Adaptive Beamforming With Compressed Sensing for Sparse Receiving Array

JIAN WANG
WEI-XING SHENG
YU-BING HAN
XIAO-FENG MA
Nanjing University of Science and Technology
China

An adaptive digital beamforming technique with compressed sensing (CS) for sparse receiving arrays is proposed. Because of the angle sparseness of arriving signals, CS theory can be adopted to sample receiving signals. Then, receiving signals from absent elements on the antenna aperture can be reconstructed by using CS method. Adaptive digital beamforming algorithms are adopted to form antenna beam, whose main lobe is steered to the desired direction and nulls steered to the directions of interference.

Manuscript received August 30, 2012; revised April 28, 2013; released for publication September 20, 2013.

DOI. No. 10.1109/TAES.2014.120532.

Refereeing of this contribution was handled by M. Weiss.

This work is supported by the National Natural Science Foundation of China (Grant No. 11273017) and the Shanghai Aerospace Science and Technology Innovation Fund of China (Grant No. SAST201356).

Authors' addresses: J. Wang, W.-X. Sheng, Y.-B. Han, X.-F Ma, Nanjing University of Science and Technology, Communication Engineering Department, 200, Xiao Ling Wei, Nanjing, Jiangsu Province 210094, China, E-mail: (shengwx@njust.edu.cn).

I. INTRODUCTION

In bistatic radar systems, digital beamforming receiving array antennas are urgently needed so that the receiving antenna beams can cover the transmitting antenna beam flexibly. On receiving radar stations, to obtain high antenna gain and high angular measurement accuracy, an antenna array with a large number of antenna elements should be used. In radio astronomy systems, there is also a serious need for receiving antenna arrays with a large number of antenna elements. On the other hand, for a large-scale adaptive array, the high cost is a major drawback. Meanwhile, the computational burden and high-rate data transmission are two bottlenecks in the implementation of an adaptive beamforming algorithm. To reduce the number of radio frequency (RF) frontends without reducing the antenna aperture, we can use sparse arrays. Sparse arrays are attractive because they can reduce number of elements as compared with a fully populated array. This is of particular interest in radar applications because a large aperture facilitates high performance with regard to angular accuracy, resolution, and detection of targets close to interference directions. By employing sparse arrays, this can be achieved by reduced receiving channels, weight, power consumption, and cost as well as easier platform integration. The sparse equally spaced arrays produce grating lobes inevitably and reduce the scanning range of the beam. To avoid grating lobes, sparse arrays are usually designed to be unequally spaced. Unequally spaced arrays, or aperiodic arrays, have been studied for several decades. Researchers have found many effective methods to solve the problem of their inherently high sidelobes, which may degrade performance in an interference environment.

Generally the reduction of the number of antenna elements calls for the design of nonuniformly spaced arrays. Many techniques have been proposed over the last 50 years to synthesize such arrays. The most common is stochastic methods, such as genetic algorithm [1, 2], particle swarm [3], ant colony [4], and simulated annealing [5]. Recently, matrix pencil methods have been efficiently applied to reconstruct focused and shaped beam patterns, while reducing the number of the array elements [6]. Differential evolution [7] and sparse-periodic hybrid array [8] are also good methods for reducing the sidelobe levels. In [9, 10], a procedure to synthesize sparse arrays with antenna selected via convex optimization is presented. In [11, 12], a method of pattern synthesis with sparse arrays based on Bayesian compressive sampling is proposed. However, these methods only optimize static patterns and are difficult to guarantee the performance of the beam patterns when we need to suppress interference adaptively.

Recently, Candés and Donoho reported a novel sampling theory called compressed sensing (CS), also known as compressive sampling [13, 14]. The theory is a newly developed theoretical framework for information acquisition and processing, which is based on matrix analysis, statistical probability theory, topological

geometry, optimization, functional analysis, and so on. CS theory asserts that one can recover certain signals and images from far fewer samples or measurements than traditional methods use. To make this possible, CS relies on one principle: sparsity, which pertains to the signals of interest. The theory states that as long as the signal is sparse or compressible, you can reconstruct the original signal from a small number of samples by solving an optimization problem with high probability. The theory has been widely used in astronomy [15], data acquisition, direction-of-arrival estimation [16, 17], cognitive radios [18], radar [19, 20], optical imaging [21], and many other fields. Some researchers are also interested in hardware implementation, and some simple hardware structures have been achieved in [22, 23].

An adaptive digital beamforming technique with CS for large-scale, sparse receiving array is proposed in this paper. Because of the angle sparseness of the arriving signal, CS theory can be adopted to sample receiving signals with a sparse array antenna. Then, receiving signals from absent elements on the antenna aperture can be reconstructed by using the orthogonal matching pursuit (OMP) method [24, 25]. Adaptive digital beamforming algorithms are then adopted to form antenna beams, whose main lobe is steered to the desired direction and nulls are steered to the directions of interference. With the proposed adaptive digital beamforming technique, it greatly reduces the number of elements; also the beam performance is similar with full arrays, which means that the beams have low sidelobes and deep nulls in the directions of interference and without grating lobes. Simulation results with the Monte Carlo method show that the beam performances of our proposed method are approaching that of full array antenna. Actual antenna elements can be reduced greatly.

The paper is organized as follows. In the next section, the principle and mathematical model of digital beamforming based on CS are discussed. Then, in Section III, a method for optimizing the elements' positions is proposed. In Section IV, when targets are not on the grid, a method for improving the results is proposed. The simulation results under different situations are provided to prove the correctness of the algorithm in Section V. Finally, conclusions are given in Section VI.

II. THE PRINCIPLE OF THE ALGORITHM

A. Signal Model

We now consider a linear array antenna with N elements. The distance between neighboring elements is $\lambda/2$, which is half of the radar's wavelength. Assume that K echo signals incident to the antenna array, of which the complex amplitude is $s_k(t)$ and the incident angle is θ_{sk}, with $k = 1, 2, \ldots, K$. One of these signals is a desired signal, and the others are interference. The received signal can be expressed as the N-dimensional vector $\mathbf{X}(t) = [x_1(t), x_2(t), \ldots, x_N(t)]^{\mathrm{T}}$. Regardless of the receiver noise, it can be written as

$$\mathbf{X}(t) = \sum_{k=1}^{K} s_k(t)\,\mathbf{a}(\theta_{sk}), \tag{1}$$

where $\mathbf{a}(\theta_{sk})$ represents the directional vector of θ_{sk} with $k = 1,2,\ldots,K$.

$$\mathbf{a}(\theta_{sk}) = \left(1, e^{j2\pi d \sin(\theta_{sk})/\lambda}, e^{j2\pi 2d \sin(\theta_{sk})/\lambda}, \cdots, e^{j2\pi(N-1)d \sin(\theta_{sk})/\lambda}\right)^{T}. \tag{2}$$

We decompose the entire space from -90° to 90° into Γ parts with $|\sin(\theta_i) - \sin(\theta_{i+1})| = 2/\Gamma$, then we obtain θ_i, with $i = 1, 2, \ldots, \Gamma$. So, we can define a transformation matrix $\mathbf{H}$ as

$$\mathbf{H} = [\mathbf{a}(\theta_1), \mathbf{a}(\theta_2), \cdots, \mathbf{a}(\theta_\Gamma)]. \tag{3}$$

Then, the received signal $\mathbf{X}(t)$ is written as

$$\mathbf{X}(t) = \mathbf{H}\mathbf{S}(t), \tag{4}$$

where $\mathbf{S}(t) = [0, 0, \ldots, s_1(t), 0, \ldots, 0, \ldots, s_K(t), 0, \ldots, 0]^{\mathrm{T}}$. It is clear that $\mathbf{S}(t)$ is sparse and has few nonzero elements. Therefore, as the CS theory indicated, the received signal $\mathbf{X}(t)$ can be recovered accurately by using a CS reconstruction algorithm.

While considering the receiver noise, (4) can be rewritten as

$$\mathbf{X}(t) = \mathbf{H}\mathbf{S}(t) + \mathbf{V}(t), \tag{5}$$

where $\mathbf{V}(t) = [v_1(t), v_2(t), \ldots, v_N(t)]^{\mathrm{T}}$ is a Gaussian white noise vector.

B. Compressed Sampling and Reconstruction

We now design an $M \times N$ dimension measurement matrix $\mathbf{\Phi}$ with $M \ll N$, which is uncorrelated with the transformation matrix $\mathbf{H}$. Then, the compressed vector $\mathbf{Z}$ can be obtained by

$$\begin{aligned}\mathbf{Z}(t) &= \mathbf{\Phi}X(t)\\ &= \mathbf{\Phi}[\mathbf{H}\mathbf{S}(t) + \mathbf{V}(t)]\\ &= \mathbf{P}\mathbf{S}(t) + \mathbf{V}'(t)\end{aligned} \tag{6}$$

where $\mathbf{V}'(t) = \mathbf{\Phi}\mathbf{V}(t)$ is the noise of the system after compressed sampling. The measurement matrix $\mathbf{\Phi}$ means a method of compressed sampling space signals, which can be achieved by randomly selecting M rows from an identity matrix of size $N \times N$. That is to say, we can select M elements from the array to sample the space signals.

The matrix $\mathbf{P} = \mathbf{\Phi}\mathbf{H}$ of size $M \times \Gamma$ in (6) is called the observation matrix. Theoretical studies in [13, 14, 17, 26] have shown that we can accurately reconstruct the receiver signal $\mathbf{X}(t)$ from the compressed vector $\mathbf{Z}(t)$ if the observation matrix $\mathbf{P}$ satisfies the condition of restricted isometry property (RIP) [26]. Therefore, the design of the measurement matrix $\mathbf{\Phi}$ is very important. There are many measurement matrixes that have been used, such as the Hadamard matrix, Gaussian random matrix, sparse random matrix, part of the Fourier matrix, and so on. Here, $\mathbf{P} = \mathbf{\Phi}\mathbf{H}$ has been proved to satisfy the RIP condition.

Here, we randomly selected M elements from the N elements. The output of the M elements can be written as $\mathbf{Z}(t) = [z_1(t), z_2(t), \ldots, z_M(t)]^{\mathrm{T}}$. We discretize the time-domain signals. Then, we have L snapshots of the echo signal in the time domain. We rewrite (6) as

$$\mathbf{Z}_{M\times L} = \mathbf{\Phi}_{M\times N}\,\mathbf{X}_{N\times L} = \mathbf{P}_{M\times \Gamma}\,\mathbf{S}_{\Gamma\times L} + \mathbf{V}'_{M\times L}. \quad (7)$$

After achieving the compressed samples of the M elements $\mathbf{Z}_{M\times L}$, we can estimate the projection coefficient vector $\mathbf{S}_{\Gamma\times L}$ by using the OMP algorithm, which is discussed in [24, 25]. Then, the receiver signal vector $\mathbf{X}'_{N\times L}$ can be reconstructed as

$$\mathbf{X}'_{N\times L} = \mathbf{H}_{N\times \Gamma}\,\mathbf{S}_{\Gamma\times L}. \quad (8)$$

Details of the OMP algorithm:

1) Initialize the residual $\mathbf{r}_0 = \mathbf{y}$, the number of iterations $n = 1$, and the index set $\mathbf{\Lambda} = \emptyset$, $\mathbf{J} = \emptyset$;

2) Compute the correlation coefficient vector $\mathbf{u} = [u_1, u_2, \ldots, u_\Gamma]$ by

$$u_j = \left\|\mathbf{P}_j^H \mathbf{r}\right\|_2,\ j = 1, 2, \ldots, \Gamma \quad (9)$$

and put the number of the maximum value of $\mathbf{u}$ into the matrix $\mathbf{J}$;

3) Update the support matrix $\mathbf{P}_{\mathbf{\Lambda}}$ with $\mathbf{\Lambda} = \mathbf{\Lambda}\cup\mathbf{J}$;

4) Compute $\hat{\boldsymbol{S}}$ by

$$\hat{\boldsymbol{S}} = \arg\min_{\hat{\boldsymbol{S}}} \left\|\mathbf{Z} - \mathbf{P}_{\Lambda}\hat{\boldsymbol{S}}\right\|_2 \quad (10)$$

and update the residual by

$$\mathbf{r}_{new} = \mathbf{Z} - \mathbf{P}_{\Lambda}\hat{\mathbf{S}} \quad (11)$$

5) If $\|\mathbf{r}_{new} - \mathbf{r}\| \geq \varepsilon$, then $\mathbf{r} = \mathbf{r}_{new}$ and $n = n + 1$, jump to step (2). Otherwise, stop.

C. Adaptive Digital Beamforming

After reconstructing the echo signals by the reconstruction algorithm, we use adaptive digital beamforming algorithms to receive the desired signal and suppress interference with the recovered data. There are two main kinds of adaptive digital beamforming algorithms: one is the orthogonal projection algorithm, and the other is the linearly constrained minimum variance (LCMV) beamforming algorithm based on iteration.

1) *Orthogonal Projection Algorithm*: The adaptive digital beamforming algorithm, based on orthogonal projection, has been widely used. It achieves the optimal weights by projecting the desired signal onto the complementary space of the interference. What we need here is a covariance matrix $\mathbf{R}_{II}$ of received signal $\mathbf{X}_I(t)$, which contains the interference and noise only. For most pulse radars, the pulse repetition interval T_{PRF} is usually designed larger than the time delay $T_{\max}$ of the echoes from targets in maximum operating range. So, the returns in period $[T_{\max}, T_{\mathrm{PRF}}]$ are mainly composed of interference, and desired signals from targets can be ignored. Sampling the array received signal in period $[T_{\max}, T_{\mathrm{PRF}}]$, we obtain the received signal $\mathbf{X}_I(t)$, which contains the interference and noise only.

By using the CS reconstruction algorithm, the received signal $\mathbf{X}'_I(t)$ of the full array can be calculated. Then, the covariance matrix $\mathbf{R}_{II}$ can be obtained as

$$\mathbf{R}_{II} = \frac{1}{L}\sum_{l=1}^{L} \mathbf{X}'_I(l)\,\mathbf{X}'^H_I(l), \quad (12)$$

where L is the number of snapshots. Then, by decomposing the matrix $\mathbf{R}_{II}$, we obtain the eigenvalues $\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_{K-1} \geq \lambda_K = \ldots = \lambda_N$ and corresponding eigenvectors $\mathbf{v}_n$ with $n = 1, 2 \ldots N$. Usually, the powers of the interference are much larger than noise. So, the eigenvalues $\lambda_1, \lambda_2, \ldots, \lambda_{K-1}$ are related to interference, and $\lambda_K, \lambda_{K+1}, \ldots, \lambda_N$ are related to noise. The weight vector of the orthogonal projection algorithm can be obtained by [27]

$$\mathbf{w} = \left(\mathbf{I}_K - \sum_{i=1}^{K-1} \mathbf{v}_i \mathbf{v}_i^H\right)\mathbf{a}(\theta_s). \quad (13)$$

2) *Iterative LCMV Beamforming Algorithm*: The LCMV algorithm minimizes the power of the output signal and obtains the weight $\mathbf{w}$ by solving the linear constraint equation [28]

$$\min\left(E\left[|\mathbf{y}(t)|^2\right] = \mathbf{w}^{\mathrm{H}}\mathbf{R}\mathbf{w}\right) st. \quad \mathbf{w}^{\mathrm{H}}\mathbf{a}(\theta_s) = 1. \quad (14)$$

To simplify the problem, we constrain in the desired direction only. $\mathbf{y}(t) = \mathbf{w}^{\mathrm{H}}\,\mathbf{x}(t)$ is the output of array, $\mathbf{R}$ is the covariance matrix of output signals, $\mathbf{a}(\theta_s)$ is the directional vector in the desired direction, and $\mathbf{w}$ is the weight vector. We can achieve the optimized weight vector by

$$\mathbf{w} = \mathbf{R}^{-1}\mathbf{a}(\theta_s)(\mathbf{a}^{\mathrm{H}}(\theta_s)\mathbf{R}^{-1}\mathbf{a}(\theta_s))^{-1}. \quad (15)$$

For online implementation, we can also develop an iterative formula for beamforming weight calculation [29]

$$\mathbf{w}(k+1) = \mathbf{A}[\mathbf{w}(k) - \mu\,\mathbf{x}(k)\,\mathbf{y}^{\mathrm{H}}(k)] + \mathbf{F},\ k = 1, 2, 3\ldots, \quad (16)$$

where μ is the iteration step size, k is the iteration index, and $\mathbf{A} = \mathbf{I} - \mathbf{a}(\theta_s)\,(\mathbf{a}^{\mathrm{H}}(\theta_s)\,\mathbf{a}(\theta_s))^{-1}\,\mathbf{a}^{\mathrm{H}}(\theta_s)$, $\mathbf{F} = \mathbf{a}(\theta_s)\,(\mathbf{a}^{\mathrm{H}}(\theta_s)\,\mathbf{a}(\theta_s))^{-1}$.

After reconstructing the receiver signal $\mathbf{X}'(t)$ by (8), we can obtain the weight coefficients by iteration as (16). When it satisfies $\|\mathbf{w}(k+1) - \mathbf{w}(k)\|/\|\mathbf{w}(k)\| < \varepsilon$ (ε is a preset error), the iteration stops. Then, we can achieve the most optimized weight coefficients by using the LCMV algorithm.

III. OPTIMIZATION OF THE POSITIONS OF THE ELEMENTS

In this section, a method of optimizing the positions of the elements is proposed, which can help reduce the recovery errors. Here, we define the recovery error as

$$recovery\,error = \frac{norm\left(\mathbf{X}'_{N\times L} - \mathbf{X}_{N\times L}\right)}{norm\left(\mathbf{X}_{N\times L}\right)}, \quad (17)$$

where $\mathbf{X}'_{N\times L}$ is the recovered signal and $\mathbf{X}_{N\times L}$ is the original signal.

Candés and Tao point out that if you want to reconstruct the signal completely, the observation matrix $\mathbf{P}$ must guarantee that two different K sparse signals will not be mapped to a same collection [13]. Here, we define the element of the correlation coefficients matrix $\mathbf{C}$ as follows

$$c_{ij} = \frac{\mathbf{P}_i^{\mathrm{H}}\mathbf{P}_j}{\|\mathbf{P}_i\|_2 \times \|\mathbf{P}_j\|_2}\,(i,\, j = 1, 2, \ldots, M)\,. \tag{18}$$

For reconstructing the signal completely, we should make the correlation coefficients c_{ij} as small as possible. With the given $\mathbf{H}$, we have the optimal correlation coefficients through optimizing the measurement matrix $\mathbf{\Phi}$. Some optimization methods have been proposed recently. Elad [30] attempted to iteratively decrease the average mutual coherence using a shrinkage operation followed by a singular value decomposition step. Abolghasemi proposed a gradient descent method to optimize the measurement matrix in [32, 33], but the measurement matrix optimized by the methods in [30–33] does not satisfy the requirements here. Here, $\mathbf{\Phi}$ shows the elements' positions and can be achieved by randomly selecting M rows from an identity matrix of size $N \times N$. This is to say that $\mathbf{\Phi}$ consists mostly of zeros and a few ones. Therefore, we select a genetic algorithm [2] to optimize the elements' positions.

Our purpose for the optimization is to minimize the correlation coefficients. The larger the correlation coefficients are, the greater the recovery errors are. We want large correlation coefficients to be as small as possible, so we define the cost function as (19):

$$f_{\mathrm{cost}} = \sum_{i=1}^{i=M} \sum_{j=1, i\neq j}^{j=M} c_{i,j}^{k}. \tag{19}$$

We increase the weight of the larger correlation coefficients. The contribution to the cost function grows with the growth of the correlation coefficients. The parameter k shows the magnitude of the growth.

We consider the grid of $\lambda/(2\beta)$ that follows. For simplicity, we will number the grids from 1 to βN, where β shows the size of the grid. The distance between the neighboring grid points is $\lambda/(2\beta)$. Then M grid points will be selected. The genetic algorithm is used to optimize the elements' positions. Because of the mutual coupling between the array elements, the distance between the array elements cannot be too small. Here, we set the minimum spacing $d_{\min}$ to be $\lambda/2$.

To ensure the spacing is greater than $\lambda/2$, we generate a parent vector as

$$\boldsymbol{f} = \begin{bmatrix} f_1 \\ f_2 \\ \vdots \\ f_M \end{bmatrix} = \begin{bmatrix} a_1 + 1 \\ a_2 + \beta + 1 \\ \vdots \\ a_M + (M-1)\,\beta + 1 \end{bmatrix} = \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_M \end{bmatrix} + \begin{bmatrix} 1 \\ \beta + 1 \\ \vdots \\ (M-1)\,\beta + 1 \end{bmatrix}, \tag{20}$$

where $f_i \in [1, \beta N]$ is the position of the ith element, and $a_1 \leqq a_2 \leqq \ldots \leqq a_M$. For $1 \leqq f_1 < \ldots < f_M \leqq \beta N$, $0 \leqq a_1 \leqq \ldots \leqq a_M \leqq \beta N - (M-1)\,\beta - 1$. The vector $\boldsymbol{a} = [a_1, a_2, \ldots, a_M]^{\mathrm{T}}$ can be achieved like this: generate M random integers in the range $[0, \beta N - (M-1)\,\beta - 1]$ and sort the numbers from small to large. The vector $\mathbf{b} = [1, \beta + 1, \ldots, (M-1)\,\beta + 1]^{\mathrm{T}}$ is a constant vector.

It is easy to prove that every individual generated above meets the requirements. We initialize a group by generating Q individuals using the method above. To avoid infeasible solutions in the offspring after genetic manipulation, we preprocess the parent population matrix $\mathbf{F}$ as follows.

Assume that the parents meet all requirements. The group of parents can be written as

$$\mathbf{F} = \begin{bmatrix} f_{1,1} & f_{1,2} & \cdots & f_{1,Q} \\ f_{2,1} & f_{2,2} & \cdots & f_{2,Q} \\ \vdots & \vdots & \ddots & \vdots \\ f_{M,1} & f_{M,2} & \cdots & f_{M,Q} \end{bmatrix}. \tag{21a}$$

Each column in the parent population matrix $\mathbf{F}$ represents a way of elements distribution. Assuming

$$\mathbf{B} = \begin{bmatrix} 1 & 1 & \cdots & 1 \\ \beta+1 & \beta+1 & \cdots & \beta+1 \\ \vdots & \vdots & \ddots & \vdots \\ (M-1)\,\beta+1 & (M-1)\,\beta+1 & \cdots & (M-1)\,\beta+1 \end{bmatrix}, \tag{21b}$$

we obtain the genetic information matrix $\mathbf{A}$ by (22).

The genetic information matrix $\mathbf{A}$ has the same information with $\mathbf{F}$. Therefore, we can do the common genetic operations, including crossover and mutation with the genetic information matrix $\mathbf{A}$.

$$\mathbf{A} = \mathbf{F} - \mathbf{B} = \begin{bmatrix} a_{1,1} & a_{1,2} & \cdots & a_{1,Q} \\ a_{2,1} & a_{2,2} & \cdots & a_{2,Q} \\ \vdots & \vdots & \ddots & \vdots \\ a_{M,1} & a_{M,2} & \cdots & a_{M,Q} \end{bmatrix}. \tag{22}$$

Details of the genetic algorithm are given as follows.

1) Initialize the iteration number $n = 0$ and generate the parent population matrix $\mathbf{F}_0$ with Q individuals by using the method proposed above;

2) Preprocess the parent population matrix $\mathbf{F}_n$ and obtain the genetic information matrix $\mathbf{A}_n$;

3) Calculate the cost values of all individuals through the cost function. Then, select the outstanding genetic information from $\mathbf{A}_n$;

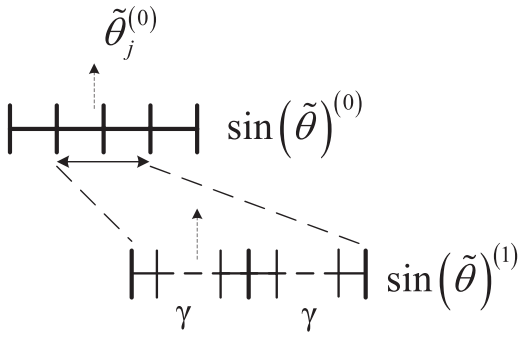


Fig. 1. Illustration of grid refinement.

4) Do the genetic operations, including crossover and mutation with the selected genetic information. Then, the new genetic information $\mathbf{A}_{n+1}$ can be achieved;

5) Get the new parent population matrix by $\mathbf{F}_{n+1} = \mathbf{A}_{n+1} + \mathbf{B}$;

6) If termination criteria are met, stop. Otherwise, go to step (2) and set $n = n + 1$.

IV. TARGETS NOT ON THE GRID

Thus far, in our framework, the estimates of the signal locations are confined to a grid. When the signals are not on the grids, the error of the output signal in (23) grows heavily. Increasing the parameter Γ to reduce the grid spacing can solve this problem, but we cannot make the grid very fine uniformly because this would increase the computational complexity significantly. Instead of having a universally fine grid, we make the grid fine only around the regions where signals are present [16]. This requires an approximate knowledge of the locations of the sources, which can be obtained by using a coarse grid first. The steps to form a nonuniform grid are as follows:

1) Create a rough grid of potential signal locations $\tilde{\theta}_i^{(0)}$, for $I = 1, \ldots, \Gamma$. The grid should not be too rough in order to not introduce substantial bias. Here, for example, we initialize $\Gamma = 200$;

2) Form $\mathbf{H}^{(0)} = \mathbf{H}\left(\tilde{\theta}^{(0)}\right)$, where $\tilde{\theta}^{(0)} = \left[\tilde{\theta}_1^{(0)}, \tilde{\theta}_2^{(0)}, \ldots, \tilde{\theta}_\Gamma^{(0)}\right]$. Use our method from Section II-B to obtain the estimates of the sparse coefficient $\mathbf{S}_{\Gamma\times 1}$. Get the source locations $\tilde{\theta}_j^{(0)}, j = 1, \ldots, K$;

3) Decompose the grid from $\tilde{\theta}_{j-1}^{(0)}$ to $\tilde{\theta}_{j+1}^{(0)}$ into 2γ parts. Then, we have the refined grid $\tilde{\theta}^{(1)}$ around the locations of the peaks $\tilde{\theta}_j^{(0)}$ as Fig. 1 shows;

4) Form a new transformation matrix $\mathbf{H}^{(1)} = \mathbf{H}\left(\tilde{\theta}^{(1)}\right)$, where $\tilde{\theta}^{(1)} = \left[\tilde{\theta}_1^{(1)}, \tilde{\theta}_2^{(1)}, \ldots, \tilde{\theta}_{\Gamma+2\gamma-2}^{(1)}\right]$. Then, do operations as shown in Section II.

V. SIMULATION RESULTS

In this section, we will perform simulations on three aspects. First, simulation results demonstrate the effectiveness of our proposed method for optimizing the elements' positions. Second, beam patterns of two DBF algorithms will be given. Then Monte Carlo simulation is applied to verify the effectiveness of the proposed algorithms. Finally, some simulations of targets that are not on the grid will be given.

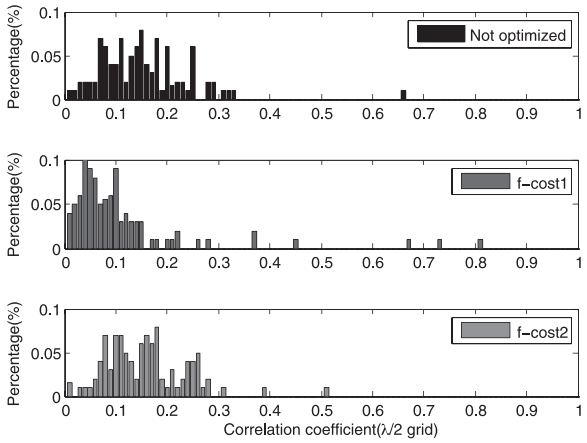


Fig. 2. Histogram distribution of absolute value of entries in C before and after optimization (λ /2 grid).

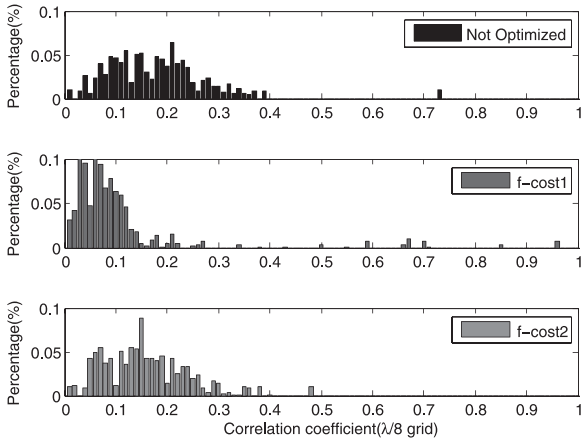


Fig. 3. Histogram distribution of absolute value of entries in C before and after optimization (λ /8 grid).

A. The Optimization of the Elements' Positions

The array parameters are selected as follows: $N = 100$, $M = 30$, $\Gamma = 200$, $\beta = 1$, or 4. Thus, two different element grid intervals are considered: λ /2 and λ /8. Some parameters of the genetic algorithm in Section III are set as follows: populations include 200 individuals; truncation selection (discard bottom 50%); the crossover and mutational probabilities are 80% and 10%, respectively; number of generations is 100.

Figs. 2 and 3 show the histogram distribution of the absolute value of entries (exclusive of the diagonal elements and the same as Table I) in correlation coefficients matrix $\mathbf{C}$ with a different element location grid and different cost function. Figures with the label "Not optimized" correspond to the case in which the sampling element is selected randomly and its location has not been optimized. Figures with label "f-cost1" correspond to the case in which the sampling element is selected randomly

TABLE I.
The Absolute Value of entries in the Correlation Coefficients Matrix **C** with Different element Location Grids and Different Cost Functions.

	$k=1$ ($\lambda/2$ grid)	$k=1$ ($\lambda/8$ grid)	$k=10$ ($\lambda/2$ grid)	$k=10$ ($\lambda/8$ grid)
Cost values	2043	1913	0.204	0.133
Averages	0.103	0.097	0.1507	0.1506
Maximums	0.8	0.95	0.498	0.472

and the location has been optimized with the method in Section III under the cost function with $k=1$ in (19). Figures with label "f-cost2" correspond to the case in which the sampling element is selected randomly and the location has been optimized with the method in Section III under the cost function with $k=10$ in (19).

Compared with Figs. 2 and 3, under the cost function with $k=10$, the maximum of the absolute value of entries in correlation coefficients matrix **C** is smaller in the case in which the element grid interval equals λ /8. The same conclusion can be achieved from Table I, which means that the recovery errors are smaller when the smaller element grid interval is used.

In Table I, while $k=1$, the average of **C** is small, but there are many large correlation coefficients that exist that will increase the recovery error heavily; While $k=10$, the average of **C** is still small, and the maximums of **C** can be optimized to less than 0.5.

B. The Evaluation of Beam Performances

In this subsection, some simulation results under different situations are provided. Assume that there are four space signals from four different directions. One is the desired signal S(t), and the other three are interferences $I_1(t)$, $I_2(t)$, and $I_3(t)$. Set SIR $=-30$ dB. A linear array with 100 elements is assumed. The distance between the neighboring elements of the array is $\lambda/2$. Then, we have the compressed signals and reconstruct the echo signals of all channels from the compressed signals, by the method discussed in Section II-B.

1) *Simulation of Beamforming Based on Orthogonal Projection Algorithm*: The directions of a desired signal and three interferences are set to be 0.1, -0.21, 0.41, and -0.45 (sin (θ)). Set SNR $=10$ dB, INR $=40$ dB. The optimized element numbers are 1, 3, 5, 6, 8, 11, 12, 16, 20, 24, 28, 29, 32, 42, 45, 51, 63, 65, 70, 71, 78, 81, 85, 87, 89, 92, 95, 96, 98, and 100.

Three kinds of the beam patterns are shown in Fig. 4 using the recovered data, the received data with 30 elements, and the received data with full array, respectively.

Fig. 4 shows that when the number of the elements reduces from 100 to 30 without reducing the antenna aperture, the sidelobe of the beam formed by the data of 30 elements directly is -8 dB. The beam formed by the method proposed in this paper has the similar performance with the beam formed by the full array (100 elements)

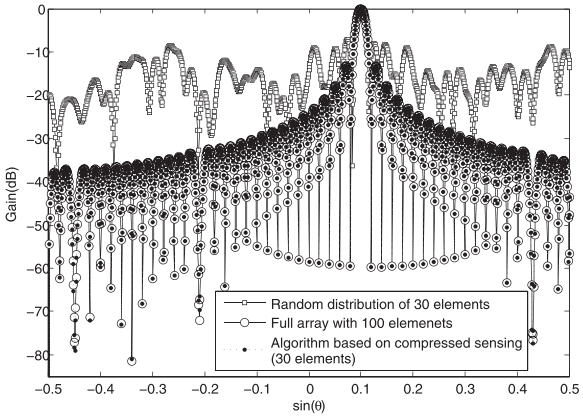


Fig. 4. Beam patterns based on orthogonal projection algorithm (SNR $=10$ dB, INR $=40$ dB).

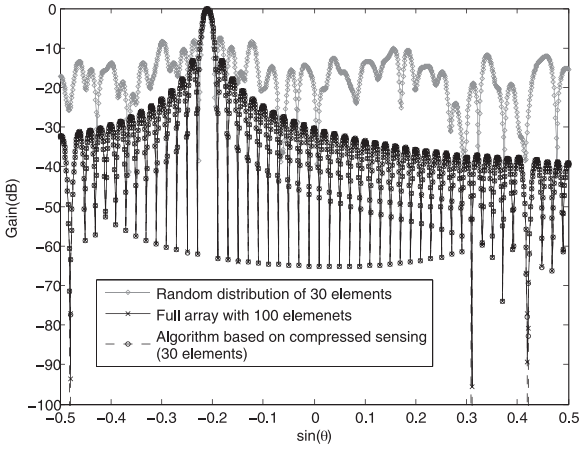


Fig. 5. Beam patterns of iterative LCMV algorithm.

data: deep nulls in the directions of interference, low sidelobes, and no grating lobes. So, the method applies to the orthogonal projection algorithm.

2) *Simulation of Iterative LCMV Algorithm*: The directions of a desired signal and three interferences are set to be -0.21, -0.48, 0.31, and 0.42 (sin(θ)). Set SNR $=10$ dB, INR $=40$ dB. Form the beams under the same three situations and compare the three kinds of the beams. Simulation results are shown in Fig. 5.

Three beam patterns in Fig. 5 show that when the number of the elements reduces from 100 to 30 without reducing the antenna aperture, the sidelobe of the beam formed by the data of 30 elements directly is -9 dB. The beam formed by the proposed method has similar

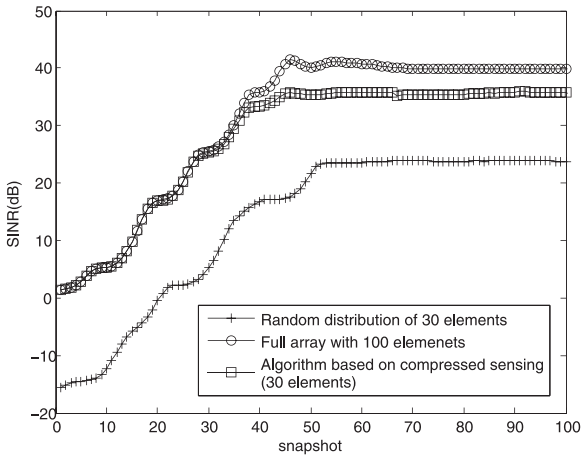


Fig. 6. Convergence in different methods (SNR = 10 dB, INR = 40 dB).

performance to the beam formed by the full array (100 elements) data.

The curves in Fig. 6 show the changes in output signal-to-interference-noise ratio (SINR) with snapshots in three different ways: after the output SINR is constrained, the SINR achieved by the proposed method with the recovered data is smaller than it achieved with the original data of the full array, but much larger than it achieved with the original data of 30 elements. So, the method also applies to the iterative LCMV algorithm.

3) *The Monte Carlo Simulation*: In radar operation, the direction of receiving beam is usually steered by the control system. The SNR of received desired signal returned from target and the residual jamming power after adaptive suppression are most important. In our proposed CS-based adaptive beamforming method, the received signal on full array is reconstructed from the received signal on sampling elements. While estimating the sparse vector $\mathbf{S}_{\Gamma\times L}$ of the desired signal and interference by using the OMP algorithm, the desired signal may be lost when SNR of the desired signal received by sampling elements is lower, which leads to serious loss for the desired signal in the recovered signal $\mathbf{X}'_{N\times L}$ and that in the beam output. To evaluate the performance on SNR loss of the desired signal in beam output, we define the desired signal error of the output signal as

$$DS_{error} = 20 \times \ln\left(\frac{norm\left(\mathbf{Y}_{T\times 1} - \mathbf{S}_{T\times 1}\right)}{norm\left(\mathbf{S}_{T\times 1}\right)}\right), \qquad (23)$$

where $\mathbf{Y}_{T\times 1}$ is the normalized output signal of the DBF system and $\mathbf{S}_{T\times 1}$ is the normalized desired signal without noise.

To verify the correctness of the algorithm in the cases of different SNRs, different INRs, and different directions of signals, we randomly select the directions of the desired signal and interference. We, then, do Monte Carlo analysis 200 times under the situations given below. The directions of the interference are out of the main lobe.

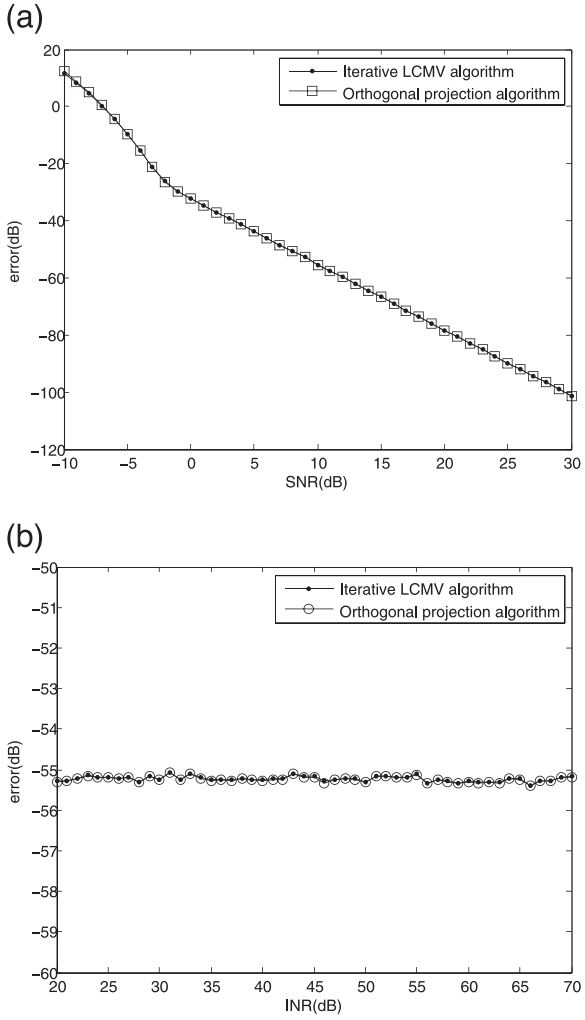


Fig. 7. Errors of output signals in different situations. (a) SNR = −10 ∼ 30 dB, SIR = −30 dB. (b) SNR = 10 dB, SIR = −70 ∼ −10 dB.

Situation 1: SNR = −10 ∼ 30 dB, SIR = −30 dB;
Situation 2: SNR = 10 dB, SIR = −70 ∼ −10 dB;

Fig. 7(a) shows the average desired signal error when SIR = −30 dB, and SNR varies from −10 dB to 30 dB. From Fig. 7(a), we can see that when the SNR is lower than 0 dB, the desired signal will be lost when reconstructing the signal sometimes. Thus, the recovered desired signal is corrupted, and the target cannot be detected. When the SNR is higher than 0 dB, the desired signal can be recovered reliably. Fig. 7(b) shows the average desired signal error when SNR = 10 dB, and SIR varies from −70 dB to −10 dB. From Fig. 7(b), we can see that no matter how large the interference is, the desired signal can be recovered reliably.

To evaluate the performance on interference suppression, Monte Carlo simulations about sidelobe level and null depth on interference directions are accomplished. Fig. 8 shows the two simulation results with SIR = −30 dB. The curves in Fig. 8(a) are the result of the Monte Carlo analysis using the orthogonal projection algorithm, and the curves in Fig. 8(b) are the result of the Monte Carlo analysis using the iterative LCMV algorithm.

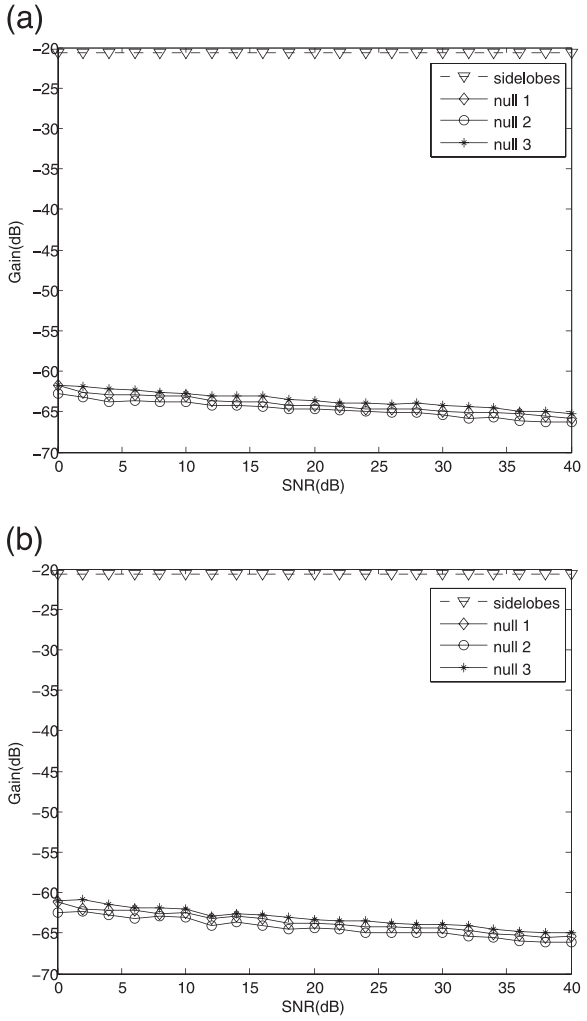


Fig. 8. Levels of sidelobe and depth of nulls in direction of interference under different methods. (a) Orthogonal projection algorithm. (b) Iterative LCMV algorithm.

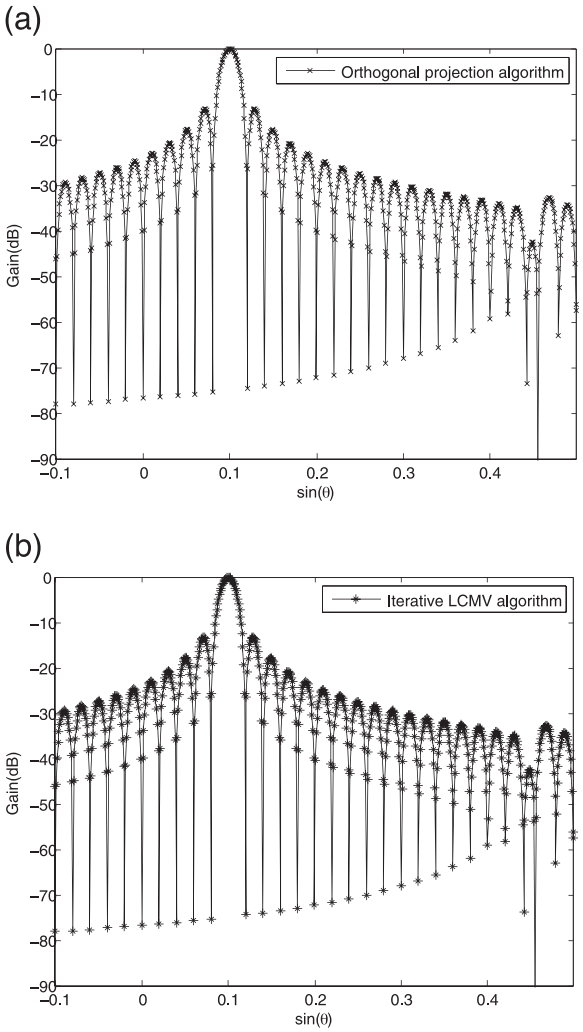


Fig. 10. Beam patterns of targets not on grid. (a) Orthogonal projection algorithm. (b) Iterative LCMV algorithm.

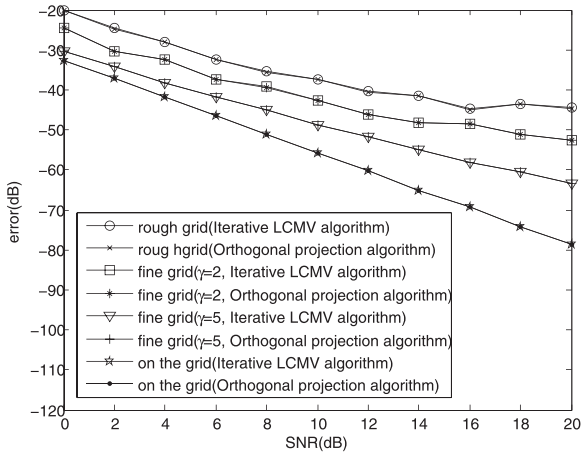


Fig. 9. Errors of output signals with rough and fine grid.

The levels of the sidelobes and the nulls are achieved after completing the iterations. As a result, good performance of the beam pattern can be obtained by both algorithms.

C. Simulations of Targets Not on the Grid

In this subsection, we consider the situation of the targets not on the grid. Assume that there are two space signals from two different directions. One is the desired signal S(t), and the other is the interference I(t). The directions of the desired signal and the interference are randomly generated (not on the grid). Set SNR $= 0 \sim 20$ dB, SIR $= -30$ dB. Then, we do Monte Carlo analysis 100 times, as shown in the following.

Fig. 9 shows the results of targets not on the grid. From Fig. 9, we can see that when the desired signal or the interference is not on the grid, the algorithm fails. Through the method proposed in Section IV, we obtain a fine grid (with $\gamma = 2$ or $\gamma = 5$) nonuniformly. For the fine grid nonuniformly, the errors of the output signals reduce. To observe the beam patterns, the directions of the desired signal and the interference are set to be 0.105 and 0.455 ($\sin(\theta)$). Fig. 10 shows the beam patterns with $\gamma = 5$. We can see that the null is deep in the direction of the interference. Therefore, when targets are not on the grid,

the algorithm still works with the method proposed in Section IV.

VI. CONCLUSIONS

A new adaptive digital beamforming in the receiving end based on CS is presented in this paper. This is a new way to reduce the elements of the array without reducing the antenna aperture or deteriorating the performance of the beams, and it greatly reduces the number of RF frontends. According to the sparsity of the target in space, the algorithm accurately reconstructs the data of the whole channels of the full array by the method of CS, with the data received by these 30 elements. It also forms the digital beams with the recovered data. Simulation results demonstrate the effectiveness of the algorithm. Simulation results show that the new adaptive digital beamforming algorithm is better than the ordinary method with 30 elements and has the similar beam performance with the full array. When the signal is too weak, the algorithm fails. The desired signal will be lost when reconstructing the signals. The recovered data is corrupted and cannot be used for signal processing. So, the algorithm can only be used when SNR is greater than 0 dB.

REFERENCES

[1] Haupt, R. L.
Thinned arrays using genetic algorithms.
IEEE Transactions on Antennas and Propagation, **42**, 7 (July 1994), 993–999.

[2] Chen, K., He, Z., and Han, C.
A modified real GA for the sparse linear array synthesis with multiple constraints.
IEEE Transactions on Antennas and Propagation, **54**, 7 (2006), 2169–2173.

[3] Hooker, J. W., and Arora, J. K.
Optimal thinning levels in linear arrays.
IEEE Antennas and Wireless Propagation Letters, **9** (2012), 771–774.

[4] Quevedo-Teruel, O., and Rajo-Iglesias, E.
Ant colony optimization in thinned array synthesis with minimum sidelobe level.
IEEE Antennas and Wireless Propagation Letters, **5** (2006), 349–352.

[5] Trucco, A., Omodei, E., and Repetto, P.
Synthesis of sparse planar arrays.
IET Electronics Letters, **33**, 22 (Oct. 1997), 1834–1835.

[6] Liu, Y., Nie, Z., and Liu, Q. H.
Reducing the number of elements in a linear antenna array by the matrix pencil method.
IEEE Transactions on Antennas and Propagation, **56**, 3 (Sep. 2008), 2955–2962.

[7] Goudos, S. K., Siakavara, K., Samaras, Vafiadis, E. E., and Sahalos, J. N.
Sparse linear array synthesis with multiple constraints using differential evolution with strategy adaptation.
IEEE Antennas and Wireless Propagation Letters, **10** (July 2011), 670–673.

[8] Wilson, M. J., and McHugh, R.
Sparse-periodic hybrid array beamformer.
IET Radar Sonar Navigation, **1**, 2 (May 2007), 116–123.

[9] Nai, S. E., Ser, W., Yu, Z. L., and Chen, H.
Beampattern synthesis for linear and planar arrays with antenna selection by convex optimization.
IEEE Transactions on Antennas and Propagation, **58**, 12 (Dec. 2010), 3923–3930.

[10] Fuchs, B.
Synthesis of sparse arrays with focused or shaped beampattern via sequential convex optimizations.
IEEE Transactions on Antennas and Propagation, **60**, 7 (July 2012), 3499–3503.

[11] Oliveri, G., and Massa, A.
Bayesian compressive sampling for pattern synthesis with maximally sparse non-uniform linear arrays.
IEEE Transactions on Antennas and Propagation, **59**, 2 (Feb. 2011), 467–481.

[12] Zhang, W., Li, L., and Li, F.
Reducing the number of elements in linear and planar antenna arrays with sparseness constrained optimization.
IEEE Transactions on Antennas and Propagation, **59**, 8 (Aug. 2011), 3106–3111.

[13] Candés, E. J., Romberg, J., and Tao, T.
Robust uncertainty principles: exact signal reconstruction from highly incomplete frequency information.
IEEE Transactions on Information Theory, **52**, 2 (Feb. 2006), 489–509.

[14] Donoho, D. L.
Compressed sensing.
IEEE Transaction on Information Theory, **52**, 4 (Apr. 2006), 1289–1306.

[15] Bobin, J., Starck, J.-L., and Ottensamer, R.
Compressed sensing in astronomy.
IEEE Journal of Selected Topics in Signal Processing, **2**, 5 (Oct. 2008), 718–726.

[16] Malioutov, D., Çetin, M., and Willsky, A. S.
A sparse signal reconstruction perspective for source localization with sensor arrays.
IEEE Transactions on Signal Processing, **53**, 8 (Aug. 2005), 3010–3023.

[17] Gurbuz, A. C., Cevher, V., and Meclellan, J. H.
Bearing estimation via spatial sparsity using compressive sensing.
IEEE Transactions on Aerospace and Electronic Systems, **48**, 2 (Apr. 2012), 1358–1369.

[18] Herman, M., and Strohmer, T.
Compressed sensing radar. In *Proceeding of IEEE International Conference on Acoustics, Speech and Processing*, Las Vegas, Nevada, Apr. 2008, 1509–1512.

[19] Zhang, J., Zhu, D., and Zhang, G.
Adaptive compressed sensing radar oriented toward cognitive detection in dynamic sparse target scene.
IEEE Transactions on Signal Processing, **60**, 4 (Apr. 2012), 1718–1729.

[20] Anitori, L., Maleki, A., Otten, M., Baraniuk, R. G., and Hoogeboom, P.
Design and analysis of compressed sensing radar detectors.
IEEE Transactions on Signal Processing, **61**, 4 (Feb. 2013), 813–827.

[21] Ma, J.
Single-pixel remote sensing.
IEEE Transactions on Geoscience and Remote Sensing, **6**, 2, (Apr. 2009), 199–203.

[22] Mamaghanian, H., Khaled, N., Atienza, D., and Vandergheynst, P.
Design and exploration of low-power analog to information conversion based on compressed sensing.
IEEE Journal on Emerging and Selected Topic in Circuits and Systems, **2**, 3 (Sep. 2012), 493–501.

[23] Zhao, Y., Hu, Y. H., and Wang, H.
Enhanced random equivalent sampling based on compressed sensing.
IEEE Transactions on Instrumentation and Measurement, **61**, 3 (Mar. 2012), 579–586.

[24] Tropp, J. A., and Gilbert, A. C.
Signal recovery from random measurements via orthogonal

matching pursuit.
IEEE Transactions on Information Theory, **53**, 12 (Dec. 2007), 4655–4666.

[25] Wang, W., Kwon, S., and Shim, B.
Generalized orthogonal matching pursuit.
IEEE Transactions on Signal Processing, **60**, 12 (Dec. 2012), 6202–6216.

[26] Baraniuk, R., Davenport, M., DeVore, R., and Wakin, M.
A simple proof of the restricted isometry property for random matrices.
Constructive Approximation, **28**, (2008), 253–263.

[27] Subbaram, H., and Abend, K.
Interference suppression via orthogonal projections: a performance analysis.
IEEE Transactions on Antennas and Propagation, **41**, 9 (Sept. 1993), 1187–1194.

[28] Griffiths, L. J., and Jim, C. W.
An alternative approach to linearly constrained adaptive beamforming.
IEEE Transactions on Antennas and Propagation, **30**, 1 (Jan. 1982), 27–34.

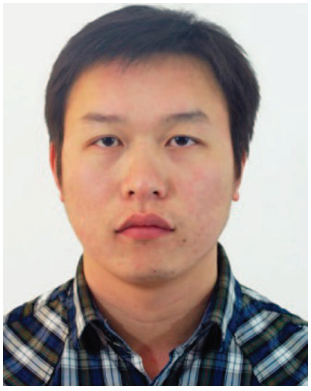
[29] Huang, F., Sheng, W., Ma, M., and Wang, W.
Robust adaptive beamforming for large-scale arrays.
Signal Processing, **90**, (2010), 165–172.

[30] Elad, M.
Optimized projections for compressed sensing.
IEEE Transactions on Signal Processing, **57**, 12 (Dec. 2007), 5695–5702.

[31] Duarte-Carvajalino, J. M., and Sapiro, G.
Learning to sense sparse signals: simultaneous sensing matrix and sparsifying dictionary optimization.
IEEE Transactions on Image Processing, **18**, 7 (Jul. 2009), 1395–1408.

[32] Abolghasemi, V., Ferdowsi, S., Makkiabadi, B., and Sanei, S.
On optimization of the measurement matrix for compressive sensing. *Presented at the 18th European Signal Processing Conference*, Aalborg, Denmark, Aug. 2010, 427–431.

[33] Abolghasemi, V., Ferdowsi, S., and Saeid Sanei, S.
A gradient-based alternating minimization approach for optimization of the measurement matrix in compressive sensing.
Signal Processing, **92**, (2012), 999–1009.



Jian Wang was born in Zhejiang Province, China, in November 1987. He received the bachelor degree in electronic and information engineering from Nanjing University of Science and Technology (NJUST), Nanjing, China, in 2010. Now, he is a Ph.D candidate in communication and information systems at NJUST, China. His research interests include compressive sensing applications and array signal processing.

Wei-Xing Sheng (M'97) was born in Jiangsu in November 1966. He received the B.Sc. M.S., and Ph.D. degrees in electronic engineering from Shanghai Jiaotong University, NJUST in 1988, 1991, and 2002, respectively. Since 1991, he has been with NJUST, where he is currently a professor in the Department of Communication Engineering. His research interests include array antenna, array signal processing, and signal processing in radar or communication systems.

Yu-bing Han received the B.S. degree from Huazhong University of Science and Technology, China, in 1994, the M.S. degree from Hohai University, China, in 1999, both in applied mathematics, and the Ph.D. degree in signal and information processing from Southeast University, China, in 2006.

From 2007 to 2009, he was a postdoctoral researcher in the Department of Electronic Engineering at NJUST, Nanjing, China. Since 2006, he has been a faculty member in the School of Electronic and Optical Engineering at NJUST, where he is an associate professor. From 2009 to 2010, he was a visiting scholar with Department of Electrical and Computer Engineering, Brigham Young University, Provo, Utah. His current research interests include array signal processing, multiple-input multiple-output (MIMO) wireless communications, and digital image processing.

Xiao-feng Ma was born in Jiangsu Province China in May 1981. He received the B.Sc. and M.S. degrees in electronic engineering and communication and information systems from NJUST, China, in 2004 and 2006, respectively. Now, he is Ph.D. candidate in communication and information systems and a lecturer in the Department of Communication Engineering at NJUST. His research interests include array signal processing, MIMO radar signal processing, and software defined radio.