Progress In Electromagnetics Research B, Vol. 41, 251–268, 2012

PERFORMANCE ANALYSIS OF STAP ALGORITHMS BASED ON FAST SPARSE RECOVERY TECHNIQUES

Z. C. Yang*, Z. Liu, X. Li, and L. Nie

Electronics Science and Engineering School, National University of Defense Technology, Changsha 410073, China

Abstract—In the field of space-time adaptive processing (STAP), spare recovery type STAP (SR-STAP) algorithms exploit formulation of the clutter estimation problem in terms of sparse representation of a small number of clutter positions among a much larger number of potential positions in the angle-Doppler plane, and provide an effective approach to suppress the clutter especially in very short snapshots. However, it differs from many situations encountered by other SR application fields in the following ways: (i) it does not require to obtain the exact solution; (ii) it highly requires low-complexity approaches. In this paper, we focus on the performance analysis and parameters setting of STAP algorithms based on five representative fast SR techniques, namely, the compressive sampling matching pursuit, the sparse reconstruction by separable approximation, the fast iterative shrinkage-thresholding algorithm, the focal underdetermined system solution and the smoothed l_0 norm method.

1. INTRODUCTION

Space-time adaptive processing (STAP) methods have been actively considered for look down airborne radar where target signals have to compete with strong ground clutter returns [1–3]. However, there are many practical limitations preventing the use of the full-rank STAP processor. One of them is the requirement of a large number of independent and identically distributed (IID) training samples to estimate the interference covariance matrix [4], which becomes even more serious in practical situations because of the non-stationary and nonhomogeneous interference environments [5].

Received 11 April 2012, Accepted 31 May 2012, Scheduled 6 June 2012

* Corresponding author: Zhao Cheng Yang (yangzhaocheng@gmail.com).

Recently, motivated by compressive sensing (CS) and sparse recovery (SR) techniques used in radar, several authors have considered CS and SR ideas for moving target indication (MTI) and STAP problems, such as sparse-recovery-based STAP type (SR-STAP) algorithms in [6–10], L1-regularized STAP filters in [11–13], etc.. The core idea in SR-STAP type algorithms is to regularize a linear inverse problem by including prior knowledge that the signal of interest is sparse [10]. The work in [6–10] shows that the SR-STAP type algorithms provide high-resolution of the clutter spectrum estimate and exhibit significant better performance than conventional STAP algorithms in very short snapshots. However, their performance and computational complexity depends on the qualities of SR algorithms.

For STAP problem in airborne radar systems, although the clutter spectrum estimate can be formulated as a linear inverse problem with sparse regularization, it differs from many situations encountered by other SR application fields [14–18] in the following ways: (i) it does not require to obtain the exact solution because it will achieve a good performance after suppressing significant clutter components; (ii) it highly requires low computational complexity approaches due to the special task in airborne radar applications. Therefore, whether SR algorithms would work in STAP problems, how difficult is the related parameters setting of these SR algorithms in STAP problems, and which algorithm is suitable for STAP problem are worthy for a deep investigation. Because it is impossible to summarize all existing fast SR algorithms in the literature, in this paper, we mainly focus on five representative approaches, namely, the compressive sampling matching pursuit (CoSaMP) [19, 20], the sparse reconstruction by separable approximation (SpaRSA) [21], the fast iterative shrinkage-thresholding algorithm (FISTA) [22], the focal underdetermined system solution (FOCUSS) [9, 23] and the smoothed l_0 norm method [24]. Furthermore, we will detail the parameters setting and show the performance analysis of these algorithms under different practical considerations, e.g., inner clutter motion (ICM), array errors, etc..

The following parts of this paper are organized as: we will first introduce the signal model in Section 2 and summarize the principle of SR-STAP type algorithms in Section 3. Then, in Section 4, above five algorithms will be briefly reviewed for further investigation. Section 5 will discuss the parameters setting and access the performance of all algorithms by showing the signal-interference-to-noise ratio (SINR), and detection performance with different scenarios. Section 6 provides the summary and conclusions.

2. SIGNAL MODEL

In airborne radar systems, a general model for the space-time clutter plus noise snapshot $\mathbf{x}$ in a target-free range bin is given by [3]

$$\mathbf{x} = \mathbf{x}_c + \mathbf{n} = \sum_{m=1}^{N_a} \sum_{n=1}^{N_c} \sigma_{m,n} \mathbf{v}(\phi_{m,n}, f_{m,n}) + \mathbf{n}, \tag{1}$$

where $\mathbf{n}$ is the Gaussian white thermal noise vector, with the noise power σ_n^2 on each channel and pulse; N_a is the number of range ambiguities; N_c is the number of independent clutter patches over the iso-range of interest; $\phi_{m,n}$ is the angle-of-arrival (AOA) of the mnth clutter patch; $f_{m,n}$ is the corresponding Doppler frequency; $\sigma_{m,n}$ is the complex amplitude for the mnth clutter patch, with each element proportional to the square-root of the clutter patches' radar cross section (RCS); $\mathbf{v}\,(\phi_{m,n},\, f_{m,n})$ is the $NM \times 1$ space-time steering vector for the clutter patch with the AOA $\phi_{m,n}$ and the Doppler frequency $f_{m,n}$. Here, N is the number of pulses in a coherent process interval (CPI) and M is the number of array channels.

The space-time steering vector is given as a Kronecker product of the temporal and spatial steering vectors, denoted as $\mathbf{v}(\phi_{m,n}, f_{m,n}) = \mathbf{v}_t(f_{m,n}) \otimes \mathbf{v}_s(\phi_{m,n})$. Ignoring the impact of range ambiguities (in the following the symbols will drop the subscript 'm') and considering a uniform linear array (ULA) with inner spacing d_a, the temporal and spatial steering vectors of the nth clutter patch are given by [1]

$$\mathbf{v}_t(f_n) = [1, \exp(j2\pi f_n), \ldots, \exp(j2\pi(N-1)f_n)]^T, \tag{2}$$

$$\mathbf{v}_s(\phi_n) = \left[1, \exp\left(\frac{j2\pi d_a}{\lambda_c} \sin\phi_n, \ldots, \exp\left(\frac{j2\pi(M-1)d_a}{\lambda_c} \sin\phi_n\right)\right]^T, \tag{3}$$

where superscript T denotes the transposition operation, and λ_c is the operate wavelength. If we stack all clutter patches' amplitudes into a vector $\boldsymbol{\sigma} = [\sigma_1, \ldots, \sigma_{N_c}]^T$, then the clutter component in (1) can be rewritten as

$$\mathbf{x}_c = \mathbf{V}\boldsymbol{\sigma}, \tag{4}$$

where $\mathbf{V} = [\mathbf{v}(\phi_1, f_1), \ldots, \mathbf{v}(\phi_{N_c}, f_{N_c})]$ denotes the clutter space-time steering matrix. Herein, the clutter covariance matrix based on (4) can be expressed as

$$\mathbf{R}_c = E\left[\mathbf{x}_c \mathbf{x}_c^H\right] = \mathbf{V}\boldsymbol{\Sigma}\mathbf{V}^H, \tag{5}$$

where superscript H denotes the conjugate transposition operation and $\boldsymbol{\Sigma} = E[\boldsymbol{\sigma}\boldsymbol{\sigma}^H]$. Under the condition that the clutter patches are independent each other, $\boldsymbol{\Sigma} = \mathrm{diag}(\mathbf{a})$, $\mathbf{a} = [a_1, \ldots, a_{N_c}]^T$ and $a_n = E[|\sigma_n|^2]$, $n = 1, \ldots, N_c$ for the statistics of the clutter patches.

3. PRINCIPLE OF SR-STAP TYPE ALGORITHMS

In this section, we provide an alternative method to describe the received signal. Since the clutter component is a function of the Doppler frequency and the AOA, we discretize the whole angle-Doppler plane into $N_s = \rho_s M$, $N_d = \rho_d N$, $(\rho_s, \rho_d > 1)$ grids, where N_s and N_d are the number of angle and Doppler, respectively. Then the clutter component in (4) can be rewritten as

$$\mathbf{x}_c = \mathbf{\Phi}\boldsymbol{\gamma}, \tag{6}$$

where $\boldsymbol{\gamma} = [\gamma_1, \gamma_2, \ldots, \gamma_{N_d N_s}]^T$ denotes the $N_d N_s \times 1$ angle-Doppler profile with non-zero elements representing the clutter patches, and the $NM \times N_d N_s$ matrix $\mathbf{\Phi}$ is the space-time steering dictionary, as given by

$$\mathbf{\Phi} = [\mathbf{v}(f_1, \phi_1), \ldots, \mathbf{v}(f_1, \phi_{N_s}), \ldots, \mathbf{v}(f_{N_d}, \phi_{N_s})]. \tag{7}$$

Thus, the received signal in the target range bin can be reformulated as

$$\mathbf{x} = \alpha\mathbf{s} + \mathbf{\Phi}\boldsymbol{\gamma} + \mathbf{n}, \tag{8}$$

where α is a complex gain, and $\mathbf{s}$ is the $NM \times 1$ space-time steering vector in the target look-direction with similar formulation as $\mathbf{v}$.

From (5), we know that we can estimate the clutter covariance matrix $\mathbf{R}_c$ by estimating the statistics $\mathbf{a}$ of the clutter patches first. In the new signal model described in (8), it is equal to estimate $\mathbf{R}_c$ by

$$\hat{\mathbf{R}}_c = \mathbf{\Phi}\hat{\hat{\mathbf{\Sigma}}}\mathbf{\Phi}^H = \mathbf{\Phi}\mathrm{diag}\left(\hat{\hat{\mathbf{a}}}\right)\mathbf{\Phi}^H, \tag{9}$$

where $\hat{\hat{\mathbf{a}}} = [\hat{\hat{a}}_1, \ldots, \hat{\hat{a}}_{N_d N_s}]^T$, and the parameters $\hat{\hat{a}}_i, i = 1, \ldots, N_d N_s$ are estimated by using snapshots from adjacent target-free range bins to avoid impact of the target signal

$$\hat{\hat{a}}_i = \sum_{l=1}^{L} |\hat{\gamma}_{l;i}|^2, \quad l = 1, 2, \ldots, L, \tag{10}$$

where L is the number of training snapshots, and $\hat{\boldsymbol{\gamma}}_l$, $l = 1, 2, \ldots, L$ are the estimated angle-Doppler profiles. The basic idea of SR-STAP type algorithms is to exploit the intrinsic sparsity of the clutter angle-Doppler profile in the angle-Doppler plane and seek to obtain the high-resolution space-time spectrum via the sparse recovery techniques [6–10], where the whole procedure can be divided into the following steps:

Step 1: Estimate the clutter angle-Doppler profiles from snapshots $\mathbf{x}_l$, $l = 1, 2, \ldots, L$ in the adjacent target-free range bins by solving the minimization problem of

$$\mathrm{P1}: \quad \hat{\boldsymbol{\gamma}}_l = \arg\min_{\boldsymbol{\gamma}_l} \left\{ \frac{1}{2}\|\mathbf{x}_l - \mathbf{\Phi}\boldsymbol{\gamma}_l\|_2^2 + \kappa\|\boldsymbol{\gamma}_l\|_p \right\}, \tag{11}$$

or

$$\text{P2}: \quad \hat{\boldsymbol{\gamma}}_l = \arg\min_{\boldsymbol{\gamma}_l} \|\boldsymbol{\gamma}_l\|_p \quad \text{subject to} \quad \|\mathbf{x}_l - \boldsymbol{\Phi}\boldsymbol{\gamma}_l\|_2^2 \leq \epsilon, \tag{12}$$

where $\|\cdot\|_p$ $(0 \leq p \leq 2)$ denotes the l_p-norm; κ is a positive regularization parameter that provides a tradeoff between l_p-norm and l_2-norm; ϵ is the noise error allowance.

Step 2: Substitute the estimated angle-Doppler profiles into (10) and compute the clutter covariance matrix according to (9).

Step 3: Estimate the noise power level and calculate the STAP filter weight vector $\hat{\mathbf{w}}$ via the Capon's optimal filter

$$\hat{\mathbf{w}} = \mu\left[\hat{\mathbf{R}}_c + \hat{\sigma}_n^2\mathbf{I}\right]^{-1}\mathbf{s}, \tag{13}$$

where μ is a constant which does not affect the SINR performance, $\hat{\sigma}_n^2$ is the estimated noise level, and $\mathbf{I}$ denotes the identity matrix.

Step 4: Compute the filter output $y = \hat{\mathbf{w}}^H\mathbf{x}$, where $\mathbf{x}$ is the received signal in the test range bin.

4. FAST SR TECHNIQUES

In this section, due to space limitation, we just review the basic ideas and list the pseudo codes of several typical fast sparse recovery algorithms, such as the CoSaMP, SpaRSA, FISTA, FOCUSS and SL0 algorithms. Readers are referred to [9, 19–24] for further derivations.

4.1. CoSaMP

CoSaMP uses an approach inspired by the restricted isometry property (RIP) to identify the locations of the largest components in the signal [19, 20]. It is supposed that $\boldsymbol{\Phi}$ satisfies the RIP of order N_n with constant $\delta_{N_n} \ll 1$, where N_n is the number of nonzero elements of $\boldsymbol{\gamma}$. The vector $\mathbf{z} = \boldsymbol{\Phi}^H\boldsymbol{\Phi}\boldsymbol{\gamma}$ can serve as a proxy for the signal because the energy in each set of N_n components of $\mathbf{z}$ approximates the energy in the corresponding N_n components of $\boldsymbol{\gamma}$. Since the samples have the form $\mathbf{x}_c = \boldsymbol{\Phi}\boldsymbol{\gamma}$, it can obtain the proxy just by applying the matrix $\boldsymbol{\Phi}^H$ to the samples. The CoSaMP algorithm invokes this idea iteratively to approximate the signal. The whole procedure is formulated by the pseudo code in Table 1.

4.2. SpaRSA

Rather than a specific algorithm, SpaRSA is an algorithmic framework for optimization problems of the form [21]:

$$\min_{\boldsymbol{\gamma}} F(\boldsymbol{\gamma}) + \kappa g(\boldsymbol{\gamma}), \tag{14}$$

Table 1. CoSaMP algorithm in [19].

Initialization: Set $\boldsymbol{\gamma}_0 = \mathbf{0}$, the residual $\mathbf{r}_0 = \mathbf{x}$, the iteration index $p = 1$, tuning parameter β.
Repeat: (1) Find βN_n columns of $\boldsymbol{\Phi}$ that are most strongly correlated with the residual: $\Omega \in \arg\min_{\lvert T\rvert \le \beta N_n} \sum_{i\in T} \lvert\langle \mathbf{r}_{p-1}, \mathbf{v}_i\rangle\rvert$. (2) Put the old and new columns into one set: $T = \mathrm{supp}(\boldsymbol{\gamma}_{p-1}) \cup \Omega$ (3) Find the best coefficients for approximating the residual with these columns: $\mathbf{z}_p = \arg\min_{\mathbf{z}} \lVert r_{p-1} - \boldsymbol{\Phi}_T \mathbf{z}\rVert_2$. (4) Retain the N_n largest coefficients: $\boldsymbol{\gamma}_p = [\mathbf{z}]_{N_n}$. (5) Update the residual: $\mathbf{r}_p = \mathbf{x} - \boldsymbol{\Phi}\boldsymbol{\gamma}_p$. **Until stopping criterion holds.**
Output: Return $\boldsymbol{\gamma} = \boldsymbol{\gamma}_p$.

where $F(\boldsymbol{\gamma})$ is a smooth function, and $g(\boldsymbol{\gamma})$ called the regularizer or regularization function, is finite for all $\boldsymbol{\gamma} \in \mathbb{C}$. Specially, for solving the optimization problem $P1$ when $p = 1$, the SpaRSA framework can be realized by generating a sequence of iterates $\boldsymbol{\gamma}_p$, $p = 0, 1, \ldots$ and is tailored to problems in which the following subproblem can be set up and solved efficiently at each iteration:

$$\boldsymbol{\gamma}_{p+1} \in \arg\min_{\boldsymbol{\gamma}} \frac{1}{2}\lVert\boldsymbol{\gamma} - \mathbf{u}_p\rVert_2^2 + \frac{\kappa}{\beta_p}\lVert\boldsymbol{\gamma}\rVert_1 \tag{15}$$

where $\mathbf{u}_p = \boldsymbol{\gamma}_p + \frac{1}{\beta_p}\boldsymbol{\Phi}^H(\mathbf{x} - \boldsymbol{\Phi}\boldsymbol{\gamma})$, and $\beta_p \in \mathbb{R}^+$. Then, $\boldsymbol{\gamma}_{p+1}$ can be obtained by

$$\gamma_{i,p+1} = \mathrm{soft}\left(u_{i,p}, \frac{\kappa}{\beta_p}\right) = \frac{\max\left\{|u_{i,p}| - \frac{\kappa}{\beta_p}, 0\right\}}{\max\left\{|u_{i,p}| - \frac{\kappa}{\beta_p}, 0\right\} + \frac{\kappa}{\beta_p}} u_{i,p}, \tag{16}$$

where $i = 1, \ldots, N_d N_s$. The above form is often under the name of iterative shrinkage/thresholding (IST) algorithms. The pseudo code of SpaRSA algorithm based on Barzilai-Borwein methods is summarized in Table 2.

4.3. FISTA

FISTA is an extension of the IST algorithm, which preserves the computational simplicity of IST algorithm but with a global rate

Table 2. SpaRSA algorithm based on Barzilai-Borwein methods in [21].

Initialization: Choose initial guess $\boldsymbol{\gamma}_0$, the iteration index $p = 1$.
Repeat:
(1) $\triangle\boldsymbol{\gamma}_p = \boldsymbol{\gamma}_p - \boldsymbol{\gamma}_{p-1}$, $\mathbf{r}_p = \boldsymbol{\Phi}^H\boldsymbol{\Phi}(\boldsymbol{\gamma}_p - \boldsymbol{\gamma}_{p-1})$
(2) Estimate parameter β_p:
$\beta_p = \arg\min_\beta \|\beta\triangle\boldsymbol{\gamma}_p - \mathbf{r}_p\|_2^2 = \frac{\|\boldsymbol{\Phi}\triangle\boldsymbol{\gamma}_p\|_2^2}{\|\triangle\boldsymbol{\gamma}_p\|_2^2}$.
(3) Compute: $\mathbf{u}_p = \boldsymbol{\gamma}_p + \frac{1}{\beta_p}\boldsymbol{\Phi}^H(\mathbf{x} - \boldsymbol{\Phi}\boldsymbol{\gamma})$,
(4) Update the solution: $\gamma_{i,p+1} = \text{soft}(u_{i,p}, \frac{\kappa}{\beta_p})$, $i = 1, \ldots, N_dN_s$
Until stopping criterion holds.
Output: Return $\boldsymbol{\gamma} = \boldsymbol{\gamma}_p$.

of convergence which is proven to be significantly better, both theoretically and practically [22]. Let $J(\boldsymbol{\gamma}) = \frac{1}{2}\|\mathbf{x}-\boldsymbol{\Phi}\boldsymbol{\gamma}\|_2^2+\kappa\|\boldsymbol{\gamma}\|_1$. The principle behind FISTA is to iteratively form quadratic approximations $Q(\boldsymbol{\gamma}, \bar{\boldsymbol{\gamma}})$ to $J(\boldsymbol{\gamma})$ around a carefully chosen point $\bar{\boldsymbol{\gamma}}$, and to minimize $Q(\boldsymbol{\gamma}, \bar{\boldsymbol{\gamma}})$ rather than the original cost function $J(\boldsymbol{\gamma})$. For convenance, let $g(\boldsymbol{\gamma}) = \|\boldsymbol{\gamma}\|_1$ and $F(\boldsymbol{\gamma}) = \frac{1}{2}\|\mathbf{x} - \boldsymbol{\Phi}\boldsymbol{\gamma}\|_2^2$. Note that $\triangledown F(\boldsymbol{\gamma}) = \boldsymbol{\Phi}^H(\boldsymbol{\Phi}\boldsymbol{\gamma} - \mathbf{x})$ is Lipschitz continuous with Lipschitz constant $\beta_F = \|\boldsymbol{\Phi}\|^2$. Define $Q(\boldsymbol{\gamma}, \bar{\boldsymbol{\gamma}})$ as:

$$Q(\boldsymbol{\gamma}, \bar{\boldsymbol{\gamma}}) = F(\bar{\boldsymbol{\gamma}}) + \langle\triangledown F(\bar{\boldsymbol{\gamma}}), \boldsymbol{\gamma} - \bar{\boldsymbol{\gamma}}\rangle + \frac{\beta_F}{2}\|\boldsymbol{\gamma} - \bar{\boldsymbol{\gamma}}\|_2^2 + \kappa g(\boldsymbol{\gamma}). \quad (17)$$

The main difference between the FISTA and the IST algorithm is that the iterative shrinkage operator soft is not employed on the previous point $\boldsymbol{\gamma}_{p-1}$, but rather at the point $\bar{\boldsymbol{\gamma}}_p$ which uses a very specific linear combination of the previous two points $\{\boldsymbol{\gamma}_{p-1}, \boldsymbol{\gamma}_{p-2}\}$. The pseudo code of FISTA with backtracking line-search strategy is summarized in Table 3.

4.4. FOCUSS

The principle of the FOCUSS algorithm is to use the weighted l_2-norm minimization to operate recursive adjustments to the weighting matrix until most elements of the solution are close to zero [23]. The basic form of the FOCUSS algorithm is composed of the following two steps:

$$\text{Step1}: \quad \mathbf{W}_p = \text{diag}(|\boldsymbol{\gamma}_{p-1}|), \quad (18)$$

$$\text{Step2}: \quad \boldsymbol{\gamma}_p = \mathbf{W}_p(\boldsymbol{\Phi}\mathbf{W}_p)^\dagger\mathbf{x}, \quad (19)$$

where $(\mathbf{A})^\dagger$ denotes the pseudo inverse operation of matrix $\mathbf{A}$. The pseudo code of FOCUSS algorithm in [9] is summarized in Table 4.

Table 3. FISTA with backtracking line-search strategy in [22].

Initialization: Choose initial guess $\boldsymbol{\gamma}_0$, $\beta_0 > 0$, and some $\eta > 1$. Set $\bar{\boldsymbol{\gamma}}_1 = \boldsymbol{\gamma}_0$, $t_1 = 1$, the iteration index $p = 1$.
Repeat: (1) Find the smallest nonnegative integers i_p such that with $\bar{\beta} = \eta^{i_p}\beta_{p-1}$ $F(\text{soft}(\bar{\boldsymbol{\gamma}}_p, \frac{\kappa}{\bar{\beta}})) \le Q(\text{soft}(\bar{\boldsymbol{\gamma}}_p, \frac{\kappa}{\bar{\beta}}), \bar{\boldsymbol{\gamma}})$. (2) Set $\beta_p = \eta^{i_p}\beta_{p-1}$ and compute $\mathbf{u}_p = \boldsymbol{\gamma}_p - \frac{1}{\beta_p}\nabla_{\boldsymbol{\gamma}}\{\frac{1}{2}\|\mathbf{x} - \boldsymbol{\Phi}\boldsymbol{\gamma}\|_2^2\}$, $\gamma_{i,p+1} = \text{soft}(u_{i,p}, \frac{\kappa}{\beta_p})$, $i = 1, \ldots, N_d N_s$ $t_{p+1} = \frac{\sqrt{1+\sqrt{1+4t_p^2}}}{2}$, $\bar{\boldsymbol{\gamma}}_{p+1} = \boldsymbol{\gamma}_p + \frac{t_p - 1}{t_{p+1}}(\boldsymbol{\gamma}_{p+1} - \boldsymbol{\gamma}_p)$. **Until stopping criterion holds.**
Output: Return $\boldsymbol{\gamma} = \boldsymbol{\gamma}_p$.

Table 4. FOCUSS algorithm in [9].

Initialization: $\boldsymbol{\gamma}_0 = \boldsymbol{\Phi}^H\mathbf{x} + \delta_{\text{foc}}$, where δ_{foc} is the loading level, $\mathbf{W}_0 = \text{diag}(\lvert\boldsymbol{\gamma}_0\rvert)$, $\Gamma = \{i, 1 \le i \le N_d N_s\}$, and the iteration index $p = 1$.
Repeat: (1) $\boldsymbol{\gamma}_p\lvert_\Gamma = \mathbf{W}_p(\boldsymbol{\Phi}\lvert_\Gamma \mathbf{W}_p)^\dagger\mathbf{x}$ (2) Update the adaptive subspace set: $\Gamma = \arg(\lvert\gamma_{i,p} \ge Th\rvert)$, $1 \le i \le N_d N_s$, where Th denotes the threshold. (3) Smooth the weighting matrix using the nearest eight cells around center $i, i \in \Gamma$: $\mathbf{W}_{p+1} = \text{diag}\big[\frac{1}{9}(\lvert\boldsymbol{\gamma}_{i,p}\rvert + \sum_{i_k}\lvert\boldsymbol{\gamma}_{i_k,p}\rvert)\big]$, where $i \in \Gamma$ and i_k is the nearest eight cells. **Until stopping criterion holds.**
Output: Return $\boldsymbol{\gamma} = \boldsymbol{\gamma}_p$.

4.5. SL0

The main idea of SL0 is to approximate the discontinuous function l_0-norm by a suitable continuous one, and minimize it by means of a minimization algorithm for continuous functions (e.g., steepest descent method) [24]. Assume a parameter ϱ determining the quality

Table 5. SL0 algorithm in [24].

Initialization: $\boldsymbol{\gamma}_0 = \boldsymbol{\Phi}^{\dagger}\mathbf{x}$, the iteration index $p = 1$, Choose a suitable decreasing sequence for ϱ, $[\varrho_1, \varrho_2, \ldots]$.
Repeat: (1) Let $\varrho = \varrho_p$, and $\mathbf{b} = \boldsymbol{\gamma}_p \odot \exp(-\frac{\boldsymbol{\gamma}_p^2}{2\varrho^2})$. (2) Update the solution: Repeat K times: $\bar{\boldsymbol{\gamma}}_{p+1} = \boldsymbol{\gamma}_p - \mu_{\mathrm{SL0}}\mathbf{b}$, where μ_{SL0} is a small positive constant. $\boldsymbol{\gamma}_{p+1} = \bar{\boldsymbol{\gamma}}_{p+1} - \boldsymbol{\Phi}^{\dagger}(\boldsymbol{\Phi}\bar{\boldsymbol{\gamma}}_{p+1} - \mathbf{x})$. **Until stopping criterion holds.**
Output: Return $\boldsymbol{\gamma} = \boldsymbol{\gamma}_p$.

of the approximation of the continuous function to $\|\boldsymbol{\gamma}\|_0$. In SL0, the continuous function is considered to be one of the family of functions:

$$e_{\varrho}(\gamma) = \exp\left(-\frac{\gamma^2}{2\varrho^2}\right). \tag{20}$$

The pseudo code of SL0 algorithm in [24] is summarized in Table 5.

5. PERFORMANCE ASSESSMENT

In this section, we try to answer the three questions presented in the Introduction by measuring the performance of STAP algorithms based on those five fast SR techniques with simulated data. Simulation results will show the curves of the SINR performance against Doppler frequency, and the probability of detection P_d versus the signal-to-noise (SNR) ratio. The output SINR is defined by

$$\mathrm{SINR} = 10\log_{10}\left\{\frac{\left|\hat{\mathbf{w}}^H\mathbf{s}\right|^2}{|\hat{\mathbf{w}}^H\mathbf{R}\hat{\mathbf{w}}|}\right\}, \tag{21}$$

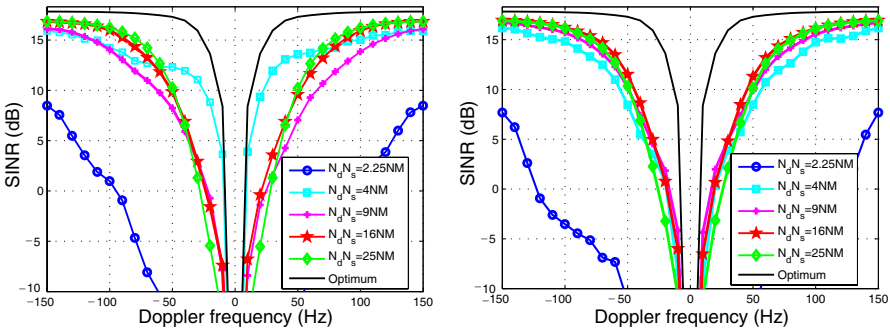
where $\mathbf{R}$ is the true clutter-noise covariance matrix.

5.1. Impact of the Dictionary

Assume a simulated scenario with the following parameters: sideway-looking ULA, 8 elements with half wavelength spacing, 8 pulses in one CPI, uniform transmit taper, carrier frequency 450 MHz, pulse repetition frequency (PRF) setting to 300 Hz, platform velocity of

50 m/s and height of 9000 m, clutter-to-noise-ratio (CNR) of 50 dB, the target located at 0° azimuth with signal-to-noise-ratio (SNR) 0 dB, and the noise power is unitary. In the following simulations, except for special illustrations, parameters used for all algorithms are set as: for the SL0, $K = 3$, $\varrho_{\min} = 0\,\text{dB}$ to noise level, $\eta_{\varrho} = 0.9$, $\mu_{\text{SL0}} = 2$; for the FISTA, $\beta_0 = 1$, $\eta_{\kappa} = 0.9$, $\eta_{\beta} = 1.5$; for the SpaRSA, $\eta_{\kappa} = 0.9$; for the CoSaMP, $\beta = 2$, $N_n = 20$; for the FOCUSS, $\delta_{\text{foc}} = 0\,\text{dB}$ to noise level, $Th = 15\,\text{dB}$ to noise level. The stopping criterion for the FISTA, SpaRSA, CoSaMP and FOCUSS algorithms is decided by that the relative change in $\boldsymbol{\gamma}$ between consecutive iterations being sufficiently small (here, it sets the tolerance 10^{-4}). The stopping criterion for the SL0 is decided by $\varrho_{\min}$. The optimum STAP performance is computed by using the true clutter-noise covariance matrix.

In the first example, we show the SINR performance against the target Doppler frequency with different sizes of the space-time steering dictionary, as illustrated in Figs. 1(a)–(e). From the figures, we see that: (i) when N_dN_s is too small, e.g., $N_dN_s = 2.25NM$, the SINR performance of all algorithms is very low and can not suppress the clutter. (ii) When N_dN_s is too large, e.g., $N_dN_s = 25NM$, the SINR performance of all algorithms exhibits some degradation. This is because the larger the size of dictionary, the higher correlation the columns of the dictionary. These five fast SR algorithms are not good at dealing with the dictionary with high correlation property. (iii) When N_dN_s is at moderate sizes, e.g., $N_dN_s = 4NM - 16NM$, the SINR performance of the SL0, the FISTA, and the SpaRSA algorithms is acceptable for STAP problems and shows a relative robust characteristic against the size of dictionary. However, the SINR performance of the CoSaMP algorithm significantly degrades with the size becoming large. This can be roughly understood to be due to the CoSaMP algorithm requiring the dictionary to satisfy the RIP, which



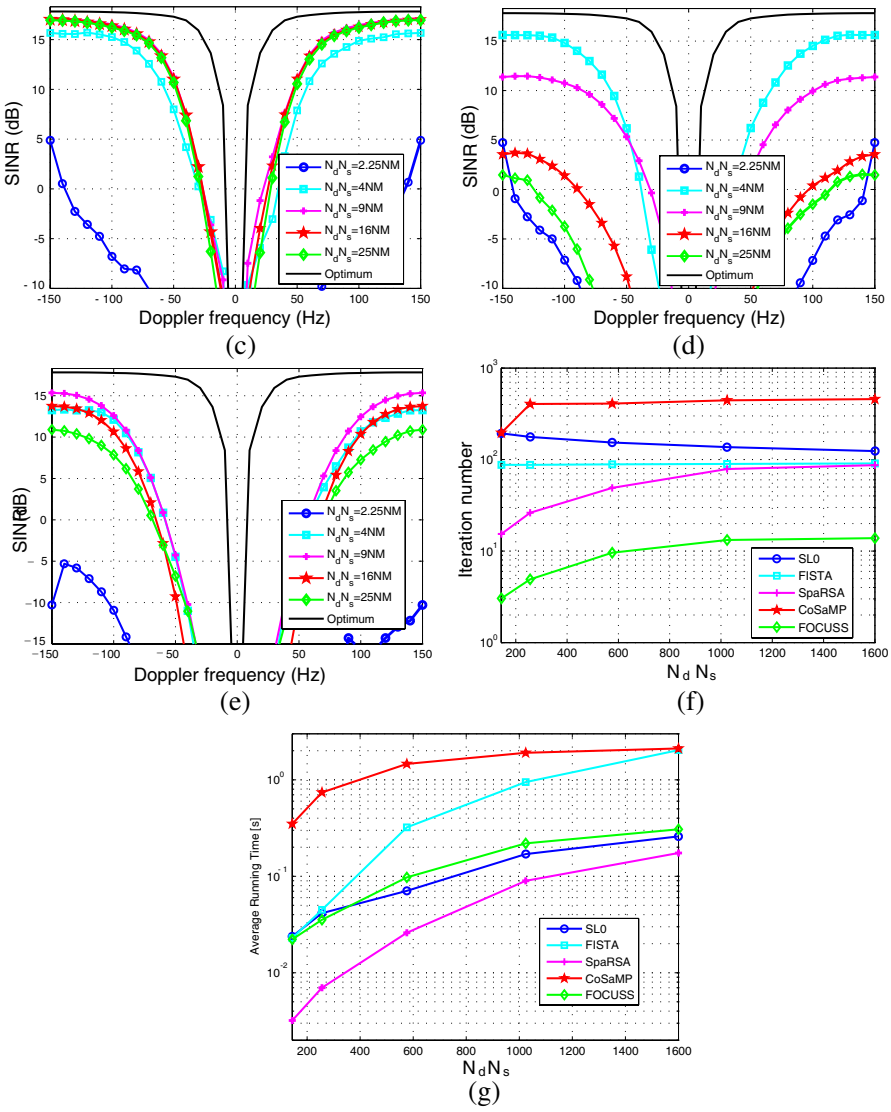


Figure 1. Impacts of sizes of dictionary on SINR performance against Doppler frequency with 6 snapshots and target Doppler frequency space from −150 Hz to 150 Hz. (a) SL0. (b) FISTA. (c) SpaRSA. (d) CoSaMP. (e) FOCUSS. (f) Iteration number versus sizes of dictionary for all algorithms. (g) Average running time versus sizes of dictionary for estimating one angle-Doppler profile.

does not ensure in STAP problems.

Figure 1(f) depicts the iteration number versus the sizes of space-time steering dictionary for all algorithms. It is found that the iteration number of all algorithm trends to a fixed value as the increasing of N_dN_s. In fact, the iteration number of each algorithm is depended on the stopping criterion. While in the simulations, the stopping criterion is kept the same for different sizes of N_dN_s. Thus, the curves illustrate that the solution among iterations will trend to a very small relative change in the case of a fixed stopping criterion and a size of N_dN_s over some value. Fig. 1(g) plots the average running time versus sizes of dictionary for estimating one angle-Doppler profile. Here, our simulations are operated on a standard desktop computer with a 2.60 GHz CPU (dual core with Matlab's multithreading option enabled) and 2 GB of memory. The curves show that the SpaRSA algorithm uses the shortest time and the CoSaMP algorithm uses the longest time compared with other algorithms. Considering both the computational complexity and the SINR performance, we observe that the SL0 and the SpaRSA algorithms are suitable for STAP problems. Besides, it is should be noted that the FISTA algorithm is also an acceptable choice for STAP problems because it can achieve a good SINR performance when $N_dN_s = 4NM$ corresponding to a short time. For comparison, in the following simulations, we use the best size of dictionary for different algorithms: $N_dN_s = 16NM$ for the SL0 and the FISTA; $N_dN_s = 9NM$ for the SpaRSA and the FOCUSS; $N_dN_s = 4NM$ for the CoSaMP.

5.2. Parameters Setting

In this subsection, we detail parameters used (not including stopping criterion) in above five fast SR algorithms for airborne radar applications, as shown in Table 6. From the table, we see that the parameters used in the SpaRSA and the FISTA algorithms are easier to set than those of other algorithms. In the CoSaMP algorithm, it is hard to obtain the exact number of non-zero elements in the angle-Doppler profile. Fig. 2 shows the SINR performance against Doppler frequency at different values of $N_n = 10, 20, 30$. It is illustrated that the CoSaMP algorithm is sensitive to the values of N_n. Even though a roughly range of non-zero elements can be obtained by using the clutter rank estimated by existing approaches [1], the CoSaMP can not ensure a good performance due to the sensitiveness to N_n. In the SL0 algorithm, $\varrho_{\min}$ is usually set to the noise level. The simulation results with different values of $\varrho_{\min}$ (-10, 0, 10, 15 dB to the noise level) are plotted in Fig. 3. The curves show that it can obtain a good performance when $\varrho_{\min}$ is in the range of -10 dB to 10 dB to the noise

Table 6. Parameters setting for different SR algorithms.

Algorithm	Parameters Used	Remarks
CoSaMP	β, N_n	A experience value of β is 2; N_n has to be estimated from the special scenarios.
SpaRSA	η_κ	A experience range of η_κ is $0.7 - 0.9$.
FISTA	β_0, η_κ, η_β	A experience range of η_κ is $0.7 - 0.9$; Experience values of β_0 and η_β are 1 and 1.5.
FOCUSS	δ_{foc}, Th	Both δ_{foc} and Th need to be estimated from scenarios.
SL0	$\varrho_{\min}$, η_ϱ, μ_{SL0}, K	Experience values of K and η_ϱ are 3 and $0.7 - 0.9$; μ_{SL0} is the step-size, 2 is acceptable in simulations; $\varrho_{\min}$ has to be estimated from scenarios.

level. In the FOCUSS algorithm, two parameters δ_{foc} and Th need to be estimated from scenarios. The former is correlated with the noise level, and the latter is decided by the clutter-to-noise ratio (CNR), which is critical to the performance, as shown in Fig. 4 (here, fixed δ_{foc} to the noise level, and Th setting to 0 dB, 10 dB, 15 dB, 20 dB and 23 dB to the noise level). Furthermore, the computational complexity of FOCUSS algorithm depends on the the size of Th. Overall, parameters setting of the CoSaMP and the FOCUSS algorithms is much difficult than other algorithms.

5.3. Impact of ICM

In this subsection, we evaluate the impacts on the performance with different ICM for all algorithms. For STAP problem in airborne radar systems, the ICM can be formulated as a model that proposed by J. Ward in [1]. The temporal autocorrelation of the fluctuations is Gaussian in shape with the form:

$$\zeta(m) = \exp\left\{ -\frac{8\pi^2\sigma_v^2 T_r^2}{\lambda_c^2} m^2 \right\}, \tag{22}$$

where T_r is the the pulse repetition interval and σ_v is the velocity standard deviation. Here, we consider two different ICM cases with

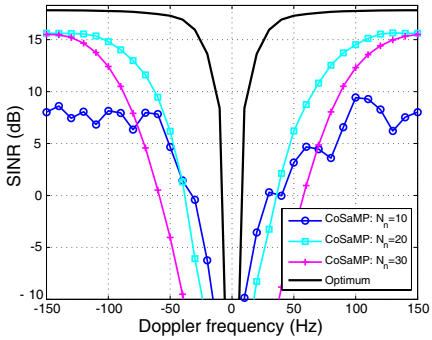


Figure 2. Impacts of N_n on SINR performance for the CoSaMP.

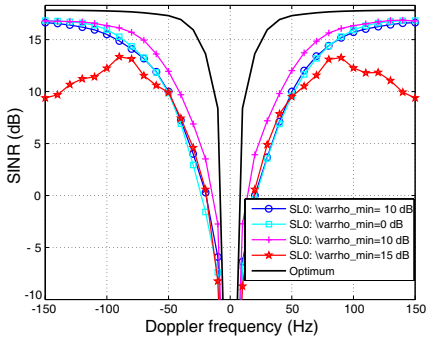


Figure 3. Impacts of $\varrho_{\min}$ on SINR performance for the SL0.

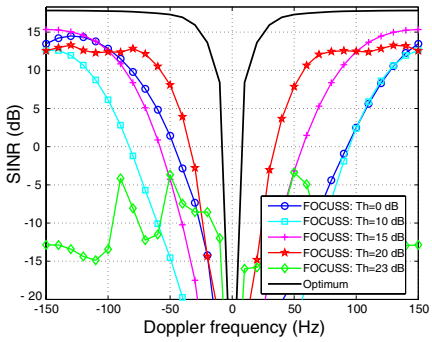


Figure 4. Impacts of Th on SINR performance for the FOCUSS.

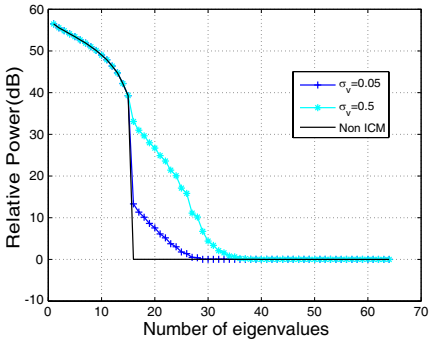


Figure 5. Clutter eigenspectrum with different ICM.

$\sigma_v = 0.05$, and $\sigma_v = 0.5$. Fig. 5 shows the clutter eigenspectrum about these two cases, which is computed by taking an eigenvalue decomposition to the true clutter-noise covariance matrix. It is found that as σ_v increases, the tails of the eigenspectrum become larger as the clutter rank of the covariance increases, corresponding to the reduction of sparsity degree of the angle-Doppler profile. In Fig. 6(a), we present the detection performance (P_d) versus SNR performance for all algorithms using 6 snapshots as the training data. The false alarm rate P_{fa} is set to 10^{-3} and for simulation purpose the threshold and P_d estimates are based on $10,000$ samples. We suppose the target is injected in the the boresight with Doppler frequency 100 Hz (here, we

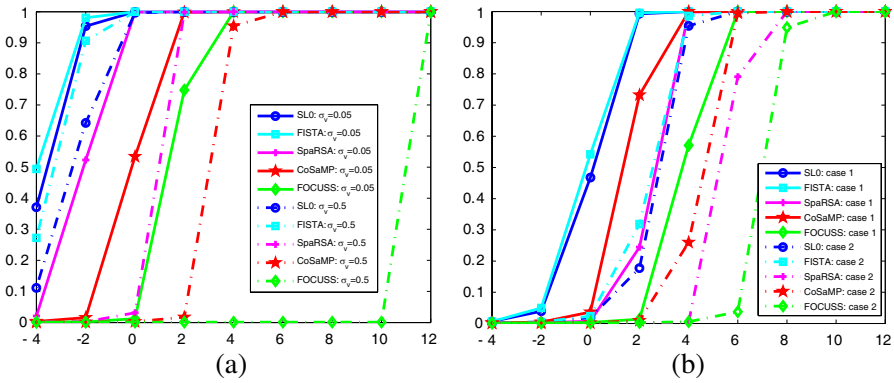


Figure 6. Detection performance for different scenarios. (a) Impacts of ICM on P_d versus SNR. (b) Impacts of array errors on P_d versus SNR.

do not consider the target spreading in multiple range bins and assume the target is only in one range bin). It is illustrated that: (i) the FISTA, the SL0 and the SpaRSA algorithms are more suitable for scenarios with low SNR target than other algorithms. (ii) the detection performance P_d of all algorithms becomes worse with the increasing of ICM. The FOCUSS algorithm exhibits the steepest performance degradation versus the ICM. In fact, this is caused by the parameter Th fixed in these two situations. When $\sigma_v = 0.5$ corresponding to very serious ICM, the sparsity degree of the angle-Doppler profile is significantly reduced. In this case, Th must be set smaller to ensure much more non-zeros elements. Otherwise, the estimated angle-Doppler profile will drop many significant clutter components resulting in performance degradation. Besides, the CoSaMP shows the sensitiveness to the ICM. This is easily understood that the performance of the CoSaMP relies on the estimation of the number of non-zero elements in the angle-Doppler profile.

5.4. Impact of Array Errors

In this subsection, we access the impacts on detection performance with different array errors. Here, we only consider the angle-independent array errors described in [2], i.e., the amplitude and phase errors are modeled as a narrowband case as follows:

$$p(\delta_{\epsilon_a}) = \begin{cases} \dfrac{1}{\Delta\epsilon_a} & \text{for} \quad 0 \le \delta_{\epsilon_a} \le \Delta\epsilon_a \\ 0 & \text{elsewhere} \end{cases}, \tag{23}$$

and

$$p(\delta_{\epsilon_p}) = \begin{cases} \dfrac{1}{\Delta\epsilon_p} & \text{for} \quad -\dfrac{\Delta\epsilon_p}{2} \le \delta_{\epsilon_p} \le \dfrac{\Delta\epsilon_p}{2} \\ 0 & \text{elsewhere} \end{cases}, \tag{24}$$

where $p(\delta_{\epsilon_a})$ and $p(\delta_{\epsilon_p})$ are the pdfs (uniform) associated with the amplitude and phase errors respectively. In the following examples, we set: case 1, $\Delta\epsilon_a = 0.01$ and $\Delta\epsilon_p = 2^\circ$, and case 2, $\Delta\epsilon_a = 0.02$ and $\Delta\epsilon_p = 5^\circ$. The other parameters, except for ICM, are the same as that in Subsection 5.3. From Fig. 6(b), we observe that (i) the detection performance of SL0 and FISTA algorithms is better than that of other algorithms for low SNR target; (ii) the detection performance degrades with the array errors increasing for all algorithms. By inspecting Figs. 6(a) and 6(b), it should be noted that the impacts of array errors are larger than that of the ICM for SR-STAP type algorithms.

6. CONCLUSION AND DISCUSSION

In this paper, extensive experiments with different practical considerations are performed to evaluate the performance and discuss the difficulties of the parameters setting of the STAP algorithms based on fast SR techniques. There is no clear winner that always obtain the best performance and the lowest computational complexity. However, we can make the following conclusions to answer the questions presented in the Introduction from the experiments: (i) the STAP algorithms based on fast SR techniques provide an efficient tool to solve the problem in very short snapshots and exhibit great potential for application in nonhomogeneous environments. (ii) Considering the difficulties of parameters setting, the computational complexity and the performance, the STAP algorithms based on the SL0, the FISTA and the SpaRSA are more suitable for dealing with STAP problems in airborne radar systems.

ACKNOWLEDGMENT

This work is funded in part by the National Natural Science Foundation of China under Grant Nos. 61171133 and 61025006.

REFERENCES

1. Ward, J., "Space-time adaptive processing for airborne radar," *Technical Report 1015*, MIT Lincoln Laboratory, Lexington, MA, Dec. 1994.

2. Guerci, J. R., *Space-time Adaptive Processing for Radar*, Artech House, 2003.
3. Melvin, W. L., "A STAP overview," *IEEE Aerosp. Electron. Syst. Mag.*, Vol. 19, No. 1, 19–35, 2004.
4. Aïssa, B., M. Barkat, B. Atrouz, M. C. E. Yagoub, and M. A. Habib, "An adaptive reduced rank STAP selection with staggered PRF, effect of array dimensionality," *Progress In Electromagnetics Research C*, Vol. 6, 37–52, 2009.
5. Gong, Q. Y. and Z. D. Zhu, "Study STAP algorithm on interference target detect under nonhomogeneous environment," *Progress In Electromagnetics Research*, Vol. 99, 211–224, 2009.
6. Maria, S. and J. J. Fuchs, "Application of the global matched filter to STAP data an efficient algorithmic approach," *Proc. IEEE Int. Conf. Acoust. Speech and Signal Process.*, 14–19, 2006.
7. Selesnick, I. W., S. U. Pillai, K. Y. Li, and B. Himed, "Angle-Doppler processing using sparse regularization," *Proc. IEEE Int. Conf. Acoust. Speech and Signal Process.*, 2750–2753, 2010.
8. Sun, K., H. Zhang, G. Li, H. Meng, and X. Wang, "A novel STAP algorithm using sparse recovery technique," *Proc. IGARSS*, 336–339, 2009.
9. Sun, K., H. Meng, Y. Wang, and X. Wang, "Direct data domain STAP using sparse representation of clutter spectrum," *Signal Process.*, Vol. 91, No. 9, 2222–2236, 2011.
10. Parker, J. T. and L. C. Potter, "A Bayesian perspective on sparse regularization for STAP post-processing," *Proc. IEEE Radar Conf.*, 1471–1475, May 2010.
11. Yang, Z., R. C. de Lamare, and X. Li, "L_1-regularized STAP algorithms with a generalized sidelobe canceler architecture for airborne radar," *IEEE Trans. on Signal Process.*, Vol. 60, No. 2, 674–686, 2012.
12. Yang, Z., R. C. de Lamare, and X. Li, "Sparsity-aware STAP algorithms for airborne radar based on conjugate gradient techniques," *Proc. Sensor Signal Process. for Defence Conf.*, London, UK, 2011.
13. Yang, Z., R. C. de Lamare, and X. Li, "L_1 regularized STAP algorithm with a generalized sidelobe canceler architecture for airborne radar," *Proc. IEEE Workshop on Statist. Signal Process.*, 329–332, 2011.
14. Liu, Y. and Q. Wan, "Total difference based partial sparse LCMV beamformer," *Progress In Electromagnetics Research Letters*, Vol. 18, 97–103, 2010.

15. Zhang, Y., Q. Wan, and A.-M. Huang, "Localization of narrow band sources in the presence of mutual coupling via sparse solution finding," *Progress In Electromagnetics Research*, Vol. 86, 243–257, 2008.
16. Yang, M. and G. Zhang, "Compressive sensing based parameter estimation for monostatic MIMO noise radar," *Progress In Electromagnetics Research Letters*, Vol. 30, 133–143, 2012.
17. Ke, W. and L. Wu, "Sparsity-based multi-target direct positioning algorithm based on joint-sparse recovery," *Progress In Electromagnetics Research C*, Vol. 27, 99–114, 2012.
18. Gui, G., N. Zheng, N. Wang, A. Mehbodniya, and F. Adachi, "Compressive estimation of cluster-sparse channels," *Progress In Electromagnetics Research C*, Vol. 24, 251–263, 2011.
19. Needell, D. and J. Tropp, "CoSaMP: Iterative signal recovery from incomplete and inaccurate samples," *Appl. Comp. Harmonic Anal.*, Vol. 26, 301–321, 2008.
20. Tropp, J. A. and J. Wright, "Computational methods for sparse solution of linear inverse problems," *Proc. of IEEE*, Vol. 98, No. 6, 948–958, 2010.
21. Wright, S. J., R. D. Nowak, and M. A. T. Figueiredo, "Sparse reconstruction by separable approximation," *IEEE Trans. on Signal Process.*, Vol. 57, No. 7, 2479–2493, 2009.
22. Beck, A. and M. Teboulle, "A fast iterative shrinkage-thresholding algorithm for linear inverse problems," *SIAM J. Imag. Sci.*, Vol. 2, No. 1, 183–202, 2009.
23. Gorodnitsky, I. F. and B. D. Rao, "Sparse signal reconstruction from limited data using FOCUSS: A re-weighted minimum norm algorithm," *IEEE Trans. on Signal Process.*, Vol. 45, No. 3, 600–616, 1997.
24. Hosein Mohimani, G., M. Babaie-Zadeh, and C. Jutten, "A fast approach for overcomplete sparse decomposition based on smoothed l_0 norm," *IEEE Trans. on Signal Process.*, Vol. 57, No. 1, 289–301, 2009.