 4[†], where the off-line training data and strategies for OGNet are introduced in the following section.

4. Network Training

The training is performed on a Windows PC equipped with 3.2 GHz AMD Ryzen 2700 CPU, 12 GB NVIDIA GeForce RTX 3060 GPU and 48 GB RAM. The generation of training data are performed by MATLAB, and the DL framework for constructing NN, training NN and simulations are based on TensorFlow 2.6.0 plus Python 3.7.12. The ADAM [36] with initial learning rate 0.001 is chosen as the optimizer for all networks within OGNet. In both training and testing phases, a ULA equipped with $M = 12$ antennas is considered. The data generation and training strategies for the AE and DCNN is described in detail as following.

4.1 Training of AE

To generate the training data of AE, we consider P sources lie in spatial from $-60°$ to $60°$. The true DOAs of sources are randomly sampled from the interval $[-60°, 60°]$ with a sampling step of $0.1°$, and the angular separation between any two DOAs is greater than $2°$. To generate sufficient data for training, 2×10^6 pairs of true DOA are randomly sampled from the interval. At each sample, let the SNR vary randomly in steps of 5 dB between -15 dB to 20 dB and fix the number of snapshots at $T = 50$. Then, the training data for AE is generated according to Eqs. (4), (5), (6), (9) and (11).

During training phase, the training data of AE is randomly divided into 90% for training and 10% for validation, hence $D_a = 2 \times 10^6 \times 0.9 = 1.8 \times 10^6$. The training for AE is carried out 100 epochs with a batch size of 1000. The learning rate drop factor is set as 0.5 with the drop period being 4 epochs.

4.2 Training of DCNN

In generating the training data for DCNN, ϕ is considered to be $60°$, and α is considered to be $\alpha = 2°$ for instance. Then, the spatial discrete grid is $\boldsymbol{\Psi} = \{-60°, -58°, -56°, \cdots, 0°, \cdots, 58°, 60°\}$ and $G = 61$. Since $\alpha = 2°$, the off-grid error $e_{g_p} \subset (-1, 1]$. The constant c is set as $c = 10$ in this paper, then $\xi_{g_p} \subset (0, 20]$. The number of sources varies from 1 to P_{max} to enable the DCNN can achieve multi-DOA estimation, where $P_{max} = 3$ is considered in this paper to relief the memory and system demands. To generate the off-grid DOAs, the on-grid angles of $P = 1, 2, 3$ sources are firstly generated from all possible combinations in $\boldsymbol{\Psi}$, i.e., we can obtain $C_{61}^1 + C_{61}^2 + C_{61}^3$ pairs of on-grid angles. Then, to release training burden, the off-grid error for each pair of on-grid angle is randomly chosen from $\boldsymbol{\varepsilon} = \{-0.9, -0.8, \cdots, -0.1, 0, 0.1, \cdots, 0.9, 1\}$ without overlap until all off-grid errors in $\boldsymbol{\varepsilon}$ have been traversed or the rest candidate off-grid errors are not enough to be assigned to the on-grid angles in the angle pair. Then, the total number of pairs of off-grid DOAs for training is $\lfloor 20/1 \rfloor \times C_{61}^1 + \lfloor 20/2 \rfloor \times C_{61}^2 + \lfloor 20/3 \rfloor \times C_{61}^3 = 235460$ with $\lfloor \cdot \rfloor$ denoting the round-down operator. After obtaining the 235460 pairs off-grid DOAs, the corresponding $\boldsymbol{R}_u$ (the training input for DCNN) with respect to each pair of off-grid DOAs at each SNR in $\{-15, -10, -5, 0, 5, 10, 15, 20\}$ dB is calculated by (4) and (6), which leads to the total number of training data being $235460 \times 8 = 1883680$. The output label $\boldsymbol{\xi}$ corresponding to each $\boldsymbol{R}_u$ is generated during generating the off-grid error according to (15), (16) and (17). For example, if the on-grid angle pair is $\{-60°, -56°\}$ and the

[†] The number of sources P is assumed to be known previously.

corresponding off-grid error is $\{-0.6°, 0.3°\}$, the output label $\boldsymbol{\xi}$ becomes $\boldsymbol{\xi} = [-0.6 \times 10 + 10, 0, 0.3 \times 10 + 10, 0, 0, \cdots, 0]^T$ where the indices of non-zeros elements in $\boldsymbol{\xi}$ is the same as the indices of the position of the corresponding angle pair in $\boldsymbol{\Psi}$.

When training the DCNN, the training data is randomly divided into 90% training set and 10% validation set, hence $D_c = 1883680 \times 0.9 = 1695312$. The training for DCNN is carried out 200 epochs with a batch size of 512, and the corresponding learning rate drop factor is set as 0.7 with the drop period being 5 epochs.

Remark 2. *It should be noted that the OGNet without AE (i.e., only DCNN) is also capable of achieving off-grid DOA estimation by taking the SUC as the input, even if the DCNN is trained using TUC. However, the estimation performance of DCNN is inferior compared to that of OGNet, as will be demonstrated in the simulation experiments later. On the other hand, the DCNN can be trained under different grid intervals α by using the similar data generation and training strategies, and such DCNN still has superior DOA estimation performance, which will also be demonstrated in the simulation experiments later.*

Remark 3. *The computational burden of OGNet is mainly dominated by the floating-point operations (FLOPs) in AE and DCNN when estimating DOA, while the FLOPs of AE and DCNN depend on the number of layers and neurons in each layer. Since AE is consisted by FC layers, its FLOPs is $2\sum_{i=1}^{N_a} I^{(i)}O^{(i)}$, where N_a denotes the number of FC layers in AE, and $I^{(i)}$ and $O^{(i)}$ represent the number of input and output neurons of each FC layer, respectively. While in DCNN, the FLOPs include two parts: FLOPs of Cov. layers and that of FC layers. The FLOPs of FC layers in DCNN can be similarly calculated as $2\sum_{i=1}^{N_d} I^{(i)}O^{(i)}$ with N_d being the number of FC layers in DCNN. The FLOPs of each Cov. layer in DCNN is $2C_I\kappa^2C_OM^2$ since all Cov. layers are identical in DCNN, then the total FLOPs of all Cov. layers is $2C_I\kappa^2C_OM^2N_{cov}$ where C_I and C_O represent the input and output channels of each convolution, k is the kernel size, and N_{cov} denotes the total number of Cov. layers. Thus the total FLOPs of OGNet is*

$$FLOPs = 2\sum_{i=1}^{N_a} I^{(i)}O^{(i)} + 2\sum_{i=1}^{N_d} I^{(i)}O^{(i)} + 2C_I\kappa^2C_OM^2N_{cov}. \tag{21}$$

Based on the similar calculation process, the FLOPs of CNN in [32] can be computed as $2\sum_{i=1}^{N_d} I^{(i)}O^{(i)} + 2C_I\kappa^2C_OM^2N_{cov}$. Obviously, according to the specific network parameters of CNN provided in [32], the computational complexity of the OGNet is a little more expensive than that of the CNN. While, since the DOA estimation achieved by NN is only based on simple additions and multiplications in trained networks, it is still within acceptable limits, which will be demonstrated as in the following section.

5. Simulation Experiments and Analyses

Numerous simulation experiments are conducted in this section to evaluate the effectiveness and superiority of the proposed OGNet. During simulations, the ULA is considered to be equipped with $M = 12$ antennas. The random independent far-field narrow-band signal is utilized for simulations. Firstly, the effectiveness of the proposed OGNet under different scenarios is evaluated. Then, the performance superiority of the OGNet is evaluated by comparing with the state-of-art NN-based methods and traditional model-driven methods.

5.1 Effectiveness of the OGNet

Firstly, the effectiveness of the AE within OGNet is evaluated by the difference between the theoretical $\boldsymbol{R}_u$ and the predicted $\hat{\boldsymbol{R}}_u$ by AE, which is referred as to predicted difference. The difference between $\boldsymbol{R}_u$ and the $\bar{\boldsymbol{R}}_u$, which is called original difference, is calculated for comparison. The number of sources is $P = 2$ with $\boldsymbol{\theta} = [-27.21°, 13.35°]$. The difference is evaluated by

$$\|\Delta\boldsymbol{R}_u\|_F = \frac{1}{N_{mc}}\sum_{i=1}^{N_{mc}} \|\boldsymbol{R}_u - \dot{\boldsymbol{R}}_u^{(i)}\|_F, \tag{22}$$

where $N_{mc} = 10^3$ denotes the total number of Monte Carlo trials, $\dot{\boldsymbol{R}}_u^{(i)}$ represents $\hat{\boldsymbol{R}}_u$ or $\bar{\boldsymbol{R}}_u$ at i-th Monte Carlo trial. The results of $\|\Delta\boldsymbol{R}_u\|_F$ under different SNRs and number of snapshots are given in Fig. 5 with the number of snapshots being $T = 500$ and SNR = 0 dB, respectively. As clearly shown in Fig. 5, the difference decreases with SNR increasing. Moreover, the predicted difference is significantly smaller than the original difference in all SNR cases. Similarly, one can also see that the difference decreases with increasing number of snapshots and the predicted difference is much smaller than

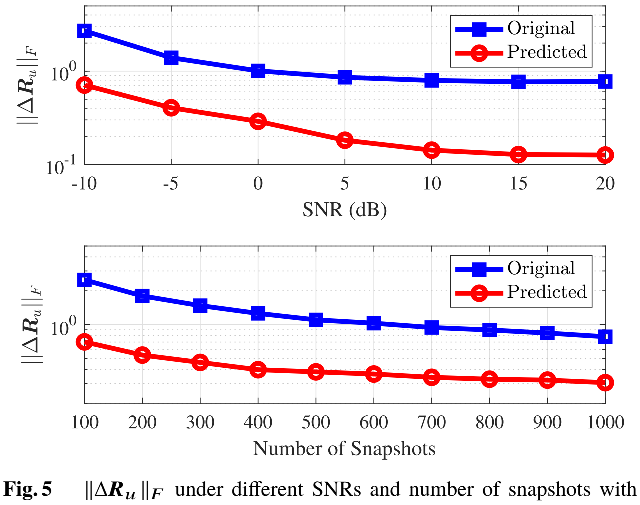


Fig. 5 $\|\Delta\boldsymbol{R}_u\|_F$ under different SNRs and number of snapshots with $P = 2$. Upper: divergence versus SNR. Lower: divergence versus number of snapshots.

the original difference in all snapshot cases. The results in Fig. 5 illustrate that the AE component within OGNet can effectively reduce the difference between the sampled $\bar{\boldsymbol{R}}_u$ and theoretical $\boldsymbol{R}_u$.

As previously claimed in Sect. 4.2, the OGNet without AE (i.e., only DCNN) can also achieve off-grid DOA estimation. Hence, the simulation experiments on DOA estimation RMSE of DCNN and OGNet is conducted to evaluate the effectiveness of the AE and DCNN with $P = 2$ and $\boldsymbol{\theta} = [-27.21°, 13.35°]$. The definition of RMSE is

$$\text{RMSE} = \sqrt{\frac{1}{N_{mc}P}\sum_{i=1}^{N_{mc}} |\boldsymbol{\theta} - \bar{\boldsymbol{\theta}}^{(i)}|_2^2}, \qquad (23)$$

where $N_{mc} = 10^3$ denotes the total number of Monte Carlo trials, $\boldsymbol{\theta}'^{(i)}$ denotes the estimated DOAs at i-th Monte Carlo trial. The corresponding results are given in Fig. 6, where the upper is the RMSE versus SNR with $T = 500$, and the lower is the RMSE versus number of snapshots with SNR = 0 dB. As can be seen from the Fig. 6 that the RMSEs of DCNN and OGNet are reasonably decreased with the increasing of SNR and snapshots. While, the RMSE of the OGNet is distinctly much smaller than that of the DCNN, especially at low SNR, which demonstrate that the DCNN can achieve off-grid DOA estimation alone and the AE can effectively improve its performance to achieve high-precision off-grid DOA estimation.

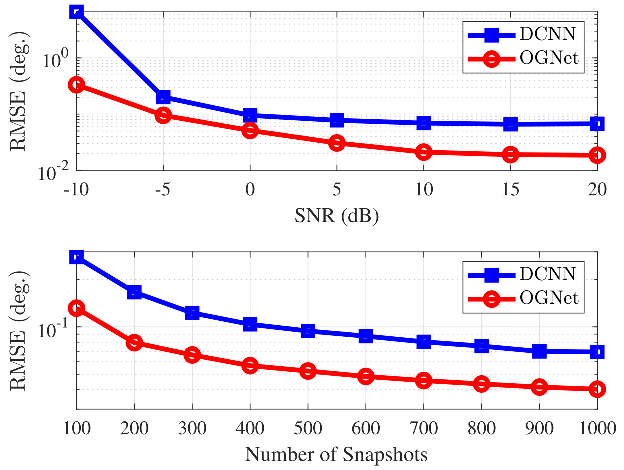


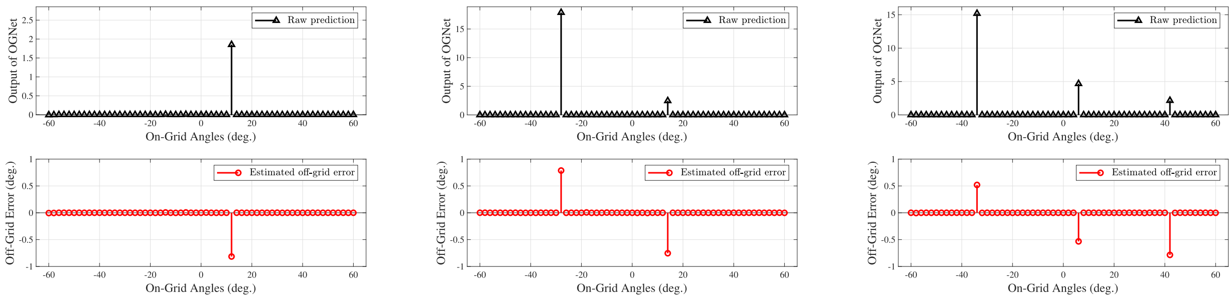
Fig. 6 RMSE of DCNN and OGNet under different SNRs and number of snapshots. Upper: RMSE versus SNR. Lower: RMSE versus number of snapshots.

Further, the effectiveness of the OGNet is evaluated by single-prediction simulation experiments with different number of sources P. When $P = 1$, the true DOA is fixed as [11.23°]. When $P = 2$ and $P = 3$, the true DOAs are fixed as $[-27.21°, 13.35°]$ and $[-33.56°, 5.42°, 41.37°]$, respectively. The simulations are conducted under SNR = 0 dB and $T = 500$. The prediction results of OGNet and the corresponding estimated off-grid errors with different number of sources P are shown in Fig. 7, and the corresponding specific numerical results of estimated off-grid errors and DOAs are given in Table 5. As can be seen in Fig. 7, in the case of different P, the raw predictions of OGNet and corresponding estimated off-grid error vector have obvious spikes and are very sparse. On the other hand, the specific numerical results in Table 5 show that the off-grid errors estimated by OGNet are very accurate for different P, thus the corresponding DOA estimation are precise. The mean estimation errors for $P = 1$, $P = 2$ and $P = 3$ reached 0.0448, 0.0531 and 0.0937, respectively, which indicates that the proposed OGNet can effectively achieve high-precision off-grid DOA estimation.

Lastly, different off-grid DOAs are estimated by using the proposed OGNet in different number of sources $P = 1, 2, 3$ to evaluate the effectiveness more widely. For different P, the initial DOA of the first source is fixed as $\theta_1 = -59.52°$, then other initial DOAs are $\theta_2 = \theta_1 + \Delta\theta$ and $\theta_3 = \theta_2 + \Delta\theta$ with $\Delta\theta = 5.3°$, respectively. Then, DOAs under different P are varies from the initial angles with an increasing step of 1° until all possible angles in the interval of $(-60°, 60°)$ are sampled. The estimation results with SNR = 0 dB and $T = 500$ are shown in Fig. 8. It is clearly demonstrated that the estimated DOAs by OGNet under different number of sources are very close to the true DOAs, which illustrate that the proposed OGNet is valid for different number of sources and different angles.

5.2 Superiority of the OGNet

The effectiveness of the proposed OGNet has been eval-



(a) $P = 1$.

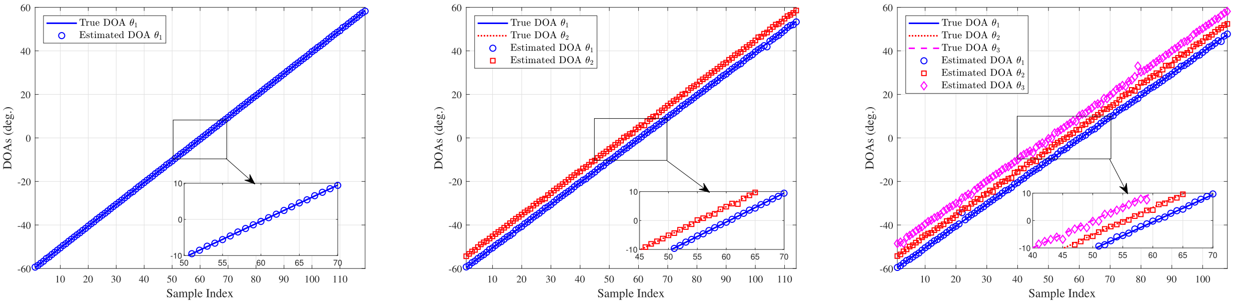
(b) $P = 2$.

(c) $P = 3$.

Fig. 7 Single-prediction results of OGNet and corresponding estimated off-grid errors with different number of sources P.

Table 5 Specific numerical results of estimated off-grid errors and DOAs by OGNet.

Sources	True off-grid errors	Estimated off-grid errors	True DOAs	Estimated DOAs
$P = 1$	$[-0.77°]$	$[-0.8148°]$	$[11.23°]$	$[11.1852°]$
$P = 2$	$[0.79°, -0.65°]$	$[0.7874°, -0.7536°]$	$[-27.21°, 13.35°]$	$[-27.2126°, 13.2464°]$
$P = 3$	$[0.44°, -0.58°, -0.63°]$	$[0.5292°, -0.5328°, -0.7848°]$	$[-33.56°, 5.42°, 41.37°]$	$[-33.4808°, 5.4672°, 41.2152°]$



(a) $P = 1$. (b) $P = 2$. (c) $P = 3$.

Fig. 8 Estimation results on different off-grid DOAs with different number of sources P by OGNet.

uated in the previous subsection. In this subsection, the performance superiority of the proposed OGNet is evaluated by comparing with the sate-of-the-art methods under $P = 2$. Unless otherwise specified, the corresponding true DOAs of sources are fixed as $\boldsymbol{\theta} = [-27.21°, 13.35°]$. The methods introduced for comparison include MUSIC [3], off-grid sparse Bayesian inference (OGSBI) [17] and CNN [32]. Additionally, the conditional Cramér-Rao bound (CRB) [37] is calculated for comparison. The evaluation metric is the RMSE defined in Eq. (23).

First of all, the normalized spectra of single DOA estimation for different methods are given in Fig. 9 with SNR = 0 dB and $T = 500$. Note that the CNN and the proposed OGNet do not really have a spectrum, hence the DOA estimation results of CNN and OGNet are shown as solid lines placed directly in the figure for intuitive comparison. As clearly shown in Fig. 9, the DOAs estimated by the proposed OGNet are closer to the true DOAs than that of the CNN, also than the spectrum peaks of the other comparison methods, which indicate that the proposed OGNet has higher precision of DOA estimation.

Afterward, the computational complexity of different methods are compared by their average time required for a single DOA estimation, the corresponding result is given in Table 6. The results are based on 10^3 independent simulations, $T = 500$, SNR = 0 dB and $\boldsymbol{\theta} = [-27.21°, 13.35°]$. As can be seen from Table 6, since the computational complexity of the proposed OGNet is higher than that of the other comparison methods, its average time for a single DOA estimation is reasonably longer than that of its rivals. Despite this, the computational complexity of the proposed OGNet is still within acceptable limits and can fulfill the requirements of real-time estimation.

Then, the simulation experiments for RMSE and probability of successful detection (PSD) of DOA estimation of different methods in terms of SNR are carried out to evaluate the superiority of the OGNet. The criteria for successful detection is $\sqrt{\frac{1}{P}|\boldsymbol{\theta} - \boldsymbol{\theta}'|_2^2} < \tau$ with $\tau = 0.3$ in this paper. The results are depicted in Figs. 10 and 11 with $T = 500$. In Fig. 10, one can find that the RMSE of CNN does not decreased anymore when SNR > 0 dB since it's an on-grid method that have a precision restriction for off-grid angles under a coarse searching grid. Conversely, the RMSE of OGNet continues to decrease as the SNR increases since it can well handle the off-grid error. Meanwhile, the proposed OGNet possesses the lowest RMSE among the other methods. But what cannot be ignored is that when SNR ≥ 10 dB, the RMSE of our proposed method seems to deviate from

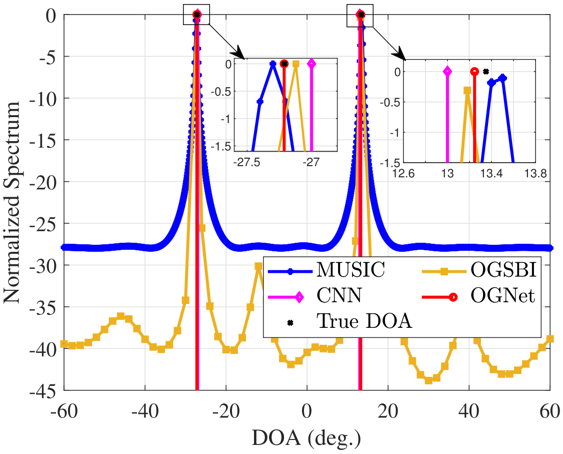


Fig. 9 Normalized spectra of different methods.

Table 6 The average time required for a single DOA estimation based on different methods.

Methods	Average time (sec.)
MUSIC	0.0127
OGSBI	0.0439
CNN	0.0518
OGNet	0.0595

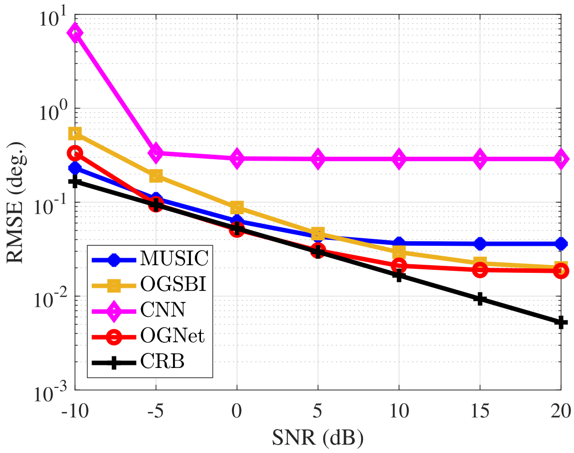


Fig. 10 RMSE of different methods under different SNRs.

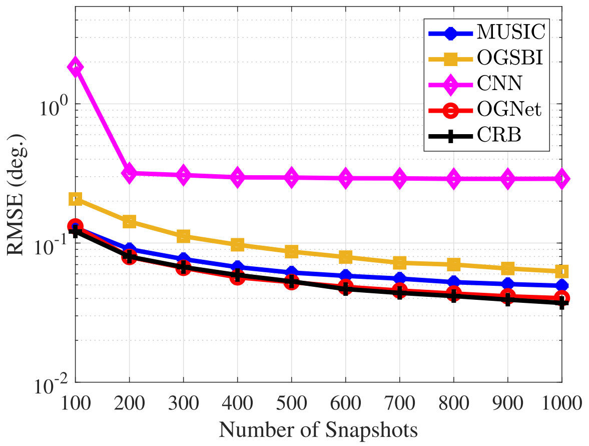


Fig. 12 RMSE of different methods under different number of snapshots.

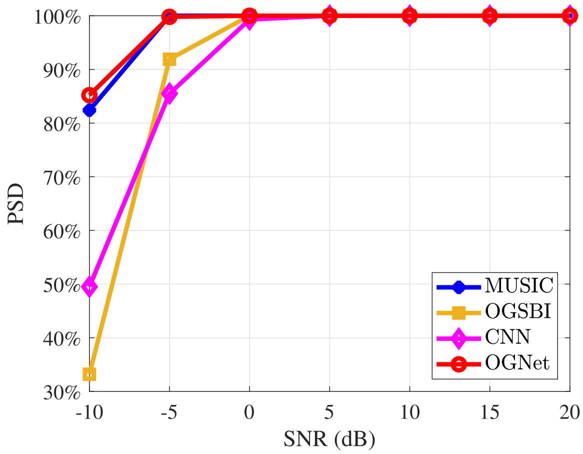


Fig. 11 PSD of different methods under different SNRs.

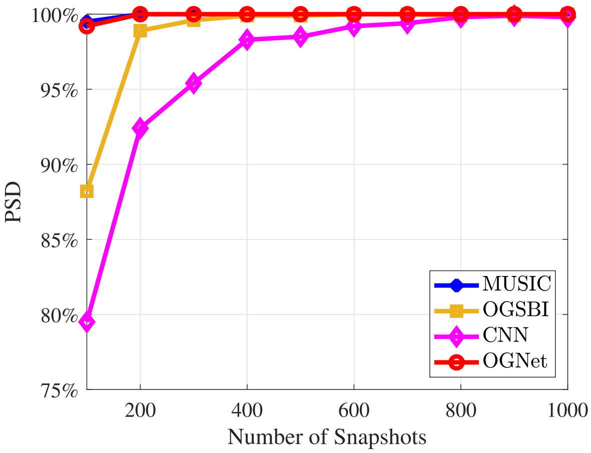


Fig. 13 PSD of different methods under different number of snapshots.

CRB and has a floor effect[†]. On the other hand, it can be easily found in Fig. 11 that the PSD of the proposed OGNet is higher than that of all the comparison methods at low SNRs, and reaches 100% faster than the other methods. These results indicate that the proposed OGNet can achieve high-precision off-grid DOA estimation and has distinct superiority under different SNRs.

Next, the simulation experiments for RMSE and PSD of DOA estimation of different methods in terms of number of snapshots are conducted to evaluate the superiority of the OGNet. The results are given in Figs. 12 and 13 with SNR = 0 dB. From the Fig. 12, we can find that the RMSE of the proposed OGNet is the lowest and is closest to the CRB in all cases of number of snapshots. While, in Fig. 13, the PSD of proposed method is similar to that of MUSIC and higher than that of other methods at small snapshots, and reaches 100% faster than other rivals. The results in Fig. 12 and 13 further illustrate the superiority of the proposed OGNet under different number of snapshots.

Further, the RMSE of different methods under different angular separations are compared. The corresponding result is depicted in Fig. 14, in which $T = 500$, SNR = 0 dB and the true DOA is set as $\boldsymbol{\theta} = [\theta_1, \theta_1 + \Delta\theta]$ with $\theta_1 = -3.67°$ and $\Delta\theta$ changing from 1° to 5°. As clearly shown in Fig. 14, the proposed OGNet shows lower RMSE as compare to other comparison methods in close proximity, which indicate that the proposed OGNet has obviously performance advantage in terms of resolution.

In the end, the RMSE of different methods in the case of different grid intervals are evaluated. Since the CNN method is an on-grid method whose performance become poor or even invalid for the off-grid angles in coarse grid, only the OGSBI method is introduced to compare with the proposed OGNet. The different grid intervals are set as $\alpha = \{2°, 3°, 4°, 5°\}$. The data generation and training strategies for OGNet when $\alpha = 2°$ have been introduced in detail in previous Sect. 4.2. While, the data generation strategies

[†] This is because that deep networks are biased estimators (this holds for all DL-based estimators and not only the proposed) and the accuracy is limited by some factors such as network architecture and the size of training data-set. With fixed network structure and training data, when the model training is done, there must be an error between its prediction results and labels, resulting in the upper limit of its performance. When the model reaches this upper limit as SNR increases, it is difficult to show further improvements.

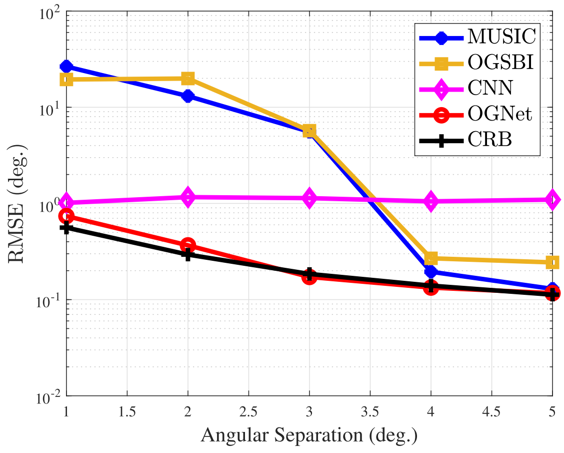


Fig. 14 RMSE of different methods under different angular separations.

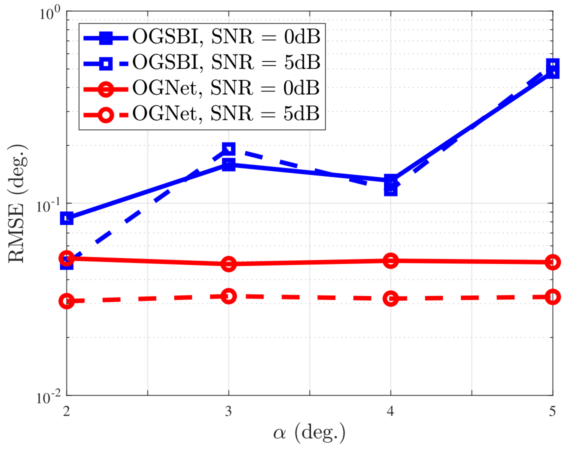


Fig. 15 RMSE of OGSBI and OGNet under different grid intervals.

for $\alpha = \{3°, 4°, 5°\}$ are the same as that for $\alpha = 2°$, and the training strategies is similar with that for $\alpha = 2°$ except for the batch size decreasing to 256. The results is shown in Fig. 15, as in which that the RMSE of the OGSBI increases with the increase of α, while the RMSE of OGNet remains stable with the increase of α. On the other hand, OGNet shows better performance than OGSBI in different SNRs and grid intervals. The results demonstrate that the proposed OGNet has performance benefits while being robust to different grid intervals.

6. Conclusion

In this paper, a cascade DNN named OGNet is designed for off-grid DOA estimation. Specifically, the upper triangular part of the sampling unitary covariance of array receiving data is firstly taken as the input of AE to reduce the difference between it and the theoretical, then the predicted unitary covariance of AE is input into the DCNN to predict the sparse off-grid error vector. The ultimate DOA estimation is realized by using the sparsity of the output of DCNN, which enables that the proposed OGNet can realize high-precision off-grid DOA estimation without a priori on-grid DOA estimation. The results under various scenarios indicate that the DOA estimation performance and resolution of the OGNet is remarkable and has noticeable advantages over its rivals.

References

[1] H. Krim and M. Viberg, "Two decades of array signal processing research: The parametric approach," IEEE Signal Process. Mag., vol.13, no.4, pp.67–94, July 1996.

[2] W. Liu, M. Haardt, M.S. Greco, C.F. Mecklenbräuker, and P. Willett, "Twenty-five years of sensor array and multichannel signal processing: A review of progress to date and potential research directions," IEEE Signal Process. Mag., vol.40, no.4, pp.80–91, 2023.

[3] R. Schmidt, "Multiple emitter location and signal parameter estimation," IEEE Trans. Antennas Propag., vol.34, no.3, pp.276–280, March 1986.

[4] R. Roy and T. Kailath, "ESPRIT-estimation of signal parameters via rotational invariance techniques," IEEE Trans. Acoust., Speech, Signal Process., vol.37, no.7, pp.984–995, July 1989.

[5] A. Barabell, "Improving the resolution performance of eigenstructure-based direction-finding algorithms," IEEE International Conference on Acoustics, Speech and Signal Processing, pp.336–339, Jan. 1983.

[6] B. Rao and K. Hari, "Performance analysis of root-MUSIC," IEEE Trans. Acoust., Speech, Signal Process., vol.37, no.12, pp.1939–1949, Dec. 1989.

[7] T.J. Shan, M. Wax, and T. Kailath, "On spatial smoothing for direction-of-arrival estimation of coherent signals," IEEE Trans. Acoust., Speech, Signal Process., vol.33, no.4, pp.806–811, Aug. 1985.

[8] K.C. Huarng and C.C. Yeh, "A unitary transformation method for angle-of-arrival estimation," IEEE Trans. Signal Process., vol.39, no.4, pp.975–977, April 1991.

[9] M. Haardt and J. Nossek, "Unitary ESPRIT: How to obtain increased estimation accuracy with a reduced computational burden," IEEE Trans. Signal Process., vol.43, no.5, pp.1232–1242, May 1995.

[10] D. Donoho, "Compressed sensing," IEEE Trans. Inf. Theory, vol.52, no.4, pp.1289–1306, April 2006.

[11] K. Hayashi, M. Nagahara, and T. Tanaka, "A user's guide to compressed sensing for communications systems," IEICE Trans. Commun., vol.E96-B, no.3, pp.685–712, March 2013.

[12] T. Terada, T. Nishimura, Y. Ogawa, T. Ohgane, and H. Yamada, "DOA estimation for multi-band signal sources using compressed sensing techniques with Khatri-Rao processing," IEICE Trans. Commun., vol.E97-B, no.10, pp.2110–2117, Oct. 2014.

[13] Y. Liu, Z. Zhang, C. Zhou, C. Yan, and Z. Shi, "Robust variational Bayesian inference for direction-of-arrival estimation with sparse array," IEEE Trans. Veh. Technol., vol.71, no.8, pp.8591–8602, 2022.

[14] Z. Yang, J. Li, P. Stoica, and L. Xie, "CHAPTER 11 - Sparse methods for direction-of-arrival estimation," Academic Press Library in Signal Processing, Volume 7, R. Chellappa and S. Theodoridis, eds., pp.509–581, Academic Press, 2018.

[15] D. Malioutov, M. Cetin, and A. Willsky, "A sparse signal reconstruction perspective for source localization with sensor arrays," IEEE Trans. Signal Process., vol.53, no.8, pp.3010–3022, Aug. 2005.

[16] M. Tipping, "Sparse Bayesian learning and the relevance vector machine," J. Mach. Learn. Res., vol.1, no.3, pp.211–244, Sept. 2001.

[17] Z. Yang, L. Xie, and C. Zhang, "Off-grid direction of arrival estimation using sparse Bayesian inference," IEEE Trans. Signal Process., vol.61, no.1, pp.38–43, Oct. 2012.

[18] J. Dai, X. Bao, W. Xu, and C. Chang, "Root sparse Bayesian learning for off-grid DOA estimation," IEEE Signal Process. Lett., vol.24, no.1, pp.46–50, Jan. 2017.

[19] Z. Yang, L. Xie, and C. Zhang, "A discretization-free sparse and parametric approach for linear array signal processing," IEEE Trans. Signal Process., vol.62, no.19, pp.4959–4973, Oct. 2014.

[20] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," Nature, vol.521, no.7553, pp.436–444, May 2015.

[21] I. Goodfellow, Y. Bengio, and A. Courville, Deep Learning, MIT Press., 2016.

[22] Y. Kase, T. Nishimura, T. Ohgane, Y. Ogawa, D. Kitayama, and Y. Kishiyama, "Fundamental trial on DOA estimation with deep learning," IEICE Trans. Commun., vol.E103-B, no.10, pp.1127–1135, Oct. 2020.

[23] J. Cong, X. Wang, C. Yan, L.T. Yang, M. Dong, and K. Ota, "CRB weighted source localization method based on deep neural networks in multi-UAV network," IEEE Internet Things J., vol.10, no.7, pp.5747–5759, 2023.

[24] Y. Kase, T. Nishimura, T. Ohgane, Y. Ogawa, T. Sato, and Y. Kishiyama, "Accuracy improvement in DOA estimation with deep learning," IEICE Trans. Commun., vol.E105-B, no.5, pp.588–599, May 2022.

[25] G.K. Papageorgiou and M. Sellathurai, "Direction-of-arrival estimation in the low-SNR regime via a denoising autoencoder," IEEE International Workshop on Signal Processing Advances in Wireless Communications, pp.1–5, Aug. 2020.

[26] G.K. Papageorgiou and M. Sellathurai, "Fast direction-of-arrival estimation of multiple targets using deep learning and sparse arrays," IEEE International Conference on Acoustics, Speech and Signal Processing, pp.4632–4636, April 2020.

[27] A. Barthelme and W. Utschick, "DoA estimation using neural network-based covariance matrix reconstruction," IEEE Signal Process. Lett., vol.28, pp.783–787, 2021.

[28] X. Wu, X. Yang, X. Jia, and F. Tian, "A gridless DOA estimation method based on convolutional neural network with Toeplitz prior," IEEE Signal Process. Lett., vol.29, pp.1247–1251, May 2022.

[29] Z.M. Liu, C. Zhang, and P.S. Yu, "Direction-of-arrival estimation based on deep neural networks with robustness to array imperfections," IEEE Trans. Antennas Propag., vol.66, no.12, pp.7315–7327, Dec. 2018.

[30] J. Cong, X. Wang, M. Huang, and L. Wan, "Robust DOA estimation method for MIMO radar via deep neural networks," IEEE Sensors J., vol.21, no.6, pp.7498–7507, March 2021.

[31] L. Wu, Z.M. Liu, and Z.T. Huang, "Deep convolution network for direction of arrival estimation With sparse prior," IEEE Signal Process. Lett., vol.26, no.11, pp.1688–1692, Nov. 2019.

[32] G.K. Papageorgiou, M. Sellathurai, and Y.C. Eldar, "Deep networks for direction-of-arrival estimation in low SNR," IEEE Trans. Signal Process., vol.69, pp.3714–3729, June 2021.

[33] H. Wang, X. Wang, M. Huang, L. Wan, and T. Su, "RxCV-based unitary SBL algorithm for off-grid DOA estimation with MIMO radar in unknown non-uniform noise," Digit. Signal Process., vol.116, p.103119, Sept. 2021.

[34] X.T. Meng, F.G. Yan, B.X. Cao, M. Jin, and Y. Zhang, "Efficient real-valued DOA estimation based on the trigonometry multiple angles transformation in monostatic MIMO radar," Digit. Signal Process., vol.123, p.103437, 2022.

[35] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," Proc. 32nd International Conference on Machine Learning, F. Bach and D. Blei, eds., Proc. Machine Learning Research, vol.37, pp.448–456, PMLR, July 2015.

[36] D.P. Kingma and J. Ba, "Adam: A method for stochastic optimization," arXiv preprint arXiv:1412.6980, Dec. 2014.

[37] P. Stoica and A. Nehorai, "Performance study of conditional and unconditional direction-of-arrival estimation," IEEE Trans. Acoust., Speech, Signal Process., vol.38, no.10, pp.1783–1795, Oct. 1990.

Huafei Wang was born in 1995. He received the B.S. degree and M.S. degrees from Hainan University, Haikou, China, in 2017 and 2020, respectively. He is currently pursuing the Ph.D. degree in information and communication engineering at Hainan University. His current research interests include array signal processing, radar signal processing using learning strategy.

Xianpeng Wang was born in 1986. He received the M.S. and Ph.D. degrees from the College of Automation, Harbin Engineering University, Harbin, China, in 2012 and 2015, respectively. He was a full-time Research Fellow with the School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore, from 2015 to 2016. He is currently a Professor with the School of Information and Communication Engineering, Hainan University, Haikou, China. He is the author of more than 100 papers published in related journals and international conference proceedings and was a Reviewer of more than 30 journals. His major research interests include communication system, array signal processing, radar signal processing, compressed sensing, and its applications.

Xiang Lan received the B.S. degree from the Huazhong University of Science and Technology, China, in 2012, and the M.Sc. and Ph.D. degrees from the Department of Electronic and Electrical Engineering, The University of Sheffield, U.K., in 2014 and 2019, respectively. From 2019 to June 2020, he worked as a Research Associate with the Department of Electronic and Electrical Engineering, The University of Sheffield. He is currently a Lecturer with the School of Information and Communication Engineering, Hainan University, China. His research interests include signal processing based on vector sensor arrays (beamforming and DOA estimation with polarized signals) and sparse array processing.

Ting Su received the B.S. degree in communication engineering and the Ph.D. degree in electronic science and technology from the Nanjing University of Science and Technology, Nanjing, China, in 2006 and 2016, respectively. She is currently a Postdoctoral Re-searcher with the Institute of Communications Engineering, Army Engineering University of PLA, Nanjing. Moreover, she is also a Lecturer with Hainan university, Haikou, China. Her research interests include computational electromagnetic, radar signal processing, and wireless communications.