IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received December 10, 2018, accepted December 22, 2018, date of publication January 1, 2019, date of current version February 20, 2019.

Digital Object Identifier 10.1109/ACCESS.2018.2890579

Sparse Detection Algorithms Based on Two-Dimensional Compressive Sensing for Sub-Nyquist Pulse Doppler Radar Systems

BEIYI LIU[1], (Student Member, IEEE), YU ZHAO[2], (Student Member, IEEE), XIAOMEI ZHU[3], SHINYA MATSUSHITA[1], AND LI XU[1], (Senior Member, IEEE)

[1]Department of Intelligent Mechatronics, Akita Prefectural University, Akita 0150055, Japan
[2]Key Laboratory of Broadband Wireless Communication and Sensor Network Technology, Nanjing University of Posts and Telecommunications, Ministry of Education, Nanjing 210003, China
[3]College of Computer Science and Technology, Nanjing Tech University, Nanjing 210028, China

Corresponding author: Yu Zhao (1018010409@njupt.edu.cn)

This work was supported in part by the Japan Society for the Promotion of Science (JSPS) under Research Grant 15K06072, in part by the National Natural Science Foundation of China under Grant 61501223, in part by the Jiangsu Specially Appointed Professor under Grant RK002STP16001, in part by the Innovation and Entrepreneurship of Jiangsu High-level Talent under Grant CZ0010617002, in part by NUPTSF under Grant NY2015026, and in part by 1311 the Talent Plan of Nanjing University of Posts and Telecommunications.

ABSTRACT Sub-Nyquist pulse Doppler radar system has received wide attention recently because it can make the sampling rate lower than the Nyquist sampling rate. However, there are still two problems that must be addressed for sub-Nyquist pulse Doppler radar. One is that the large memory is required for sparse signal recovery. Another is that the performance of the sparse signal recovery will be distorted in the non-Gaussian impulse noise environment. For the first problem, this paper proposes four 2-D compressive sensing (CS) algorithms which are extended from the traditional 1-D CS algorithms (the ZAP, IHT, ISTA, and FISTA algorithms). The proposed 2-D-CS algorithms recover the signal in the delay-Doppler domain, which is a matrix domain. For the second problem, robust 2-D-CS algorithms are proposed for the non-Gaussian impulse noise environment. The proposed 2-D-CS algorithms can achieve comparable detection performance with lower memory requirement. The proposed robust 2-D-CS algorithms can eliminate the interference of impulsive noise in the non-Gaussian impulse noise environment. Simulation results are given to verify the effectiveness of the proposed algorithms.

INDEX TERMS Pulse Doppler radar, compressive sensing, sparse representation, Doppler frequency.

I. INTRODUCTION

Radar, an object detection system utilizing radio waves to determine the range and velocity of objects, has been applied in many fields, such as ground penetrating radar, ballistic missile defense, air-traffic control, law-enforcement and highway safety and planetary exploration [1], [2]. The demand of the modern radar system for high resolution is increasing. According to Shannon-Nyquist theorem, it causes higher bandwidth radar signals—requiring higher sampling rates. Thus, the modern radar system requires the high rate analog-to-digital convert (ADC), and a large memory capacity for large amounts of sampled data [2], [3]. On the other hand, only few targets are concerned in practice, such as some airplanes in the wide sky, thus the final output is often sparse. An example of sparse scene is shown in Fig. 1. Hence, it is not efficient to detect the sparse target utilizing the large amounts of sampled data.

Recently, compressive sensing (CS) theory has been proposed to reduce the sampling rate for signals which can be represented as sparse signals in a specific domain [4], [5]. Evolving from CS theory, a novel sampling method, which enables the successful reconstruction of signals by sampling at the sub-Nyquist rate, has attracted considerable research attention, such as the random convolution (RC) [6], the random filter (RF) [7], the random demodulation (RD) [8], and the modulated wideband converter (MWC) [9]. The RD and MWC can both be thought of as being based on the underlying concept of the RF and RC. The RD shares the similar structure to the MWC, while the RD is a single channel sub-Nyquist sampling strategy and the MWC is a multi-channel

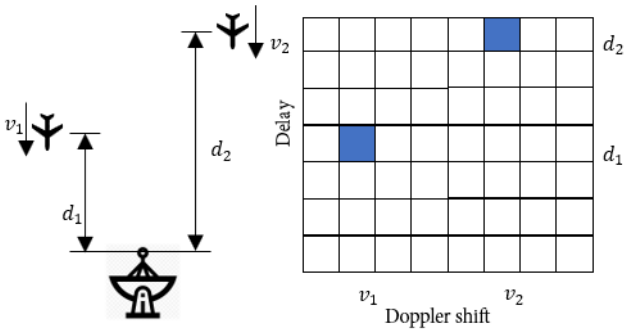

FIGURE 1. The delay-Doppler map.

sub-Nyquist sampling strategy [10]. Among them, the RD is a very attractive technique in microwave spectral analysis [11], because it is very simple and its prototype hardware has already been developed in [12] and [13].

A lot of CS-based algorithms have been proposed to reconstruct the signal sampled by CS sampling methods [14]–[19]. Orthogonal matching pursuit (OMP) algorithm and compressive sampling matching pursuit (CoSaMP) algorithm [14] are the greedy method that build up the support set of the reconstructed sparse vector iteratively by adding indices of the elements that are non-zero to the current support set at each iteration. Iterative hard thresholding (IHT) algorithm [15] is a thresholding method that keeps the indices of the elements that are non-zero of the sparse vector by a hard thresholding at each iteration. Zero-attractive projection (ZAP) algorithm [16] utilizes the zero attraction term to attract the vector from a least-square solution to a sparse solution.

Frequency determination based on robust Chinese remainder theorem (CRT) with the sub-Nyquist sampling method has been proposed in [20] and [21], and it has been applied in the multichannel synthetic aperture radar (SAR) [22]. The feasibility of CS-based pulse Doppler radar for delay-Doppler joint estimation has been shown in [3] and [23]–[26]. Although these methods estimate both delays and Doppler shifts from few sampled data, there are the following problems:

1. They use the traditional 1D-CS algorithm to solve the 2D delay-Doppler signal. The 2D signal should be stacked into a huge column vector based on the vector space, and then recover the huge vector in the 1D domain. However, such operation increases exponentially the complexity and memory usage [27], [28].

2. In the pulse Doppler radar system, the signal is usually contaminated with non-Gaussian impulsive noise such as environmental effects of atmospherics (lighting) and meteor train echoes [29]. Hence, robust CS algorithms for non-Gaussian impulsive noise are necessary for pulse Doppler radar system.

For the first point, the 2D orthogonal matching pursuit (2D-OMP) algorithm has been proposed in [27]. It is extended from the OMP algorithm. This algorithm significantly reduces the complexity, but it still requires a large memory.

For the second point, the Lorentzian IHT (LIHT) algorithm has been proposed to reduce the influences of the impulsive noise [30], [31]. However, it is for the 1D signal. The 2D signal should be stacked into a huge vector, first. It will increase complexity and memory usage.

In this paper, we propose a 2D data model for pulse Doppler radar system with the RD. First, the received signal is sampled by the RD with a low rate ADC. Then 2D-CS algorithms, including the 2D-ZAP, 2D-IHT, 2D-ISTA, and 2D-FISTA algorithms, are proposed to estimate the delays and Doppler shifts of sparse targets. Since the 2D-CS algorithms solve the 2D data model without stacking the matrix of 2D signals into a huge column vector, the memory requirement and complexity are reduced. Moreover, robust 2D-CS algorithms, including the 2D-RZAP and 2D-RIHT algorithms, are proposed for non-Gaussian impulsive noise environment. Numerical simulations were provided to validate the performances of our proposed algorithms.

The rest of this paper is organized as follows. In Section 2, the 2D system mathematical model for the proposed method is presented. In Section 3, 2D-CS algorithms are proposed for estimating the delays and Doppler shifts of sparse targets in Gaussian noise. In Section 4, we extend 2D-RZAP and 2D-RIHT algorithms from 2D-ZAP and 2D-IHT algorithms to estimate the delays and Doppler shifts of sparse targets in non-Gaussian impulsive noise. In Section 5, some experiments using proposed 2D CS algorithms are presented. Finally, the conclusion is made in Section 5.

II. SYSTEM MODEL

The pulse Doppler radar systems transmit a frequency or phase modulated pulse sequence to suppress clutter and detect targets at each angle of interest. The excellent references for the mathematical model of radar processing are available in the literature (see, e.g., [1]). In this paper, a classical pulse Doppler radar with a co-located receiver and a transmitter is considered. Assuming that the transmitted signal hits K non-fluctuating point targets with delays of τ_k and Doppler shifts of υ_k, $(1 \leq k \leq K)$, the received signal can be written as:

$$y(t) = \sum_{k=1}^{K} \alpha_k x\left(t - \tau_k\right) e^{j2\pi \nu_k t} + w(t), \tag{1}$$

where $x(t)$ is the transmitted signal; α_k denotes the reflectivity of the individual target; $w(t)$ is the noise. When the transmitted signal $x(t)$ is known, the received signal $y(t)$ is decided by the target parameters — delays τ_k, Doppler shifts υ_k and the reflectivity α_k. Estimating the target parameters can be treated as the problem of identification of underspread linear system (ULSs) [32], whose responses locate in a unit-area region in the delay-Doppler map $[\tau_0, \tau_f] \times [\nu_0, \nu_f]$ $(\tau_k \in [\tau_0, \tau_f]$ and $\nu_k \in [\nu_0, \nu_f])$. The values of τ_0 and τ_f are determined by the range and the values of ν_0 and ν_f are determined by the velocity of the possible targets, which can be written

as [32]

$$y(t) = \sum_{n=1}^{N} \alpha_n x\left(t - \tau_k\right) e^{j2\pi v_n t} + w(t), \tag{2}$$

where $\tau_n \in [\tau_0, \tau_f]$ denotes the possible delay. α_n is the amplitude. If there is a target in τ_n, $|\alpha_n|$ should be larger than zero, otherwise zero.

When $v_k \ll 1/p_w$, where p_w is the pulse width, the Doppler shift is difficult to estimate by only single pulse. Hence, multiple pulse signals are required to be transmitted for Doppler shift detecting, which can be written as [33]:

$$y(\Delta t, p) = \sum_{n=1}^{N}\sum_{m=1}^{M} \alpha_{nm} x\left(\Delta t - \tau_n\right) e^{j2\pi v_m p T_{PRI}} + w(\Delta t, p), \tag{3}$$

where $p \in [1, ..., P]$ is the pulse number; T_{PRI} is the pulse repetition interval (PRI); $\Delta t = t - pT_{PRI}$, and $x(\Delta t - \tau_k) = x(pT_{PRI} + \Delta t - \tau_k) = x(t - \tau_k)$ due to the periodicity of the transmitted pulse signal. The amplitude a_{nm} in the grid (τ_n, v_m) is determined by the reflectivity. If there is a target at (τ_n, v_m), $|a_{nm}|$ should be larger than zero, otherwise zero. The goal is to detect the positions of the nonzero values of $|a_{nm}|$ with unknown K. In general, the number of real targets is much smaller than the possible targets ($K \ll M \times N$). Hence, it is not effective that the received signal $y(t)$ is sampled at twice the bandwidth of the transmitted signal as demanded by the Nyquist sampling theorem.

The RD has been proposed to sample the received signal y to reduce the sampling rate [8]. It consists of three main components: demodulation, analog filter and uniform sampling. The grid diagram of the RD is shown in Fig. 2. In the demodulation step, the received signal is mixed with a pseudo-random noise (PN) sequence of ± 1 which repeats at each pulse-repetition time. The PN sequence is called as $p_c(t)$, and it must alternate between two values at or faster than the Nyquist frequency of the received signal. The demodulation step spreads the frequency content of the received signal, and allows that the information contained in the received signal has not been lost after low pass filter with impulse response $h(t)$. At last, the signal is sampled at rate $\mathcal{M}$ by a traditional ADC. The sampled signal can be expressed as:

$$y_{sub}[r, p] = \int_{-\infty}^{\infty} y\left(t_\tau, p\right) p_c\left(t_\tau\right) h\left(t - t_\tau\right) dt_\tau |_{t=rM}$$

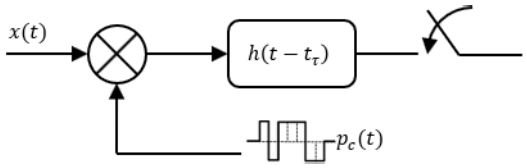

FIGURE 2. The grid diagram of the RD.

$$= \sum_{n=1}^{N}\sum_{m=1}^{M} \alpha_{nm} \times \int_{-\infty}^{\infty} x\left(\Delta t - \tau_n\right) p_c\left(t_\tau\right) \times h\left(rM - t_\tau\right) dt_\tau e^{j2\pi v_m p T_{PRI}} \tag{4}$$

where $r \in [1, ..., R]$ is the sample times in one pulse repetition time. Due to the sub-Nyquist sampling rate, R is much smaller than N. The goal is to estimate parameters: $\{\alpha_{nm}, \tau_n, v_m\}$ from a few sampled signal $y_{sub}[r, p]$. It can be written as a matrix form:

$$\boldsymbol{Y}_{sub} = \boldsymbol{A}\boldsymbol{\Psi}\boldsymbol{S}\boldsymbol{E}^H + \boldsymbol{W} = \boldsymbol{H}\boldsymbol{P}_{\boldsymbol{c}}\boldsymbol{\Psi}\boldsymbol{S}\boldsymbol{E}^H + \boldsymbol{W}, \tag{5}$$

where $\boldsymbol{W}$ denotes the noise, $\boldsymbol{Y}_{sub} = [\boldsymbol{y}(1), ..., \boldsymbol{y}(P)]$, $(1 \le p \le P)$, X^H denotes the transposed and complex conjugated matrix of X, and $\boldsymbol{A} = \boldsymbol{H}\boldsymbol{P}_{\boldsymbol{c}}$ denotes the RD system. $\boldsymbol{P}_{\boldsymbol{c}}$ is a diagonal matrix which can be written as:

$$\boldsymbol{P}_{\boldsymbol{c}} = \begin{bmatrix} p_c(t_{\tau 1}) & & & \\ & p_c(t_{\tau 2}) & & \\ & & \ddots & \\ & & & p_c(t_{\tau N}) \end{bmatrix}. \tag{6}$$

It denotes the demodulation processing. $\boldsymbol{H}$ is an integrator matrix:

$$\boldsymbol{H} = \begin{bmatrix} \overbrace{1 \ldots 1}^{\lceil N/R \rceil} & & & \\ & \overbrace{1 \ldots 1}^{\lceil N/R \rceil} & & \\ & & \ddots & \\ & & & \overbrace{1 \ldots 1}^{N - \lceil N/R \rceil (R-1)} \end{bmatrix}, \tag{7}$$

where $\lceil \cdot \rceil$ denotes ceiling operator. $\boldsymbol{\Psi}$ is the delay line of the transmitted pulse $x(t)$. More specifically, $\boldsymbol{\Psi} = [\phi_1, ..., \phi_N]$, where $\phi_n = x(t - \tau_n)$. Matrix $\boldsymbol{E}^H$ is a twiddle factor, which is used for any data-independent multiplicative constant in a fast Fourier transform (FFT). Matrix $\boldsymbol{S}$ is the delay-Doppler map whose element α_{nm} denotes the echo power of the target with delay τ_n and Doppler shift v_m.

However, the traditional CS theory is only applicable to 1D signal, while $\boldsymbol{Y}_{sub}$ and $\boldsymbol{S}$ are matrices (2D signal). In general, Eq. (5) is rewritten as:

$$vec\left(\boldsymbol{Y}_{sub}\right) = \left(\boldsymbol{E} \otimes \boldsymbol{\Phi}\right) vec\left(\boldsymbol{S}\right) = \boldsymbol{\Phi}_v vec\left(\boldsymbol{S}\right), \tag{8}$$

where $\boldsymbol{\Phi} = \boldsymbol{A}\boldsymbol{\Psi}$, and $\otimes$ denotes Kronecker product, $\boldsymbol{\Phi}_v = \boldsymbol{E} \otimes \boldsymbol{\Phi}$, $vec(\cdot)$ denotes the vectorization of a matrix by stacking the columns of the matrix into a single column vector. Therefore, some properties of 1D-CS are also suitable for Eq. (5). However, it increases the memory requirement to save $\boldsymbol{\Phi}_v$. In the following section, 2D-CS without the vectorization is proposed for Eq. (5). Utilizing 2D-CS, the memory requirement and the reconstruction time can be reduced to less than 1D-CS.

III. THE PROPOSED ALGORITHM FOR GAUSSIAN NOISE CASE

In this work, four different algorithms are proposed to search a sparse solution in Eq. (5), i.e., 2D zero-attractive projection (2D-ZAP) algorithm, 2D iterate hard thresholding (2D-IHT) algorithm, 2D iterative shrinkage-thresholding algorithm (2D-ISTA) and 2D fast iterative shrinkage-thresholding algorithm (2D-FISTA), which are described in detail in the following.

Inspired by traditional 1D-CS algorithms, 2D-CS algorithms are proposed to estimate delay-Doppler map in Eq. (5) by solving the following optimization problem:

$$\tilde{\boldsymbol{S}} = \min \left\| \boldsymbol{Y}_{sub} - \boldsymbol{\Phi}\tilde{\boldsymbol{S}}\boldsymbol{E}^H \right\|_F^2 + \lambda \left\| \tilde{\boldsymbol{S}} \right\|_0, \tag{9}$$

where $\|\boldsymbol{X}\|_F = \left(\sum_{i=1}^{N} \sum_{j=1}^{M} |x_{nm}|^2 \right)^{\frac{1}{2}}$ is the Frobenius norm (ℓ_F-norm). $\|\boldsymbol{S}\|_0$ is ℓ_0-norm which denotes the number of the non-zero elements in $\boldsymbol{S}$.

It is obvious that the ℓ_F-norm is to guarantee $\boldsymbol{Y}_{sub} = \boldsymbol{\Phi}\tilde{\boldsymbol{S}}\boldsymbol{E}^H$ (data-fitting term), while sparse constraint ℓ_0-norm is to guarantee the sparsity of the solution (sparse term). Let $\boldsymbol{G}(\tilde{\boldsymbol{S}}) = \lambda\|\tilde{\boldsymbol{S}}\|_0$, and ℓ_F-norm function $\boldsymbol{F}(\tilde{\boldsymbol{S}}) = \|\boldsymbol{Y}_{sub} - \boldsymbol{\Phi}\tilde{\boldsymbol{S}}\boldsymbol{E}^H\|_F^2$. Then Eq. (9) can be rewritten as $\min \boldsymbol{F}(\tilde{\boldsymbol{S}}) + \boldsymbol{G}(\tilde{\boldsymbol{S}})$ for simplicity of viewing. First, we focus on the problem of $\min \boldsymbol{F}(\tilde{\boldsymbol{S}})$. It can be solved by the projection method.

In projection method, the linear least-squares algorithm is used to find the least-squares solution. Utilizing 1D-CS algorithm, it should be transformed as Eq. (8). Then the solution is searched by: $\tilde{\boldsymbol{s}}(i+1) = \tilde{\boldsymbol{s}}(i) + \boldsymbol{\Phi}_v^H \left(\boldsymbol{y}_{sub} - \boldsymbol{\Phi}_v\tilde{\boldsymbol{s}}(i) \right)$, where $\tilde{\boldsymbol{s}}(i)$ denotes the solution in i-th iteration, and $\boldsymbol{y}_{sub} = vec\,(\boldsymbol{Y}_{sub})$. At last, each element in $\tilde{\boldsymbol{s}}(i)$ is rearranged as a delay-Doppler map. This process results in a large measurement matrix $\boldsymbol{\Phi}_v$ and it is not effective that the equation is transformed between 1D and 2D models. In our proposed algorithm, it is modified as follows:

$$\tilde{\boldsymbol{S}}(i+1) = \tilde{\boldsymbol{S}}(i) + \nabla \boldsymbol{F}(\tilde{\boldsymbol{S}}(i)) = \tilde{\boldsymbol{S}}(i) + \boldsymbol{\Phi}^H \boldsymbol{R}\boldsymbol{E}, \tag{10}$$

where $\boldsymbol{R} = \boldsymbol{Y}_{sub} - \boldsymbol{\Phi}\tilde{\boldsymbol{S}}(i)E^H$ denotes the residual at each iteration.

Theorem 1. Let $\tilde{\boldsymbol{S}}(i) = \tilde{\boldsymbol{S}}(i-1) + \boldsymbol{\Phi}^H\boldsymbol{R}\boldsymbol{E}$ and $\tilde{\boldsymbol{s}}(i) = \tilde{\boldsymbol{s}}(i-1) + \boldsymbol{\Phi}_v^H \left(\boldsymbol{y}_{sub} - \boldsymbol{\Phi}_v\tilde{\boldsymbol{s}}(i-1) \right)$. Then for any $i > 0$,

$$vec\left(\tilde{\boldsymbol{S}}(i)\right) = \tilde{\boldsymbol{s}}(i) \tag{11}$$

proof. Applying $vec(\cdot)$ on both sides of $\tilde{\boldsymbol{S}}(i) = \tilde{\boldsymbol{S}}(i-1) + \boldsymbol{\Phi}^H\boldsymbol{R}\boldsymbol{E}$, we have

$$\begin{aligned} vec\left(\tilde{\boldsymbol{S}}(i)\right) &= vec\left(\tilde{\boldsymbol{S}}(i-1) + \boldsymbol{\Phi}^H\boldsymbol{R}\boldsymbol{E}\right) \\ &= vec\left(\tilde{\boldsymbol{S}}(i-1)\right) + vec\left(\boldsymbol{\Phi}^H\boldsymbol{R}\boldsymbol{E}\right) \end{aligned} \tag{12}$$

According to [34], we have

$$\begin{aligned} vec\left(\boldsymbol{\Phi}^H\boldsymbol{R}\boldsymbol{E}\right) &= \left(\boldsymbol{E}^H \otimes \boldsymbol{\Phi}^H\right) vec\,(\boldsymbol{R}) \\ &= \boldsymbol{\Phi}_v^H vec\left(\boldsymbol{Y}_{sub} - \boldsymbol{\Phi}\tilde{\boldsymbol{S}}(i)\boldsymbol{E}^H\right) \\ &= \boldsymbol{\Phi}_v^H \left[vec\,(\boldsymbol{Y}_{sub}) - vec\left(\boldsymbol{\Phi}\tilde{\boldsymbol{S}}(i-1)\boldsymbol{E}^H\right)\right] \\ &= \boldsymbol{\Phi}_v^H \left[\boldsymbol{y}_{sub} - \boldsymbol{E} \otimes \boldsymbol{\Phi} vec\left(\tilde{\boldsymbol{S}}(i-1)\right)\right] \\ &= \boldsymbol{\Phi}_v^H \left[\boldsymbol{y}_{sub} - \boldsymbol{\Phi}_v vec\left(\tilde{\boldsymbol{S}}(i-1)\right)\right] \end{aligned} \tag{13}$$

Using Eq. (13) into Eq. (12), it follows that

$$\begin{aligned} &vec\left(\tilde{\boldsymbol{S}}(i)\right) \\ &= vec\left(\tilde{\boldsymbol{S}}(i-1)\right) + \boldsymbol{\Phi}_v^H \left[\boldsymbol{y}_{sub} - \boldsymbol{\Phi}_v vec\left(\tilde{\boldsymbol{S}}(i-1)\right)\right] \\ &= \left(1 - \boldsymbol{\Phi}_v^H\boldsymbol{\Phi}_v\right) vec\left(\tilde{\boldsymbol{S}}(i-1)\right) + \boldsymbol{\Phi}_v^H\boldsymbol{y}_{sub} \end{aligned} \tag{14}$$

Let $vec\left(\tilde{\boldsymbol{S}}(i)\right) = \boldsymbol{x}(i)$, $\left(1 - \boldsymbol{\Phi}_v^H\boldsymbol{\Phi}_v\right) = \boldsymbol{a}$, and $\boldsymbol{\Phi}_v^H\boldsymbol{y}_{sub} = \boldsymbol{b}$, Eq. (14) can be simplified as follows:

$$\boldsymbol{x}(i) = \boldsymbol{a}\boldsymbol{x}(i-1) + \boldsymbol{b}. \tag{15}$$

Then the difference equation can be rewritten as follows:

$$\boldsymbol{x}(i) = \boldsymbol{a}^{i+1}\boldsymbol{x}(0) + \boldsymbol{b}\sum_{n=0}^{i} \boldsymbol{a}^n \tag{16}$$

By the similar derivation, $\tilde{\boldsymbol{s}}(i) = \tilde{\boldsymbol{s}}(i-1) + \boldsymbol{\Phi}_v^H(\boldsymbol{y}_{sub} - \boldsymbol{\Phi}_v\tilde{\boldsymbol{s}}(i-1))$ can be simplified as follow:

$$\tilde{\boldsymbol{s}}(i) = \boldsymbol{a}^{i+1}\tilde{\boldsymbol{s}}(0) + \boldsymbol{b}\sum_{n=0}^{i} \boldsymbol{a}^n \tag{17}$$

Due to $\boldsymbol{x}(0) = \tilde{\boldsymbol{s}}(0)$, $vec\left(\tilde{\boldsymbol{S}}(i)\right) = \tilde{\boldsymbol{s}}(i)$ can be achieved.

Eq. (10) solves the ℓ_F-minimization. However, due to the lack of the sampled data, the sparse solution cannot be searched by the projection method directly. Hence, $\min \boldsymbol{G}(\tilde{\boldsymbol{S}})$ is used to attract the solution to be sparse. In this section, four methods to solve $\min \boldsymbol{G}(\tilde{\boldsymbol{S}})$ are proposed.

A. 2D ZERO-ATTRACTIVE PROJECTION (2D-ZAP) ALGORITHM

Inspired by the traditional 1D-ZAP algorithm, the 2D-ZAP algorithm is proposed to estimate delay-Doppler map by solving Eq. (9). Since ℓ_0-minimization problem is NP hard, ℓ_p-norm ($0 < p \leq 1$) is often used. It is well known that p is closer to 0, the algorithm is more effective. Thus, an approximate ℓ_0-norm is used in the 2D-ZAP algorithm, which can be written as follows:

$$\|\boldsymbol{X}\|_0 \approx \sum_{n=1}^{N}\sum_{m=1}^{M} \left(1 - e^{-\sigma|x_{nm}|}\right), \tag{18}$$

where the two sides of Eq. (18) is strictly equal when parameter $\sigma \to \infty$. Thus, Eq. (9) can be rewritten as follows:

$$\tilde{\boldsymbol{S}} = \min \underbrace{\left\| \boldsymbol{Y}_{sub} - \boldsymbol{\Phi}\tilde{\boldsymbol{S}}\boldsymbol{E}^H \right\|_F^2}_{\ell_F - norm} + \lambda \underbrace{\sum_{n=1}^{N}\sum_{m=1}^{M} \left(1 - e^{-\sigma|\tilde{s}_{nm}|}\right)}_{optimasation\ \ell_0 - norm}, \tag{19}$$

where $\tilde{s}_{nm}$ is the element in $\tilde{\boldsymbol{S}}$. Then $\boldsymbol{G}\left(\tilde{\boldsymbol{S}}\right)$ can be solved by gradient descent method.

In gradient descent method, the gradient (derivative) $\nabla \boldsymbol{G}\left(\tilde{\boldsymbol{S}}\right)$ of the function $\boldsymbol{G}(\tilde{\boldsymbol{S}})$ at the current point is calculated as follows:

$$\begin{aligned} g(\tilde{s}_{nm}) &= \nabla G\left(\tilde{s}_{nm}\right) \\ &= \frac{\partial 1 - e^{-\sigma|\tilde{s}_{nm}|}}{\partial \tilde{s}_{nm}} \\ &= \sigma \operatorname{sgn}(\tilde{s}_{nm}) e^{-\sigma|\tilde{s}_{nm}|}, \end{aligned} \tag{20}$$

where $\operatorname{sgn}(x)$ is a sign function defined as follows:

$$\operatorname{sgn}(x) = \begin{cases} 1, & x > 0; \\ -1, & x < 0; \\ 0, & x = 0. \end{cases} \tag{21}$$

To further reduce the complexity, the first two terms of Taylor series expansion of exponential function is used:

$$e^{-\sigma|\tilde{s}_{nm}|} = \sum_{q=0}^{\infty} \frac{(-\sigma\,|\tilde{s}_{nm}|)^q}{q!} \approx 1 - \sigma\,|\tilde{s}_{nm}|\,. \tag{22}$$

Then, a new point is updated along the negative of the gradient descent

$$\begin{aligned} g(\tilde{s}_{nm}) &= -\nabla G\left(\tilde{s}_{nm}\right) \\ &= \begin{cases} \sigma\left(\sigma \tilde{s}_{nm} + 1\right), & \frac{1}{\sigma} \le \tilde{s}_{nm} < 0; \\ \sigma\left(\sigma \tilde{s}_{nm} - 1\right), & 0 < \tilde{s}_{nm} \le \frac{1}{\sigma}; \\ 0, & otherwise. \end{cases} \end{aligned} \tag{23}$$

Finally, a local minimum of the function $\boldsymbol{F}(\tilde{\boldsymbol{S}})$ is achieved when the derivative of the function $\boldsymbol{F}(\tilde{\boldsymbol{S}}) = \left\|\boldsymbol{Y}_{sub} - \boldsymbol{\Phi}\tilde{\boldsymbol{S}}\boldsymbol{E}^H\right\|_F^2$ is close to 0.

Compared with the 1D-ZAP algorithm, the proposed 2D-ZAP algorithm reduces the memory requirement with the same estimation accuracy. Assuming that $\boldsymbol{\Phi} \in \mathbb{C}^{N_1 \times N_2}$ and $\boldsymbol{E} \in \mathbb{C}^{M_1 \times M_2}$. 1D-ZAP algorithm requires $N_1N_2 \times M_1M_2$ to store measurement matrix $\boldsymbol{\Phi}_v$, while 2D-ZAP algorithm only requires $N_1N_2 + M_1M_2$. In addition, in the 1D-ZAP algorithm, the complexity of the conjugate transpose $\boldsymbol{\Phi}_v$ is $\mathcal{O}\left(N_1N_2 \times M_1M_2\right)$, while in the 2D-ZAP algorithm, the conjugate transpose $\boldsymbol{E}$ and $\boldsymbol{\Phi}^H$ are required. Hence the complexity is only $\mathcal{O}\left(N_1N_2 + M_1M_2\right)$. The 2D-ZAP algorithm reduces the memory requirement for measurement matrix and the complexity of the conjugate transpose, which is valuable in the pulse Doppler radar system.

The proposed 2D-ZAP algorithm is listed in Algorithm 1.

Algorithm 1 2D Zero-Attractive Projection (2D-ZAP) Algorithm

Input: measurement matrices $\boldsymbol{A}, \boldsymbol{\Psi}, \boldsymbol{E}^T$ sampled signal $\boldsymbol{Y}_{sub}$, parameter λ, Maximum number of iterations i_{max},
Initialization: $\tilde{\boldsymbol{S}}^0 = \boldsymbol{0}, i = 0, \boldsymbol{R} = \boldsymbol{Y}_{sub}$,
Iteration:
$\tilde{\boldsymbol{S}}(i+1) \leftarrow \tilde{\boldsymbol{S}}(i) + \boldsymbol{\Phi}^H\boldsymbol{R}\boldsymbol{E}$,
$\tilde{\boldsymbol{S}}(i+1) \leftarrow \tilde{\boldsymbol{S}}(i+1) - g(\tilde{\boldsymbol{S}}(i+1))$,
$\boldsymbol{R} \leftarrow \boldsymbol{Y}_{sub} + \boldsymbol{\Phi}\tilde{\boldsymbol{S}}(i+1)\boldsymbol{E}^H$,
$i \leftarrow i + 1$,
Until $i = i_{max}$,
Output: estimated range Doppler map $\boldsymbol{S}^{\#} = \tilde{\boldsymbol{S}}(i)$.

B. 2D ITERATE HARD THRESHOLDING (2D-IHT) ALGORITHM

Inspired by the traditional 1D-IHT algorithm, the 2D-IHT algorithm is proposed to estimate delay-Doppler map by solving Eq. (9) as same as the 2D-ZAP algorithm. To further reduce the complexity of $\boldsymbol{G}(\tilde{\boldsymbol{S}})$, the above optimization problem can be solved as follows:

$$\tilde{\boldsymbol{S}}(i+1) = \mathbb{H}_\sigma\left[\tilde{\boldsymbol{S}}(i) + \boldsymbol{\Phi}^H\left(\boldsymbol{Y}_{sub} - \boldsymbol{\Phi}\tilde{\boldsymbol{S}}(i)\boldsymbol{E}^H\right)\boldsymbol{E}\right], \tag{24}$$

where $\mathbb{H}_\sigma\left[x\right]$ is the nonlinear operation that sets x as zero when x is smaller than the threshold value σ while keeps x when x is larger or equal to σ, which is shown as follows:

$$\mathbb{H}_\sigma\left[x\right] = \begin{cases} x, & x \ge \sigma; \\ 0, & x < \sigma. \end{cases} \tag{25}$$

Another well-known nonlinear operator $\mathbb{H}^K[\boldsymbol{X}]$ is that keeps K largest absolute elements in $\boldsymbol{X}$ and sets the other elements to zero. It can also be used when the target number K is known. Otherwise, a too large K will lead to the high false alarm probability while a too small K will lead to the low detection probability. However, it is difficult to know the target number before the detection. Hence it is not suitable for the pulse Doppler radar.

The 2D-IHT algorithm also reduces the memory requirement for the measurement matrix and the complexity of the conjugate transpose, which is similar to the 2D-ZAP algorithm.

The proposed 2D-IHT algorithm is listed in Algorithm 2.

Algorithm 2 2D Iterate Hard Thresholding (2D-IHT) Algorithm

Input: measurement matrices $\boldsymbol{A}, \boldsymbol{\Psi}, \boldsymbol{E}^T$ sampled signal $\boldsymbol{Y}_{sub}$, threshold value σ, Maximum number of iterations i_{max},
Initialization: $\tilde{\boldsymbol{S}}^0 = \boldsymbol{0}, i = 0, \boldsymbol{R} = \boldsymbol{Y}_{sub}$,
Iteration:
$\tilde{\boldsymbol{S}}(i+1) \leftarrow \tilde{\boldsymbol{S}}(i) + \boldsymbol{\Phi}^H\boldsymbol{R}\boldsymbol{E}$,
$\tilde{\boldsymbol{S}}(i+1) \leftarrow \mathbb{H}_\sigma\left[\tilde{\boldsymbol{S}}(i+1)\right]$,
$\boldsymbol{R} \leftarrow \boldsymbol{Y}_{sub} - \boldsymbol{\Phi}\tilde{\boldsymbol{S}}(i+1)\boldsymbol{E}^H$,
$i \leftarrow i + 1$,
Until $i = i_{max}$,
Output: estimated range Doppler map $\boldsymbol{S}^{\#} = \tilde{\boldsymbol{S}}(i)$.

C. 2D ITERATIVE SHRINKAGE-THRESHOLDING ALGORITHM (2D-ISTA)

Inspired by the traditional 1D-ISTA algorithm, the 2D-ISTA algorithm is proposed to estimate delay-Doppler map by solving Eq. (9) as same as the 2D-IHT algorithm. In ISTA,

$\mathbb{H}_\sigma[x]$ is replaced by the soft thresholding operator $\mathbb{S}_\sigma[x]$ which is defined as:

$$\mathbb{S}_\sigma[x] = \begin{cases} \text{sgn}(x) \cdot (|x| - \sigma), & |x| > \sigma; \\ 0, & |x| \le \sigma. \end{cases} \tag{26}$$

where σ is a small threshold value. Furthermore, the optimization problem can be solved as follows:

$$\tilde{\boldsymbol{S}}(i+1) = \mathbb{S}_\sigma\left[\tilde{\boldsymbol{S}}(i) + \frac{1}{L}\boldsymbol{\Phi}^H\left(\boldsymbol{Y}_{sub} - \boldsymbol{\Phi}\tilde{\boldsymbol{S}}(i)\boldsymbol{E}^H\right)\boldsymbol{E}\right], \tag{27}$$

where $\frac{1}{L}$ plays the role of a step-size to control the convergence rate. When $\frac{1}{L} \to 0$, Eq. (9) can be solved accurately, while the convergence rate is very slow. In contrast, if a too large $\frac{1}{L}$ is chosen, overshoot will occur. It is well known that $\frac{1}{L}$ depends on the eigenvalues of $\boldsymbol{\Phi}$ and $\boldsymbol{E}$. For the 1D-ISTA algorithm, $L = 2\lambda_{max}\left(\boldsymbol{\Phi}_v^H\boldsymbol{\Phi}_v\right)$, where $\lambda_{max}(\boldsymbol{X})$ denotes the max eigenvalues of $\boldsymbol{X}$. In the 2D-ISTA algorithm, $L = 2\lambda_{max}\left(\boldsymbol{\Phi}_v^H\boldsymbol{\Phi}_v\right)$ is replaced by $L = 2\lambda_{max}\left(\boldsymbol{E}^H\boldsymbol{E}\right) \times \lambda_{max}\left(\boldsymbol{\Phi}\boldsymbol{\Phi}^H\right)$. Thus, instead of the large matrix $\boldsymbol{\Phi}_v^H\boldsymbol{\Phi}_v$, the max eigenvalue of two small matrices $\boldsymbol{E}^H\boldsymbol{E}$ and $\boldsymbol{\Phi}\boldsymbol{\Phi}^H$ are calculated. As a result, the unnecessary complexity can be avoided.

The proposed 2D-ISTA is listed in Algorithm 3.

Algorithm 3 2D Iterative Shrinkage-Thresholding Algorithm (2D-ISTA)

Input: measurement matrices $\boldsymbol{A}, \boldsymbol{\Psi}, \boldsymbol{E}^T$ sampled signal $\boldsymbol{Y}_{sub}$, parameter σ, Maximum number of iterations i_{max}
Initialization: $\tilde{\boldsymbol{S}}^0 = \boldsymbol{0}$, $i = 0$, $\boldsymbol{R} = \boldsymbol{Y}_{sub}$, $L = 2\lambda_{max}\left(\boldsymbol{E}^H\boldsymbol{E}\right) \times \lambda_{max}\left(\boldsymbol{\Phi}\boldsymbol{\Phi}^H\right)$,
Iteration:
$\tilde{\boldsymbol{S}}(i+1) \leftarrow \tilde{\boldsymbol{S}}(i) + \frac{1}{L}\boldsymbol{\Phi}^H\boldsymbol{R}\boldsymbol{E}$,
$\tilde{\boldsymbol{S}}(i+1) \leftarrow \mathbb{S}_\sigma\left[\tilde{\boldsymbol{S}}(i+1)\right]$,
$\boldsymbol{R} \leftarrow \boldsymbol{Y}_{sub} - \boldsymbol{\Phi}\tilde{\boldsymbol{S}}(i+1)\boldsymbol{E}^H$,
$i \leftarrow i+1$,
Until $i = i_{max}$,
Output: estimated range Doppler map $\boldsymbol{S}^{\#} = \tilde{\boldsymbol{S}}(i)$.

D. 2D FAST ITERATIVE SHRINKAGE-THRESHOLDING ALGORITHM (2D-FISTA)

The ISTA algorithm has slow convergence in general. Therefore, some acceleration techniques have been proposed, such as an adaptive threshold value $\sigma(i)$, or an adaptive step-size $t(i)$ instead of $\frac{1}{L}$ [35], [36]. One of the most popular algorithms is the fast iterative shrinkage-thresholding algorithm (FISTA) [37]. Inspired by the 1D-FISTA algorithm, the 2D-FISTA algorithm is proposed in this paper to reduce the memory requirement and complexity. According to Theorem 1, the results of each step utilized the 2D-ISTA and 2D-FISTA algorithms are as the same as the 1D-ISTA and 1D-FISTA algorithms, respectively. Hence, similar to the relationship between the 1D-ISTA and 1D-FISTA algorithms, the 2D-ISTA and 2D-FISTA algorithms improve the worst-case complexity result of $\mathcal{O}\left(1/i^2\right)$ from 2D-ISTA complexity result of $\mathcal{O}(1/i)$.

The proposed 2D-FISTA is listed in Algorithm 4.

Algorithm 4 2D Fast Iterative Shrinkage-Thresholding Algorithm (2D-FISTA)

Input: measurement matrices $\boldsymbol{A}, \boldsymbol{\Psi}, \boldsymbol{E}^T$ sampled signal $\boldsymbol{Y}_{sub}$, parameter σ, Maximum number of iterations i_{max}
Initialization: $\tilde{\boldsymbol{S}}^0 = \boldsymbol{0}$, $i = 0$, $\boldsymbol{R} = \boldsymbol{Y}_{sub}$, $L = 2\lambda_{max}\left(\boldsymbol{E}^H\boldsymbol{E}\right) \times \lambda_{max}\left(\boldsymbol{\Phi}\boldsymbol{\Phi}^H\right)$, $t(1) = 1$.
Iteration:
$\tilde{\boldsymbol{S}}(i+1) \leftarrow \tilde{\boldsymbol{S}}(i) + \frac{1}{L}\boldsymbol{\Phi}^H\boldsymbol{R}\boldsymbol{E}$
$\tilde{\boldsymbol{S}}(i+1) \leftarrow \mathbb{S}_\sigma\left[\tilde{\boldsymbol{S}}(i+1)\right]$
$t(i+1) = \frac{1+\sqrt{1+4t^2(i)}}{2}$,
$\mu = \frac{t(i)-1}{t(i+1)}$,
$\tilde{\boldsymbol{S}}(i+1) \leftarrow \tilde{\boldsymbol{S}}(i+1) + \mu\left(\tilde{\boldsymbol{S}}(i+1) - \tilde{\boldsymbol{S}}(i)\right)$,
$\boldsymbol{R} \leftarrow \boldsymbol{Y}_{sub} - \boldsymbol{\Phi}\tilde{\boldsymbol{S}}(i+1)\boldsymbol{E}^H$,
$i \leftarrow i+1$,
Until $i = i_{max}$,
Output: estimated range Doppler map $\boldsymbol{S}^{\#} = \tilde{\boldsymbol{S}}(i)$.

IV. THE PROPOSED ALGORITHMS FOR NON-GAUSSIAN IMPULSIVE NOISE CASE

In the radar system, the signals are usually contaminated with non-Gaussian impulsive noise which can be described by the symmetric α-stable noise model [38]–[40]. The radar performance is easy to be degraded by non-Gaussian impulsive noise interference such as environmental effects of atmospherics (lighting) and meteor train echoes [29].

Recall the sampled signal model which is given by:

$$\boldsymbol{Y}_{sub} = \boldsymbol{\Phi}\boldsymbol{S}\boldsymbol{E}^H + \boldsymbol{W}, \tag{28}$$

where $w_l \in \boldsymbol{W}$ denotes the noise. In non-Gaussian impulsive noise case, w_l is a α-stable noise matrix with $w_l \sim \phi(\alpha, \beta, c, u)$. The characteristic function of α-stable distribution is defined as

$$r(l; \alpha, \beta, c, u) = \exp\{jul - |cl|^\alpha\left[1 - j\beta \cdot \text{sgn}(l)\varphi(l, \alpha)\right]\}, \tag{29}$$

where

1) $\alpha \in (0, 2]$ denotes the characteristic exponent. Only when $\alpha = 2$, the distribution becomes a Gaussian distribution;
2) $\beta \in [-1, 1]$ denotes the skewness parameter which controls the symmetry scenarios;
3) $c \in [0, \infty)$ is a scale factor which plays a similar role as the variance of Gaussian distribution;
4) $u \in (-\infty, \infty)$ is the location parameter. When $\beta = 0$, the distribution is symmetric about u;

and

$$\varphi(l,\alpha)=\begin{cases}\tan\left(\frac{\pi\alpha}{2}\right), & \alpha\neq 1;\\ -\frac{2}{\pi}\log|l|, & \alpha=1.\end{cases} \tag{30}$$

In this paper, the symmetric α-stable noise model (i.e., $\beta=0$ and $u=0$) is considered, since it usually exists in a practical radar system [38]. Hence, the characteristic function of the symmetric α-stable distribution is simplified as $r(l;\alpha,c)=\exp\{-|cl|^{\alpha}\}$. For convenience, the variance of the symmetric α-stable distribution is defined as $\sigma_n=c^{1/\alpha}$ which plays the role as the noise variance. The large variance results in that the approaches of the above 2D-CS algorithms are far from the desired original delay-Doppler map $\boldsymbol{S}$. Thus, robust 2D CS algorithms are expected.

The key idea in our proposed robust 2D CS algorithms is that ℓ_F-norm is replaced by more robust cost function, such as ℓ_1-norm, ℓ_p-norm and Lorentzian norm (LL_2-norm), which are presented in this section. These cost functions provide the bias and efficiency when the data is contaminated by a small noise and reliable albeit not optimal behavior when the data is contaminated by impulsive noise.

A. ℓ_1-BASED 2D METHOD

If ℓ_F-norm is replaced by ℓ_1-norm in the data-fitting term in Eq. (9), the solving method becomes:

$$\tilde{\boldsymbol{S}}=\min\left\|\boldsymbol{Y}_{sub}-\boldsymbol{\Phi}\tilde{\boldsymbol{S}}\boldsymbol{E}^{H}\right\|_1+\boldsymbol{G}(\tilde{\boldsymbol{S}}). \tag{31}$$

Then the data fitting term can be solved as:

$$\tilde{\boldsymbol{S}}(i+1)=\tilde{\boldsymbol{S}}(i)+\nabla\boldsymbol{F}(\tilde{\boldsymbol{S}})=\tilde{\boldsymbol{S}}(i)+\boldsymbol{\Phi}^{H}\boldsymbol{R}_{\ell_1}\boldsymbol{E}. \tag{32}$$

where $\boldsymbol{R}_{\ell_1}=\mathrm{sgn}(\boldsymbol{R})$, and it provides low variance gradient estimates, with the effect of reducing the residual error. It means that ℓ_1-based 2D method does not pay attention to the amplitude of the residual error, because in the non-Gaussian impulsive noise environment, it is very possible that the large residual error is caused by impulsive noise. If we pay the indiscriminate attention to the amplitude of the residual error, the approaches of the above 2D-CS algorithms are far from the desired original delay-Doppler map.

ℓ_1-based 2D-robust-ZAP (2D-RZAP (L1)) algorithm is extended from 2D-ZAP algorithm as follows:

$$\tilde{\boldsymbol{S}}(i+1)=\tilde{\boldsymbol{S}}(i)+\mu\boldsymbol{\Phi}^{H}\boldsymbol{R}_{\ell_1}\boldsymbol{E}+\lambda g(\tilde{\boldsymbol{S}}(i)), \tag{33}$$

and ℓ_1-based 2D-robust-IHT (2D-RIHT (L1)) algorithm is extended from 2D-IHT algorithm as follows:

$$\tilde{\boldsymbol{S}}(i+1)=\mathbb{H}_{\sigma}\left[\tilde{\boldsymbol{S}}(i)+\mu\boldsymbol{\Phi}^{H}\boldsymbol{R}_{\ell_1}\boldsymbol{E}\right], \tag{34}$$

where μ is the step-size which reduces the update rate of each element.

B. ℓ_p-BASED 2D METHOD

The ℓ_1-based 2D method is an easy method to limit the influences of the too large residual error, however, it still cannot distinguish that the data is contaminated by a small noise or a large impulsive noise. Thus, this method does not utilize the data which is contaminated by a small noise effectively. ℓ_p-based 2D method is proposed to reduce the influence of a large residual error while the small residual error is utilized effectively. It is given by:

$$\tilde{\boldsymbol{S}}=\min\left\|\boldsymbol{Y}_{sub}-\boldsymbol{\Phi}\tilde{\boldsymbol{S}}\boldsymbol{E}^{H}\right\|_p^p+\boldsymbol{G}(\tilde{\boldsymbol{S}}). \tag{35}$$

Then the data fitting term can be solved as:

$$\tilde{\boldsymbol{S}}(i+1)=\tilde{\boldsymbol{S}}(i)+\nabla\boldsymbol{F}(\tilde{\boldsymbol{S}})=\tilde{\boldsymbol{S}}(i)+\boldsymbol{\Phi}^{H}\boldsymbol{R}_{\ell_p}\boldsymbol{E}. \tag{36}$$

where $\boldsymbol{R}_{\ell_p}$ is the reweighted residual error matrix. The i-th element $\boldsymbol{R}_{\ell_p}(i)=\frac{\|\boldsymbol{R}\|_p^{1-p}\mathrm{sgn}(\boldsymbol{R}(i))}{\varepsilon+|\boldsymbol{R}(i)|^{1-p}}$, where $0<p<1$, and ε is a small constant bounding the term to avoid denominator becoming 0. It is a robust method since a large residual error does not influence the approach while a small residual error is utilized effectively.

ℓ_p-based 2D-robust-ZAP (2D-RZAP (Lp)) algorithm is extended from 2D-ZAP algorithm as follows:

$$\tilde{\boldsymbol{S}}(i+1)=\tilde{\boldsymbol{S}}(i)+\mu\boldsymbol{\Phi}^{H}\boldsymbol{R}_{\ell_p}\boldsymbol{E}+\lambda g(\tilde{\boldsymbol{S}}(i)), \tag{37}$$

and ℓ_p-based 2D-robust-IHT (2D-RIHT (Lp)) algorithm is extended from 2D-IHT algorithm as follows:

$$\tilde{\boldsymbol{S}}(i+1)=\mathbb{H}_{\sigma}\left[\tilde{\boldsymbol{S}}(i)+\mu\boldsymbol{\Phi}^{H}\boldsymbol{R}_{\ell_p}\boldsymbol{E}\right], \tag{38}$$

where μ is the step-size which reduces the update rate of each element.

C. LL_2-BASED 2D METHOD

The Lorentzian norm (LL_2-norm) is a robust norm that does not heavily penalized large deviations with the robustness depending on the scale parameter γ. The Lorentzian norm of a matrix $\boldsymbol{X}\in\mathbb{C}^{N\times M}$ is defined as:

$$\|\boldsymbol{X}\|_{LL_2,\gamma}=\sum_{n=1}^{N}\sum_{m=1}^{M}\log\left(1+\frac{x_{nm}^2}{\gamma^2}\right) \tag{39}$$

where setting γ to half the data range of $\boldsymbol{X}$ as $(\boldsymbol{X}_{(1)}-\boldsymbol{X}_{(0)})/2$ ($\boldsymbol{X}_{(p)}$ is the p-quantile of $vec(\boldsymbol{X})$) often makes the Lorentzian norm to approximate ℓ_2-norm. In general, γ is set as $(\boldsymbol{X}_{(0.875)}-\boldsymbol{X}_{(0.125)})/2$ (see [31]). It means that the measurement matrix with 25% of data corrupted and 75% well behaved by this value of γ.

Therefore, LL_2-based 2D method is proposed to reduce the influence of a large residual error while the large residual error is utilized effectively. It is given as:

$$\tilde{\boldsymbol{S}}=\min\left\|\boldsymbol{Y}_{sub}-\boldsymbol{\Phi}\tilde{\boldsymbol{S}}\boldsymbol{E}^{H}\right\|_{LL_2,\gamma}+\boldsymbol{G}(\tilde{\boldsymbol{S}}). \tag{40}$$

Then the data fitting term can be solved as:

$$\tilde{\boldsymbol{S}}(i+1)=\tilde{\boldsymbol{S}}(i)+\nabla\boldsymbol{F}(\tilde{\boldsymbol{S}})=\tilde{\boldsymbol{S}}(i)+\boldsymbol{\Phi}^{H}\boldsymbol{R}_{LL_2,\gamma}\boldsymbol{E}. \tag{41}$$

where $\boldsymbol{R}_{LL_2,\gamma}$ is the reweighted residual error matrix, and the i-th element $\boldsymbol{R}_{LL_2,\gamma}(i)=\frac{\gamma^2\boldsymbol{R}(i)}{\gamma^2+\boldsymbol{R}^2(i)}$.

LL_2-based 2D-robust-ZAP (2D-RZAP (LL2)) algorithm is extended from 2D-ZAP algorithm as follows:

$$\tilde{\boldsymbol{S}}(i+1) = \tilde{\boldsymbol{S}}(i) + \mu \boldsymbol{\Phi}^H \boldsymbol{R}_{LL_2,\gamma} \boldsymbol{E} + \lambda g(\tilde{\boldsymbol{S}}(i)), \tag{42}$$

and LL_2-based 2D-robust-IHT (2D-RIHT (LL2)) algorithm is extended from 2D-IHT algorithm as follows:

$$\tilde{\boldsymbol{S}}(i+1) = \mathbb{H}_\sigma \left[\tilde{\boldsymbol{S}}(i) + \mu \boldsymbol{\Phi}^H \boldsymbol{R}_{LL_2,\gamma} \boldsymbol{E} \right], \tag{43}$$

where μ is the step-size which reduces the update rate of each element.

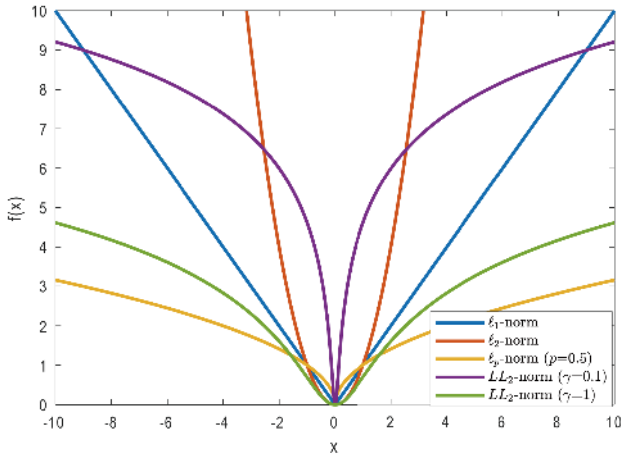

FIGURE 3. The cost function of ℓ_1-norm (blue), ℓ_p-norm with $p = 0.5$ (yellow), and the Lorentzian norm with $\gamma = 0.1$ (purple) and $\gamma = 1$ (green). ℓ_2-norm (red) is plotted as reference.

Fig. 3 shows the cost function of ℓ_1-norm (blue), ℓ_p-norm with $p = 0.5$ (yellow), and Lorentzian norm with $\gamma = 0.1$ (purple) and $\gamma = 1$ (green) and ℓ_2-norm (red) is plotted as reference. Compared with ℓ_2-norm, the functions of ℓ_1-norm, ℓ_p-norm and Lorentzian norm do not over penalize large deviations, which results in more robust for non-Gaussian impulsive noise. Moreover, ℓ_p-norm and Lorentzian norm are more robust to outliers than ℓ_1-norm because when $\boldsymbol{R}(i) \to \infty$, the formers do not increase their value as fast as the latter.

V. SIMULATION

The performance of the proposed algorithms are compared with those of the 1D-IHT [15], 1D-ZAP [16], 1D-ISTA [41], and 1D-FISTA [37] algorithms. Mean square error (MSE) is defined as:

$$MSE = \frac{1}{M \times N} \sum_{m=1}^{M} \sum_{n=1}^{N} \sqrt{\left(\tilde{s}_{n,p} - s_{n,p}\right)^2} \tag{44}$$

where $\tilde{s}_{n,p}$ is the estimation of $s_{n,p}$. A smaller MSE means a better estimation performance. The Bandwidth of transmitted signal B is set at 1.5kHz. The pulse width T_d is set as 1ms. The T_{PRI} is set as 2ms. The Nyquist sampling rate should be set as $2.25 \times B$ to void the aliasing. $P = 16$ pulses are transmitted to estimate the Doppler shift frequency. The delay resolution of a grid is $1/B$, and the Doppler resolution of

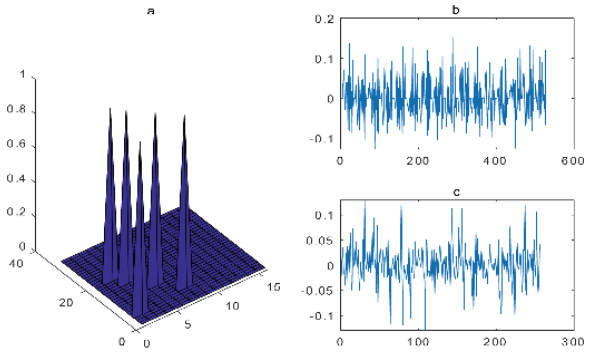

FIGURE 4. (a) The original delay-Doppler map; (b) The sampled data at Nyquist sampling rate; (c) The data sampled by the RD.

a grid is $2\pi/N$ [42]. Thus, the number of the delay grid is $M = \lfloor (T_{PRI} - T_d) * 2.25B \rfloor = 33$, while the number of the Doppler shift grids is $N = 16$. Operator $\lfloor \cdot \rfloor$ denotes the largest integer. It is shown in Fig. 4 (b) that the received signal should be sampled 528 times at Nyquist sampling rate. In the following experiments, the received signal is sampled by RD at half of the Nyquist sampling rate. Fig. 4 (c) shows that the received signal is only sampled 256 times. Thus, the matrix $\boldsymbol{\Phi} \in \mathbb{R}^{M' \times N}$, where $M' = \lfloor 0.5 \times M \rfloor = 19$ and $\boldsymbol{E} \in \mathbb{R}^{P \times P}$. According to 1D-CS algorithms, the requirement of the memory should be at least $\left(M' \times N\right) \times P^2 = 160512$, while in the 2D-CS algorithms, it reduces to $\left(M' \times N\right) + P^2 = 883$. In the first experiment, there are 5 existing targets in the search field with the white Gaussian noise whose signal-to-noise ratio (SNR) is set as 10dB. The original delay-Doppler map is shown in Fig. 4 (a).

From Fig. 5 to Fig. 8 show that the estimated delay-Doppler map by the 1D-ZAP ($\lambda = 0.01$), 2D-ZAP ($\lambda = 0.01$), 1D-IHT ($\sigma = 0.03$), 2D-IHT ($\sigma = 0.03$), 1D-ISTA ($\lambda = 0.01$), 2D-ISTA ($\lambda = 0.01$), 1D-FISTA ($\lambda = 0.01$) and 2D-FISTA ($\lambda = 0.01$) algorithms, respectively. One can find that 2D-CS algorithms can estimate the delay-Doppler map as the same as 1D-CS algorithms.

Table. 1 shows the calculating time of each algorithm. The CPU time is used as an index of complexity. The simulations

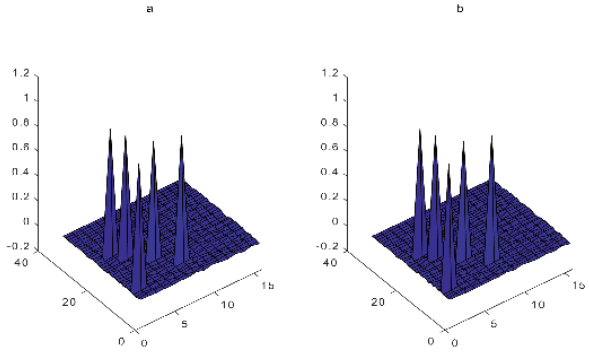

FIGURE 5. (a) The estimated delay-Doppler map by 1D-ZAP algorithm; (b) The estimated delay-Doppler map by 2D-ZAP algorithm.

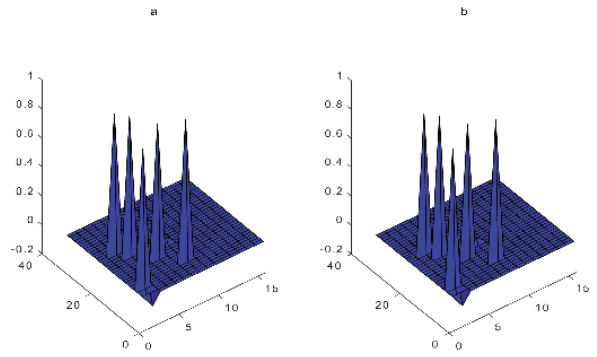

FIGURE 6. (a) The estimated delay-Doppler map by 1D-IHT algorithm; (b) The estimated delay-Doppler map by 2D-IHT algorithm.

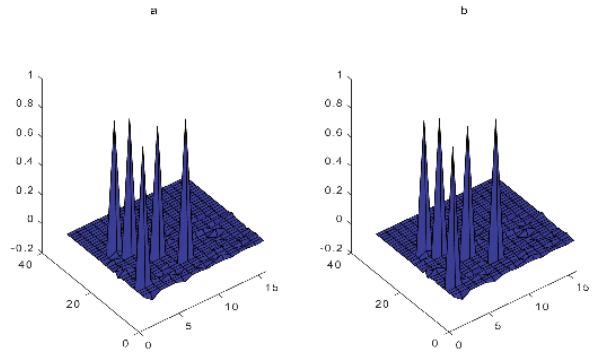

FIGURE 7. (a) The estimated delay-Doppler map by 1D-ISTA; (b) The estimated delay-Doppler map by 2D-ISTA.

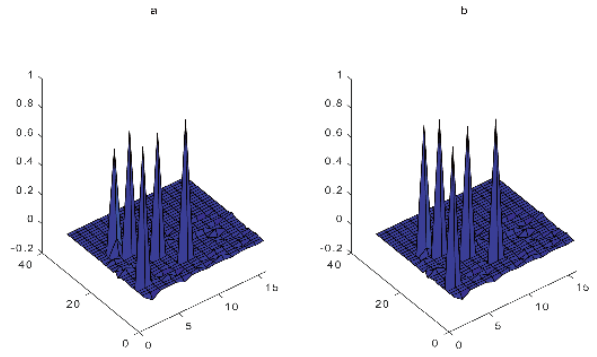

FIGURE 8. (a) The estimated delay-Doppler map by 1D-FISTA; (b) The estimated delay-Doppler map by 2D-FISTA.

are run in MATLAB R2017b with Intel Xeon E3-1270 v5, 3.60GHz processor and 16GB of memory under Microsoft Windows 10 Professional (64bit). The parameters of 2D-CS algorithms are set as the same as 1D-CS algorithms. The table shows that 2D-CS algorithms have a faster calculation time than 1D-CS algorithms.

Fig. 9 shows the MSE performances of the proposed 2D-ZAP algorithms against the parameter λ. The MSE performances of the 1D-ZAP algorithm are plotted as a reference. The parameter α is set as 1. It can be found that the 2D-ZAP algorithm has the same MSE performance as the 1D-ZAP algorithm with the same λ. In other words, utilizing

TABLE 1. The calculating time of each algorithm.

	ZAP	IHT	ISTA	FISTA
1D	1.2397	0.8883	5.1217	1.7720
2D	0.2103	0.1386	0.2091	0.1728

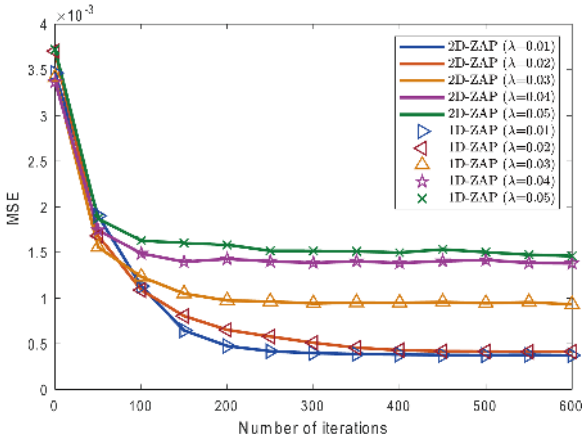

FIGURE 9. MSE curves of 1D-ZAP and 2D-ZAP algorithms with respect to λ.

the 2D-ZAP algorithm does not reduce the estimation performance. Moreover, for either 1D or 2D-ZAP algorithm, a smaller λ results in a smaller steady state MSE at a slower convergence rate, while a larger λ results in a larger steady state MSE at a faster convergence rate.

Fig. 10 shows the MSE performances of the proposed 2D-IHT algorithms against the parameter σ. The MSE performances of the 1D-IHT algorithm are plotted as a reference. It can be found that the 2D-IHT algorithm has the same MSE performance as the 1D-IHT algorithm with the same σ. In other words, utilizing the 2D-IHT algorithm does not reduce the estimation performance. Moreover, for either 1D or 2D-IHT algorithm, the MSE performance depends on the thresholding value σ. In this case, the least MSE can be

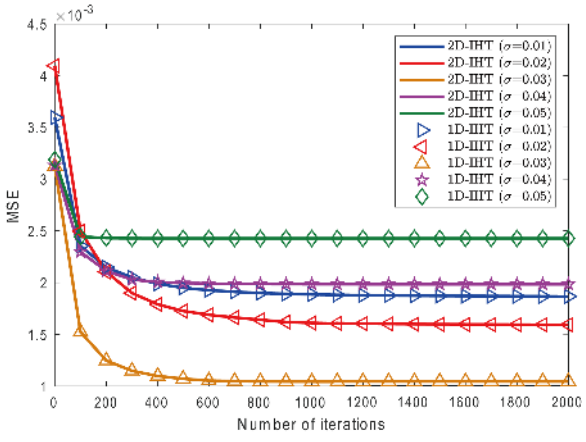

FIGURE 10. MSE curves of 1D-IHT and 2D-IHT algorithms with respect to σ.

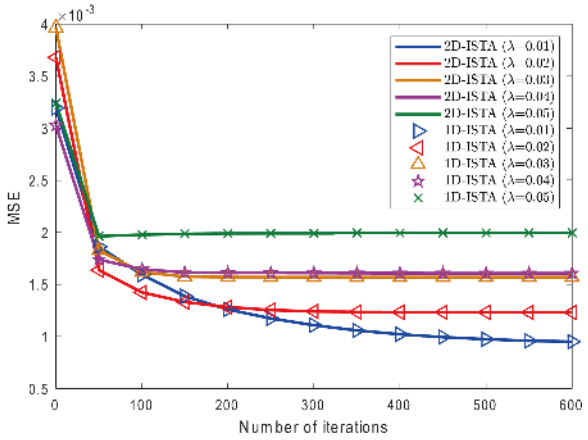

FIGURE 11. MSE curves of 1D-ISTA and 2D-ISTA algorithms with respect to λ.

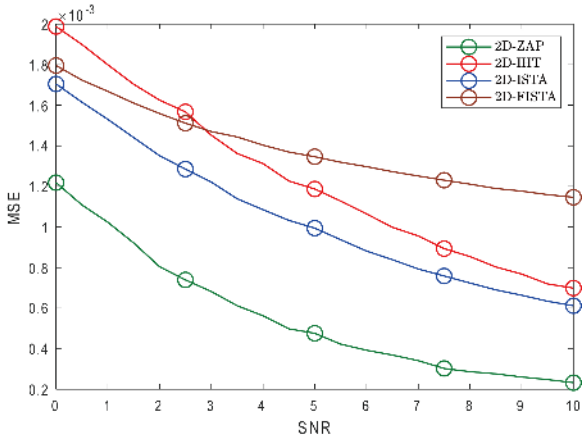

FIGURE 13. MSE curves of 2D-CS algorithms against SNR.

achieved when $\sigma = 0.03$. Therefore, finding a suitable σ is the key problem in the 2D-IHT algorithm. It still needs to be studied deeply.

Fig. 11 shows the MSE performances of the proposed 2D-ISTA algorithms against the parameter λ. The MSE performances of the 1D-ISTA algorithms with different λ are plotted as a reference. It can be found that the 2D-ISTA algorithm has the same MSE performance as the 1D-ISTA algorithm with the same λ. In other words, utilizing the 2D-ISTA algorithm does not reduce the estimation performance. Moreover, for either 1D or 2D-ISTA algorithm, a smaller λ results in a smaller steady state MSE and a slower convergence rate, while a larger λ results in a larger steady state MSE and a faster convergence rate. It is similar to 1D and 2D-ZAP algorithms.

Fig. 12 shows the MSE performances of the proposed 2D-FISTA algorithm against the parameter λ. The MSE performances of the 1D-FISTA algorithms with different λ are plotted as a reference. It can be found that the 2D-FISTA algorithm has the same MSE performance as the 1D-FISTA algorithm with the same λ. In other words, utilizing the 2D-FISTA algorithm does not reduce the estimation performance. Since the 2D-FISTA algorithm is an improved 2D-ISTA algorithm, the two kinds algorithms have the similar feature that a smaller λ results in a smaller steady state MSE and a slower convergence rate, while a larger λ results in a larger steady state MSE and a faster convergence rate. Moreover, compared with the 2D-ISTA algorithm, the number of iterations is much smaller when the 2D-FISTA algorithm gets the convergence.

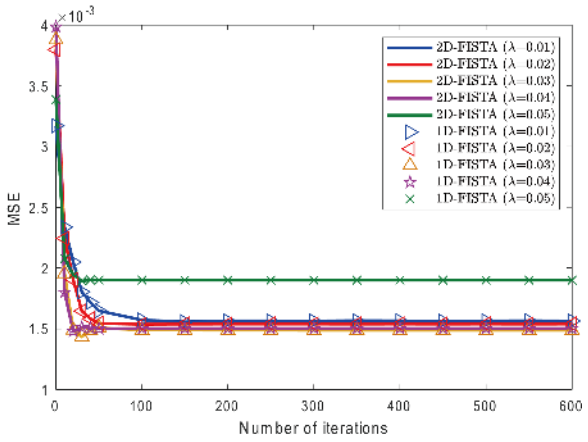

FIGURE 12. MSE curves of 1D-FISTA and 2D-FISTA algorithms with respect to λ.

In the third experiment, the robustness to the Gaussian noise is considered. There are 5 existing targets within the search field. SNR is chosen from 0 to 10dB. For each SNR, all these algorithms are repeated 1000 times to calculate MSEs. Other parameters are as the same as the first experiment. Fig. 13 shows that the MSEs of all algorithms decrease with the SNR increasing. Among these algorithms, the 2D-ZAP algorithm has the smallest MSE, and the 2D-ISTA algorithm has a smaller MSE than the 2D-FISTA algorithm, in each SNR environment. When SNR$<$ 3dB, the 2D-FISTA algorithm has a smaller MSE than the 2D-IHT algorithm, while the 2D-FISTA algorithm has a larger MSE than the 2D-IHT algorithm when SNR$>$ 3dB.

In the fourth experiment, the robustness to the target number is considered. There are 1-10 existing targets within the search field. SNR is chosen as10dB. For each target number, all these algorithms are repeated 1000 times to calculate MSEs. Other parameters are as the same as the first experiment. Fig. 14 shows that the MSEs of these algorithms increase as the number of targets increases. Among all algorithms, the 2D-ZAP algorithm has the smallest MSE. The 2D-ISTA and 2D-IHT algorithms have smaller MSEs than the 2D-FISTA algorithm, in each SNR environment. When target number$<$ 3, the 2D-IHT algorithm has smaller MSE than the 2D-ISTA algorithm, while the 2D-IHT algorithm has larger MSE than the 2D-ISTA algorithm when target number$>$ 3.

In the fifth experiment, the robustness to the compression ratio (CR) is considered. The CR is set from 0.2 to 1.

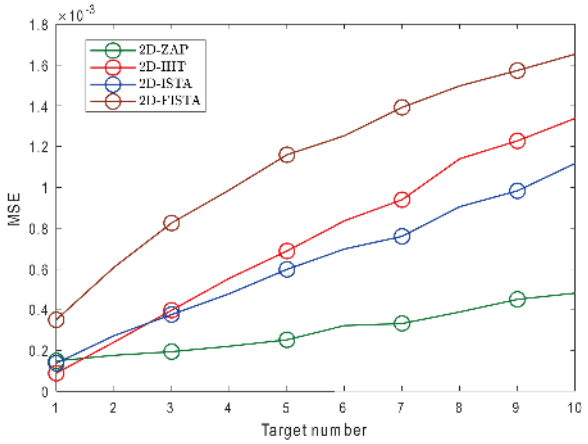

FIGURE 14. MSE curves of 2D-CS algorithms against the number of targets.

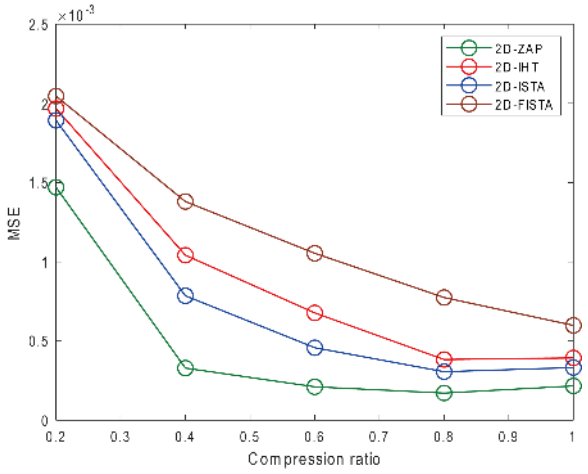

FIGURE 15. MSE curves of 2D-CS algorithms against compressive radio.

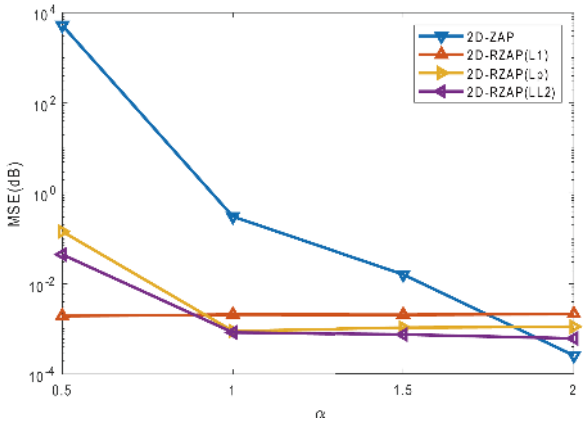

FIGURE 16. MSE curves of 2D-ZAP and 2D-RZAP algorithms with respect to α.

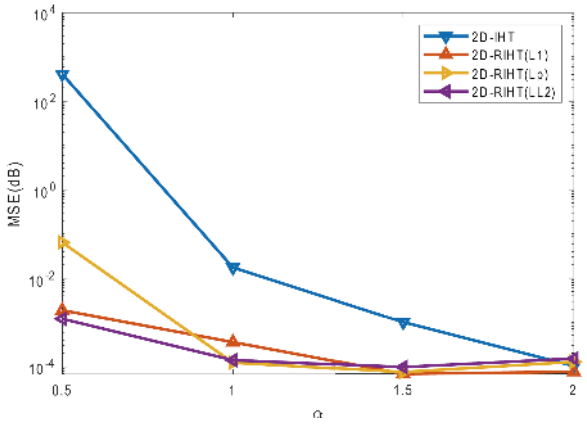

FIGURE 17. MSE curves of 2D-IHT and 2D-RIHT algorithms with respect to α.

CR = 1 means that the received signal is sampled at the Nyquist sampling rate. For each compression ratio, all these algorithms are repeated 1000 times to calculate MSEs. Other parameters are as the same as the first experiment. Fig. 15 shows that the MSEs of these algorithms reduce as the compression ratio increases. When compression ratio is larger than 0.4, the 2D-ZAP algorithm has a similar MSE to the original sampling rate.

In the sixth experiment, the robustness of the ZAP algorithms to the non-Gaussian noise is considered. The non-Gaussian impulsive noise satisfies the symmetric $\alpha-$stable distribution, where $\alpha \in [0.5, 1, 1.5, 2]$. When α is close to 2, the noise is similar to a Gaussian noise, while when α is close to 0, the noise is similar to a non-Gaussian impulsive noise. $\lambda = 0.01$, $\mu_{L0} = 0.01$, $\lambda_{L0} = 0.01$, $\mu_{Lp} = 0.0001$, $\lambda_{Lp} = 0.001$, $\mu_{LL2} = 0.5$, $\lambda_{LL2} = 0.0005$. In Fig. 16, one can find that the MSE of the 2D-ZAP algorithm becomes large in the non-Gaussian impulsive noise environment. When $\alpha = 0.5$, only the 2D-RZAP(L0) algorithm has a small MSE. When $1 \leq \alpha \leq 1.5$, the 2D-RZAP(L1) and 2D-RZAP(LL2) algorithms also have small MSE. And the MSE of these two algorithms is smaller than the 2D-RZAP(L0) algorithm. Only when $\alpha = 2$, namely in the Gaussian noise environment, the 2D-ZAP algorithm has the smallest MSE. The main reason is that the cost function of the 2D-ZAP algorithm is most sensitive to the noise, while the cost function of the 2D-RZAP(L0) algorithm is most non-sensitive to noise.

In the seventh experiment, the robustness of the IHT algorithms to the non-Gaussian noise is considered. $\sigma = 0.004$, $\mu_{L0} = 0.02$, $\sigma_{L0} = 0.1$, $\mu_{Lp} = 0.0002$, $\sigma_{Lp} = 0.1$, $\mu_{LL2} = 0.1$, $\lambda_{LL2} = 0.1$. In Fig. 17, as the same as the 2D-ZAP algorithm, the MSE of the 2D-IHT algorithm becomes large in the non-Gaussian impulsive noise environment. When $\alpha = 0.5$, the 2D-RIHT(Lp) and 2D-RIHT(LL2) algorithms have small MSE. When $1 \leq \alpha \leq 1.5$, the 2D-RIHT(L1), 2D-RIHT(Lp) and 2D-RIHT(LL2) algorithms also have small MSE. Only when $\alpha = 2$, namely in the Gaussian noise environment, the 2D-IHT algorithm has the smallest MSE. The main reason is that the cost function of the 2D-IHT algorithm is most sensitive to the noise, while the cost function of the 2D-RIHT(L0) algorithm is most non-sensitive to noise.

VI. CONCLUDING REMARKS

In this paper, we have proposed a 2D data model for the pulse Doppler radar system with the RD method. In this method, the data is under-sampled by a low rate ADC. Then the 2D-CS (i.e., 2D-ZAP, 2D-IHT, 2D-ISTA, and 2D-FISTA) algorithms have been proposed for detecting the sparse targets from the under-sampled data. Since the 2D-CS algorithms solve the 2D data model without vectorizing, the memory requirement and complexity are significantly reduced. Moreover, robust 2D-CS algorithms (2D-RZAP(L1, Lp, LL2) and 2D-RIHT(L1, Lp, LL2)) have also been given for non-Gaussian impulsive noise environment. Numerical simulations have been provided to validate the performances of our proposed algorithms. Motivated by artificial intelligence and its applications [43]–[49], we plan to introduce deep learning to develop intelligent sub-Nyquist pulse Doppler radar systems.

REFERENCES

[1] B. Moran, *Mathematics of Radar*. Dordrecht, The Netherlands: Springer, 2001, pp. 295–328.
[2] M. A. Hadi, S. Alshebeili, K. Jamil, and F. E. A. El-Samie, "Compressive sensing applied to radar systems: An overview," *Signal, Image Video Process.*, vol. 9, no. 1, pp. 25–39, 2015.
[3] J. H. G. Ender, "On compressive sensing applied to radar," *Signal Process.*, vol. 90, no. 5, pp. 1402–1414, May 2010.
[4] D. L. Donoho, "Compressed sensing," *IEEE Trans. Inf. Theory*, vol. 52, no. 4, pp. 1289–1306, Apr. 2006.
[5] E. Candès and J. Romberg, "Sparsity and incoherence in compressive sampling," *Inverse Problems*, vol. 23, no. 3, pp. 969–985, 2007.
[6] J. Romberg, "Compressive sensing by random convolution," *SIAM J. Imag. Sci.*, vol. 2, no. 4, pp. 1098–1128, Nov. 2009.
[7] J. A. Tropp, M. B. Wakin, M. F. Duarte, D. Baron, and R. G. Baraniuk, "Random filters for compressive sampling and reconstruction," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, vol. 3, May 2006, pp. IIIÂ872–IIIÂ875.
[8] J. A. Tropp, J. N. Laska, M. F. Duarte, J. K. Romberg, and R. G. Baraniuk, "Beyond Nyquist: Efficient sampling of sparse bandlimited signals," *IEEE Trans. Inf. Theory*, vol. 56, no. 1, pp. 520–544, Jan. 2010.
[9] M. Mishali and Y. C. Eldar, "From theory to practice: Sub-Nyquist sampling of sparse wideband analog signals," *IEEE J. Sel. Topics Signal Process.*, vol. 4, no. 2, pp. 375–391, Apr. 2010.
[10] M. A. Lexa, M. E. Davies, and J. S. Thompson, "Reconciling compressive sampling systems for spectrally sparse continuous-time signals," *IEEE Trans. Signal Process.*, vol. 60, no. 1, pp. 155–171, Jan. 2012.
[11] H. Chi *et al.*, "Microwave spectral analysis based on photonic compressive sampling with random demodulation," *Opt. Lett.*, vol. 37, no. 22, pp. 4636–4638, 2012.
[12] T. Ragheb, J. N. Laska, H. Nejati, S. Kirolos, R. G. Baraniuk, and Y. Massoud, "A prototype hardware for random demodulation based compressive analog-to-digital conversion," in *Proc. IEEE Int. Midwest Symp. Circuits Syst. (MWSCAS)*, Aug. 2008, pp. 37–40.
[13] W. Xu, Y. Cui, Y. Wang, S. Wang, and J. Lin, "A hardware implementation of random demodulation analog-to-information converter," *IEICE Electron. Express*, vol. 13, no. 16, p. 20160465, 2016.
[14] D. Needell and J. A. Tropp, "CoSaMP: Iterative signal recovery from incomplete and inaccurate samples," *Appl. Comput. Harmon. Anal.*, vol. 26, no. 3, pp. 301–321, 2009.
[15] T. Blumensath and M. E. Davies, "Iterative hard thresholding for compressed sensing," *Appl. Comput. Harmon. Anal.*, vol. 27, no. 3, pp. 265–274, Nov. 2009.
[16] J. Jin, Y. Gu, and S. Mei, "A stochastic gradient approach on compressive sensing signal reconstruction based on adaptive filtering framework," *IEEE J. Sel. Topics Signal Process.*, vol. 4, no. 2, pp. 409–420, Apr. 2010.
[17] Y. Li, Y. Lin, X. Cheng, Z. Xiao, F. Shu, and G. Gui, "Nonconvex penalized regularization for robust sparse recovery in the presence of $S\alpha S$ noise," *IEEE Access*, vol. 6, pp. 25474–25485, 2018.
[18] Y. Li, X. Cheng, and G. Gui, "Co-robust-ADMM-net: Joint ADMM framework and DNN for robust sparse composite regularization," *IEEE Access*, vol. 6, pp. 47943–47952, 2018.
[19] Y. Li *et al.*, "MUSAI-$L_{0.5}$: Multiple sub-wavelet-dictionaries-based adaptively-weighted iterative half thresholding algorithm for compressive imaging," *IEEE Access*, vol. 6, pp. 16795–16805, 2018.
[20] L. Xiao, X.-G. Xia, and W. Wang, "Multi-stage robust Chinese remainder theorem," *IEEE Trans. Signal Process.*, vol. 62, no. 18, pp. 4772–4785, Sep. 2014.
[21] L. Xiao and X.-G. Xia, "Frequency determination from truly sub-Nyquist samplers based on robust Chinese remainder theorem," *Signal Process.*, vol. 150, pp. 248–258, Sep. 2018.
[22] J. Xu, Z.-Z. Huang, Z.-R. Wang, L. Xiao, X.-G. Xia, and T. Long, "Radial velocity retrieval for multichannel SAR moving targets with time–space Doppler deambiguity," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 1, pp. 35–48, Jan. 2018.
[23] M. A. Herman and T. Strohmer, "High-resolution radar via compressed sensing," *IEEE Trans. Signal Process.*, vol. 57, no. 6, pp. 2275–2284, Feb. 2009.
[24] G. Shi, J. Lin, X. Chen, F. Qi, D. Liu, and L. Zhang, "UWB echo signal detection with ultra-low rate sampling based on compressed sensing," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 55, no. 4, pp. 379–383, Apr. 2008.
[25] G. E. Smith, T. Diethe, Z. Hussain, J. Shawe-Taylor, and D. R. Hardoon, "Compressed sampling for pulse Doppler radar," in *Proc. IEEE Radar Conf.*, May 2010, pp. 887–892.
[26] R. A. Sevimli, M. Tofighi, and A. E. Cetin, "Range-Doppler radar target detection using denoising within the compressive sensing framework," in *Proc. Eur. Signal Process. Conf. (EUSIPCO)*, Sep. 2014, pp. 1950–1954.
[27] Y. Fang, J. Wu, and B. Huang, "2D sparse signal recovery via 2D orthogonal matching pursuit," *Sci. China Inf. Sci.*, vol. 55, no. 4, pp. 889–897, 2012.
[28] B. Liu, G. Gui, S.-Y. Matsushita, and L. Xu, "Sparse target detection of pulse Doppler radar based on two dimensional iterative hard thresholding algorithm," in *Proc. Chin. Control Decis. Conf. (CCDC)*, May 2017, pp. 4856–4860, doi: 10.1109/CCDC.2017.7979354.
[29] D. G. Khairnar, S. N. Merchant, and U. B. Desai, "Radar signal detection in non-Gaussian noise using RBF neural network," *J. Comput.*, vol. 3, no. 1, pp. 32–39, 2008.
[30] R. E. Carrillo, K. E. Barner, and T. C. Aysal, "Robust sampling and reconstruction methods for sparse signals in the presence of impulsive noise," *IEEE J. Sel. Topics Signal Process.*, vol. 4, no. 2, pp. 392–408, Apr. 2010.
[31] R. E. Carrillo and K. E. Barner, "Lorentzian iterative hard thresholding: Robust compressed sensing with prior information," *IEEE Trans. Signal Process.*, vol. 61, no. 19, pp. 4822–4833, Oct. 2013.
[32] W. U. Bajwa, K. Gedalyahu, and Y. C. Eldar, "Identification of parametric underspread linear systems and super-resolution radar," *IEEE Trans. Signal Process.*, vol. 59, no. 6, pp. 2548–2561, Jun. 2011.
[33] N. Levanon, *Radar Principles*. New York, NY, USA: Wiley, 1988.
[34] K. B. Petersen *et al.*, "The matrix cookbook," Techn. Univ. Denmark, Lyngby, Denmark, Tech. Rep., 2008, p. 15, vol. 7.
[35] S. Dahlke, M. Fornasier, and T. Raasch, "Multilevel preconditioning and adaptive sparse solution of inverse problems," *Math. Comput.*, vol. 81, no. 277, pp. 419–446, 2012.
[36] E. van den Berg and M. P. Friedlander, "Probing the Pareto frontier for basis pursuit solutions," *SIAM J. Sci. Comput.*, vol. 31, no. 2, pp. 890–912, 2008.
[37] A. Beck and M. Teboulle, "A fast iterative shrinkage-thresholding algorithm for linear inverse problems," *SIAM J. Imag. Sci.*, vol. 2, no. 1, pp. 183–202, 2009.
[38] M. Shao and C. L. Nikias, "Signal processing with fractional lower order moments: Stable processes and their applications," *Proc. IEEE*, vol. 81, no. 7, pp. 986–1010, Jul. 1993.
[39] D. Middleton, "Non-Gaussian noise models in signal processing for telecommunications: New methods an results for class A and class B noise models," *IEEE Trans. Inf. Theory*, vol. 45, no. 4, pp. 1129–1149, May 1999.
[40] G. Gui, L. Xu, W. Ma, and B. Chen, "Robust adaptive sparse channel estimation in the presence of impulsive noises," in *Proc. IEEE Int. Conf. Digit. Signal Process. (DSP)*, Jul. 2015, pp. 628–632.
[41] P. L. Combettes and J.-C. Pesquet, "Proximal splitting methods in signal processing," in *Fixed-Point Algorithms for Inverse Problems in Science and Engineering*. 2011, pp. 185–212.

[42] B. R. Mahafza, *Radar Systems Analysis and Design Using MATLAB*. London, U.K.: Chapman & Hall, 2005.
[43] G. Gui, H. Huang, Y. Song, and H. Sari, "Deep learning for an effective nonorthogonal multiple access scheme," *IEEE Trans. Veh. Technol.*, vol. 67, no. 9, pp. 8440–8450, Sep. 2018.
[44] H. Huang, J. Yang, H. Huang, Y. Song, and G. Gui, "Deep learning for super-resolution channel estimation and DOA estimation based massive MIMO system," *IEEE Trans. Veh. Technol.*, vol. 67, no. 9, pp. 8549–8560, Sep. 2018.
[45] Y. Tu, Y. Lin, J. Wang, and J. U. Kim, "Semi-supervised learning with generative adversarial networks on digital signal modulation classification," *Comput. Mater. Continua*, vol. 55, no. 2, pp. 243–254, 2018.
[46] M. Liu, T. Song, and G. Gui, "Deep cognitive perspective: Resource allocation for NOMA based heterogeneous IoT with imperfect SIC," *IEEE Internet Things J.*, to be published, doi: 10.1109/JIOT.2018.2876152.
[47] M. Liu, J. Yang, T. Song, J. Hu, and G. Gui, "Deep learning-inspired message passing algorithm for efficient resource allocation in cognitive radio networks," *IEEE Trans. Veh. Technol.*, to be published, doi: 10.1109/TVT.2018.2883669.
[48] J. Pan, Y. Yin, J. Xiong, W. Luo, G. Gui, and H. Sari, "Deep learning-based unmanned surveillance systems for observing water levels," *IEEE Access*, to be published, doi: 10.1109/ACCESS.2018.2883702.
[49] X. Sun, G. Gui, Y. Li, R. P. Liu, and Y. An, "ResInNet: A novel deep neural network with feature re-use for Internet of Things," *IEEE Internet Things J.*, to be published, doi: 10.1109/JIOT.2018.2853663.

Authors' photographs and biographies not available at the time of publication.