Correspondence

Low-Complexity Reduced-Dimension Space–Time Adaptive Processing for Navigation Receivers

In this paper, we propose a low-complexity adaptive reduced-rank interference suppression scheme based on alternating low-rank decomposition techniques for navigation receivers. The proposed scheme, which employs a generalized sidelobe canceler structure, makes use of a projection matrix for rank reduction followed by a reduced-dimension receive filter. An alternating optimization strategy based on recursive least squares is devised to compute the basis vectors of the projection matrix and the receive filter adaptively. Simulation results show that the proposed algorithm achieves a better performance than existing methods with a reduced computational complexity.

I. INTRODUCTION

Satellite-based navigation systems, such as GPS and Galileo [1], have been widely used in civilian applications, including agriculture, aviation, land-vehicle navigation, and marine-navigation [1]–[3], to provide location, velocity, and time information for users. However, due to the use of spread spectrum modulation with low power level at the satellite transmitter in these systems, the navigation signal at the terrestrial receiver remains vulnerable to interferences from wideband and narrowband jammers. Consequently, the design of navigation receivers must be robust in the face of high-power interferences (often over 30 dB higher than the navigation signal) and noise. In the case of receivers equipped with multiple antennas, advanced space–time (ST) processing techniques can be employed to reliably receive the desired signal and simultaneously suppress a large number of high-power wideband and narrowband jamming signals before the despreading operation.

In ST approaches, a set of received spatial vectors collected over a finite temporal window are stacked together

Manuscript received November 18, 2017; revised January 22, 2018; released for publication March 21, 2018. Date of publication April 23, 2018; date of current version December 5, 2018.

DOI. No. 10.1109/TAES.2018.2829380

Refereeing to this contribution was handled by A. G. Dempster.

The work of Y. Cai was supported in part by the National Natural Science Foundation of China under Grant 61471319 and in part by the Fundamental Research Funds for the Central Universities. The work of M. Zhao was supported by the National Natural Science Foundation of China under Grant 91538103.

Authors' addresses: Y. Cai, X. Wu, and M. Zhao are with the College of Information Science and Electronic Engineering, Zhejiang University, Hangzhou 310027, China, E-mail: (ylcai521@gmail.com; wxm075910@zju.edu.cn; mjzhao@zju.edu.cn); R. C. de Lamare is with CETUC-PUC-Rio, Rio de Janeiro 22453-900, Brazil, and also with the Communications Research Group, Department of Electronics, University of York, York YO10 5DD, U.K., E-mail: (rcdl500@ohm.york.ac.uk); B. Champagne is with the Department of Electrical and Computer Engineering, McGill University, Montreal, QC H3A 0E9, Canada, E-mail: (benoit.champagne@mcgill.ca). *(Corresponding author is Yunlong Cai.)*

into an augmented high-dimensional data vector. The latter is operated upon by a receive filter of same dimension whose design usually involves optimization of a chosen cost function under various constraints. ST processing can be implemented at a reduced computational complexity with adaptive algorithms [4], which also have the ability to track slow variations in the signals. In particular, adaptive algorithms can continually adjust the filter weights operating on the augmented data vector, allowing for improved detection of the desired signal in nonstationary environments. However, one problem for the conventional ST, i.e., full-rank adaptive algorithms is that their convergence performance depends on the eigenvalue spread of the received data covariance matrix [5]–[8]. This condition is often worse when an ST processor with a large number of adaptive weights is employed, as needed to deal with a large-dimensional augmented data vector. In this context, reduced-rank signal processing, which projects the augmented data vector onto a lower dimensional subspace and performs filtering, estimation, and optimization within this subspace, can provide faster convergence performance and increased robustness against interference as compared to conventional adaptive ST processing for navigation receivers.

A number of reduced-rank techniques have been developed to design the subspace projection matrix and the reduced-rank receive filter [5]–[18]. Among the early schemes are the eigendecomposition-based algorithms [5], [6], the multistage Wiener filter (MSWF) [7], [8] and the auxiliary vector filtering (AVF) algorithms [9], all requiring high computational complexity to construct the transformation matrix. A strategy based on the joint and iterative optimization (JIO) of a subspace projection matrix and a reduced-rank receive filter has been reported in [10]–[15]. The resulting algorithms have to jointly adjust a large number of parameters for adaptation, and are still characterized by high computational complexity. The joint preprocessing, decimation, and filtering scheme [16]–[19], which constructs the transformation matrix based on shifted versions of a single basis vector, can significantly reduce the complexity. In addition, a low-complexity hierarchical recursive least squares (RLS) algorithm has been proposed in [20] to increase the convergence rate of the original RLS algorithm. However, there is still a need to improve the convergence performance while reducing the complexity of ST processing in navigation applications.

In this paper, we propose a low-complexity alternating low-rank decomposition (ALRD) algorithm for ST adaptive interference suppression in navigation receivers. Simulation results show that the performance of the proposed algorithm is superior to that of the ST full-rank and reduced-rank adaptive techniques commonly used in navigation systems. The main contributions of this paper are summarized as follows.

1) Unlike the conventional reduced-rank schemes, the proposed algorithm processes the augmented data vector using a projection matrix comprised of a set of basis vector obtained in a different way, where all the basis vectors are jointly optimized together with the reduced-rank filter.
2) We consider a generalized sidelobe canceler (GSC) structure, where the reduced-rank receive filtering is implemented in the auxiliary branch for interference suppression. An alternating RLS-type algorithm is developed to jointly update the basis vectors forming the projection matrix and the reduced-rank receive filter.
3) Detailed computational complexity and convergence analyses are carried out for the proposed adaptive reduced-dimension algorithm.

The paper is organized as follows. In Section II, we outline the system model and the problem statement. The proposed ALRD algorithm is presented in Section III along with complexity and convergence analysis. In Section IV, we illustrate and discuss the simulation results. Finally, Section V concludes this paper.

II. SYSTEM MODEL AND PROBLEM STATEMENT

In navigation antijamming systems, an ST processor with M identical omnidirectional antenna elements and L filter coefficients per antenna is considered, as illustrated in Fig. 1. After downconversion and downsampling at the chip rate, the $M \times 1$ received data vector $\mathbf{r}(i) = [r_1(i), \ldots, r_M(i)]^T$ at the ith time instant can be expressed as

$$\mathbf{r}(i) = s(i)\mathbf{a}(\theta_0) + \sum_{j=1}^{K_w} c_j(i)\mathbf{a}(\theta_j) + \sum_{l=K_w+1}^{K_w+K_n} c_l(i)\mathbf{a}(\theta_l) + \mathbf{n}(i). \tag{1}$$

In this expression, $s(i)$ denotes the desired BPSK-modulated navigation signal from the satellite, which is spread by a specific code. The signals $c_j(i)$ and $c_l(i)$ represent the interfering signal from the jth wideband jammer and the lth narrowband jammer, respectively. We assume that the navigation signal $s(i)$, and the K_w wideband jamming signals $c_j(i)$ as well as K_n narrowband jamming signals $c_l(i)$ are mutually independent. In (1), the parameter θ_0 denotes the direction of arrival (DOA) corresponding to the desired GPS satellite signal, which is known beforehand by the receiver, while angles θ_j and θ_l denote the DOAs of the wideband and narrowband jammers, respectively. Vector $\mathbf{a}(\theta)$ describes the normalized steering vector with DOA θ as an argument. In the case of a uniform linear array (ULA), with λ_c denoting the wavelength at the operating frequency and $d = \lambda_c/2$ the interelement spacing of the ULA, the corresponding $M \times 1$ normalized steering vector is given by[1]

$$\begin{aligned}\mathbf{a}(\theta) &= \frac{1}{\sqrt{M}}\left[1, e^{-j2\pi\frac{d\cos\theta}{\lambda_c}}, \ldots, e^{-j2\pi\frac{(M-1)d\cos\theta}{\lambda_c}}\right]^T \\ &= \frac{1}{\sqrt{M}}\left[1, e^{-j\pi\cos\theta}, \ldots, e^{-j\pi(M-1)\cos\theta}\right]^T.\end{aligned} \tag{2}$$

[1] A ULA is considered in order to simplify the presentation, while generalization to other antenna configurations is straightforward.

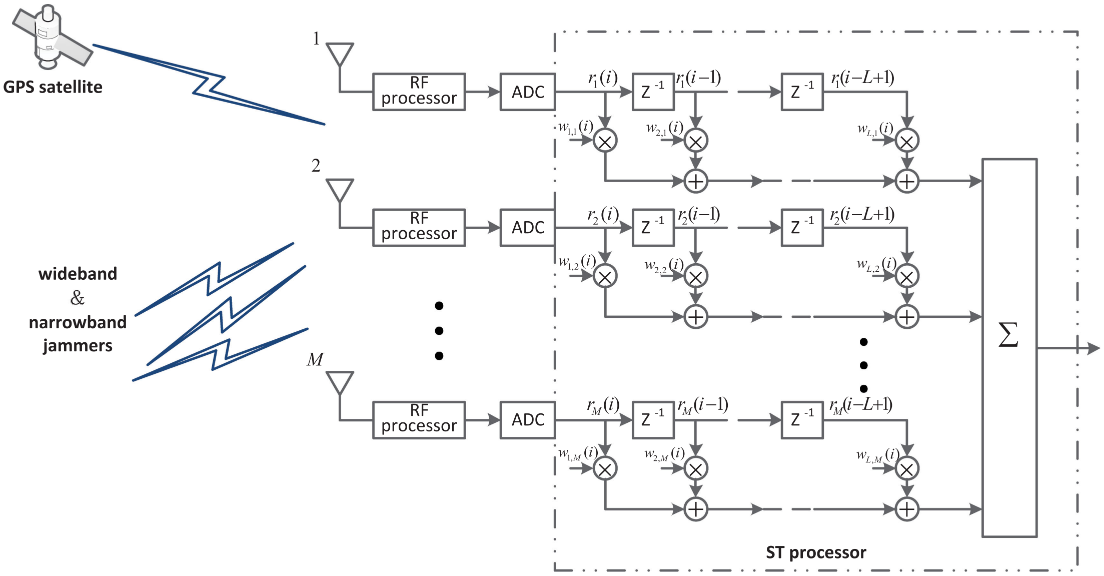

Fig. 1. Block diagram of the ST adaptive processing structure.

The term $\mathbf{n}(i) \in \mathbb{C}^{M\times 1}$ is the additive noise, which is modeled as an independent and identically distributed sequence of spatially white Gaussian random vector with zero-mean and covariance matrix $\mathbb{E}\{\mathbf{n}(i)\mathbf{n}^H(i)\} = \sigma_n^2 \mathbf{I}_M$, where σ_n^2 denotes the noise variance, $\mathbf{I}_M$ is an identity matrix of order M, $\mathbb{E}\{\cdot\}$ stands for expected value, and $(\cdot)^H$ stands for the Hermitian transpose operation. The input signal-to-noise ratio (SNR) specific for each antenna element is defined as $\text{SNR} = \frac{\mathbb{E}\{|s(i)|^2\}}{M\sigma_n^2}$.

In ST processing schemes, the augmented $ML \times 1$ snapshot vector $\mathbf{x}(i)$ is obtained by stacking L consecutive data vectors, and is expressed as

$$\mathbf{x}(i) = \left[\mathbf{r}^T(i), \mathbf{r}^T(i-1), \ldots, \mathbf{r}^T(i-L+1)\right]^T. \quad (3)$$

The full-rank ST processor is equivalent to designing an augmented spatio-temporal filter $\mathbf{w}(i) = [\mathbf{f}_1^T(i), \ldots, \mathbf{f}_l^T(i), \ldots, \mathbf{f}_L^T(i)]^T \in \mathbb{C}^{ML\times 1}$, where $\mathbf{f}_l(i) = [w_{l,1}(i), \ldots, w_{l,M}(i)]^T \in \mathbb{C}^{M\times 1}$ denotes the spatial filter corresponding to the lth delay taps, $l \in \{1, \ldots, L\}$. The filter $\mathbf{w}(i)$ provides an estimate of the desired navigation signal $s(i)$, given by the inner product $y(i) = \mathbf{w}^H(i)\mathbf{x}(i)$, which is then processed by the navigation receiver. To obtain the augmented filter, one can formulate the following optimization problem based on the constrained minimum variance criterion [22]

$$\min_{\mathbf{w}(i)} \ \mathbb{E}\left\{\left|\mathbf{w}^H(i)\mathbf{x}(i)\right|^2\right\}$$
$$\text{s.t.} \ \ \mathbf{w}^H(i)\mathbf{a}_a(\theta_0) = 1. \quad (4)$$

Here, we use a single constraint with

$$\mathbf{a}_a(\theta_0) = \left[\mathbf{a}(\theta_0)^T, \mathbf{0}_{ML-M}^T\right]^T. \quad (5)$$

As pointed out in [21] and [23], this constraint does not force the delayed version of the navigation signal to zero, which may cause frequency distortion. However, this problem can be handled by the subsequent despreading operation within the receiver, so we use this constraint for its simplicity. By using the GSC structure [4], the constrained optimization problem in (4) can be converted into an unconstrained one with the following cost function:

$$\min_{\mathbf{w}_g(i)} \ J = \mathbb{E}\left\{\left|(\mathbf{a}_a(\theta_0) - \mathbf{B}\mathbf{w}_g(i))^H \mathbf{x}(i)\right|^2\right\}. \quad (6)$$

The full-rank weight vector is then given by $\mathbf{w}(i) = \mathbf{a}_a(\theta_0) - \mathbf{B}\mathbf{w}_g(i)$, where $\mathbf{B}$ is the signal blocking matrix, which spans a subspace that is orthogonal to the augmented steering vector $\mathbf{a}_a(\theta_0)$ [24]. For an augmented vector $\mathbf{x}(i)$ with a large dimension ML, the convergence speed for full-rank ST adaptive GSC beamformers based on (6) is generally slow. As a result, we resort to reduced-rank techniques to solve this problem.

III. PROPOSED ALRD REDUCED-RANK ALGORITHM

Apart from the existing reduced-rank techniques, the most straightforward approach to reduce the dimensionality of the received augmented data vector $\mathbf{x}(i)$ is to decimate its content, i.e., to retain a subset of its elements while discarding the rest. However, this approach may entail a loss of information and, therefore, result in poor convergence performance. To overcome this problem, the proposed technique first performs a linear selection operation on the received vector. This is achieved by uniformly dividing the elements of the data vector into D groups and picking up the first I elements within each group. Then, the selected elements in each group are preprocessed by a basis vector and the desired reduced-dimension data vector, say $\bar{\mathbf{x}}(i)$, is generated by stacking all the resulting signals from the D groups. Finally, the resulting reduced-dimension data vector $\bar{\mathbf{x}}(i)$ is processed by a reduced-rank receive filter. Compared to the existing reduced-rank algorithms, the proposed technique, which we referred to as ALRD, can significantly reduce the computational complexity and improve the convergence performance. The proposed ALRD technique and

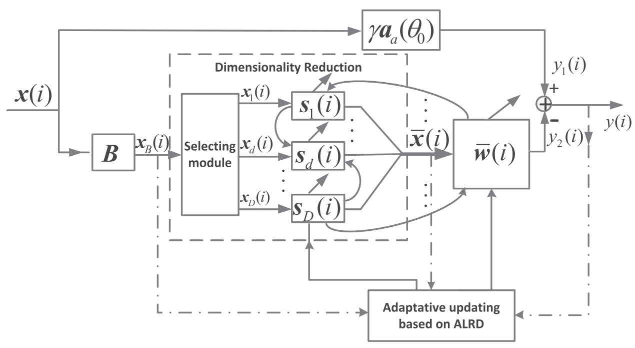

Fig. 2. Block diagram of proposed ALRD reduced-rank scheme based on the GSC structure.

its adaptive implementation within the GSC framework are explained in further details ahead.

A. Proposed ALRD Reduced-Rank Technique

The proposed ALRD reduced-rank beamformer based on the GSC structure is illustrated in Fig. 2. As can be seen, similar to the full-rank GSC processing scheme, the reduced-rank structure is composed of a constrained component (bottom branch) and an unconstrained component (top branch). Let us focus on the bottom branch, and more specifically on the construction of the projection matrix and the design of the reduced-rank receive filter. The augmented data vector $\mathbf{x}(i)$ first passes through a signal blocking matrix $\mathbf{B}$, which can be obtained by the singular value decomposition, the QR decomposition [24], [25], or the correlation subtractive structure (CSS) [21]. In this paper, we use the CSS of the blocking matrix where

$$\mathbf{B} = \mathbf{I}_{ML} - \mathbf{a}_a(\theta_0)\mathbf{a}_a^H(\theta_0) \in \mathbb{C}^{ML\times ML} \tag{7}$$

which is Hermitian and idempotent. We point out that the computational complexity of the product $\mathbf{Bx}(i)$ is restricted to $O(ML)$ instead of $O((ML)^2)$ for a general matrix $\mathbf{B}$. Further considering the specific structure of $\mathbf{a}_a(\theta_0)$ defined in (5), which is obtained by zero padding the $M \times 1$ steering vector $\mathbf{a}(\theta_0)$, the number of operations can be reduced to $O(M)$. The so-called *blocked* signal vector $\mathbf{x}_B(i) = [x_{B,0}(i), \ldots, x_{B,ML-1}(i)]^T \in \mathbb{C}^{ML\times 1}$ is given by

$$\mathbf{x}_B(i) = \mathbf{B}^H\mathbf{x}(i) = \mathbf{Bx}(i). \tag{8}$$

Vector $\mathbf{x}_B(i)$ is then fed to the dimensionality reduction module, where it is processed by the "projection" matrix, thereby producing the rank-reduced vector

$$\bar{\mathbf{x}}(i) = \mathbf{P}_D^H\mathbf{x}_B(i) = \sum_{d=1}^{D}\mathbf{p}_d^H\mathbf{x}_B(i)\mathbf{q}_d \tag{9}$$

where $\mathbf{P}_D = [\mathbf{p}_1, \ldots, \mathbf{p}_d, \ldots, \mathbf{p}_D] \in \mathbb{C}^{ML\times D}$ with $D \ll ML$ denotes the projection matrix. Hence, vector $\mathbf{p}_d \in \mathbb{C}^{ML\times 1}$, which occupies the dth column of $\mathbf{P}_D$, forms one of the basis vectors used for dimensionality reduction. By means of the second equality in (9), where $\mathbf{q}_d$ is a $D \times 1$ selection vector with expression

$$\mathbf{q}_d = [\underbrace{0, \ldots, 0}_{d-1}, 1, \underbrace{0, \ldots, 0}_{D-d}]^T \tag{10}$$

the projection matrix $\mathbf{P}_D$ is explicitly by the D basis vectors $\mathbf{p}_d$, $d \in \{1, \ldots, D\}$.

In order to further simplify the design scheme, we restrict the number of nonzero elements for each basis vector in the following way:

$$\mathbf{p}_d = [\underbrace{0, \ldots, 0}_{\gamma_d}, \mathbf{s}_d^T, \underbrace{0, \ldots, 0}_{ML-\gamma_d-I}]^T \tag{11}$$

where $\mathbf{s}_d$ is an $I \times 1$ basis vector with $I \ll ML$, and the scalar γ_d specifies its exact position in the original $ML \times 1$ basis vector $\mathbf{p}_d$. In this paper, because of its simplicity and satisfactory performance, we use the shifting pattern defined by $\gamma_d = (d-1)\lfloor ML/D \rfloor$, where $\lfloor\cdot\rfloor$ represents the floor function, which returns the largest integer that is smaller than or equal to its argument. This pattern leads to a uniform distribution of the vectors $\mathbf{s}_d$ over the corresponding basis vectors $\mathbf{p}_d$, so as to extract as much information as possible from the high-dimensional input vector $\mathbf{x}_B(i)$. Then, referring to (9), the reduced-rank received data vector $\bar{\mathbf{x}}(i)$ can be represented as

$$\bar{\mathbf{x}}(i) = \sum_{d=1}^{D}\mathbf{d}_d^H\mathbf{C}_x(i)\mathbf{s}_d^*\mathbf{q}_d \tag{12}$$

where $\mathbf{d}_d = [\underbrace{0, \ldots, 0}_{\gamma_d}, 1, \underbrace{0, \ldots, 0}_{ML-\gamma_d-1}]^T$ is an $ML \times 1$ selection vector and the $ML \times I$ matrix $\mathbf{C}_x(i)$ is a $ML \times I$ matrix with the Hankel structure [26], obtained from the samples of $\mathbf{x}_B(i)$ in the following manner:

$$\mathbf{C}_x(i) = \begin{bmatrix} x_{B,0}(i) & x_{B,1}(i) & \cdots & x_{B,I-1}(i) \\ \vdots & \vdots & \cdots & \vdots \\ x_{B,ML-I}(i) & x_{B,ML-I+1}(i) & \cdots & x_{B,ML-1}(i) \\ x_{B,ML-I+1}(i) & x_{B,ML-I+2}(i) & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ x_{B,ML-1}(i) & 0 & \cdots & 0 \end{bmatrix}. \tag{13}$$

Defining $\mathbf{x}_d(i) = (\mathbf{d}_d^T\mathbf{C}_x(i))^T$, which corresponds to a selection operation from the $ML \times 1$ blocked signal vector $\mathbf{x}_B(i)$, the reduced-rank vector $\bar{\mathbf{x}}(i)$ can be further simplified as

$$\bar{\mathbf{x}}(i) = \sum_{d=1}^{D}\mathbf{s}_d^H\mathbf{x}_d(i)\mathbf{q}_d. \tag{14}$$

The resulting unconstrained output for the bottom branch of the GSC structure is given by

$$y_2(i) = \bar{\mathbf{w}}^H\bar{\mathbf{x}}(i) = \bar{\mathbf{w}}^H\sum_{d=1}^{D}\mathbf{s}_d^H\mathbf{x}_d(i)\mathbf{q}_d \tag{15}$$

where $\bar{\mathbf{w}}$ denotes the reduced-rank filter. Finally, the beamformer output is obtained as the difference

$$y(i) = \mathbf{a}_a^H(\theta_0)\mathbf{x}(i) - \bar{\mathbf{w}}^H \sum_{d=1}^{D} \mathbf{s}_d^H \mathbf{x}_d(i)\mathbf{q}_d. \tag{16}$$

As further explained ahead, the proposed reduced-rank algorithm jointly optimizes each one of the basis vectors $\mathbf{p}_d$ and the reduced-rank filter $\bar{\mathbf{w}}$.

B. Adaptive Joint Implementation of the Basis Vectors and Reduced-Rank Receive Filter

Next, we derive the structured RLS algorithm for the adaptive joint realization of the basis vectors and the reduced-rank filter. The filters $\mathbf{s}_d(i)$, $d = 1, \ldots, D$ and $\bar{\mathbf{w}}(i)$ are obtained by minimizing the unconstrained exponentially weighted output power cost function:

$$\begin{aligned} J(\bar{\mathbf{w}}(i), \mathbf{s}_d(i)) &= \sum_{n=1}^{i} \alpha^{i-n} |y(n)|^2 \\ &= \sum_{n=1}^{i} \alpha^{i-n} \left| \mathbf{a}_a^H(\theta_0)\mathbf{x}(n) - \sum_{d=1}^{D} \mathbf{s}_d^H(i)\mathbf{x}_d(n)\bar{w}_d^*(i) \right|^2 \end{aligned} \tag{17}$$

where α is a forgetting factor chosen as a positive scalar, close to, but less than 1 and $\bar{w}_d(i)$ denotes the dth component of $\bar{\mathbf{w}}(i)$. By fixing $\bar{\mathbf{w}}(i)$ and taking the gradient of (17) with respect to $\mathbf{s}_d^*(i)$, we obtain

$$\begin{aligned} \frac{\partial J}{\partial \mathbf{s}_d^*(i)} &= |\bar{w}_d(i)|^2 \left(\sum_{n=1}^{i} \alpha^{i-n} \mathbf{x}_d(n)\mathbf{x}_d^H(n) \right) \mathbf{s}_d(i) \\ &\quad - \bar{w}_d^*(i) \left(\sum_{n=1}^{i} \alpha^{i-n} \mathbf{x}^H(n)\mathbf{a}_a(\theta_0)\mathbf{x}_d(n) \right) \\ &\quad - \sum_{\substack{e=1 \\ e\neq d}}^{D} \bar{w}_e(i)\bar{w}_d^*(i) \left(\sum_{n=1}^{i} \alpha^{i-n} \mathbf{x}_d(n)\mathbf{x}_e^H(n) \right) \mathbf{s}_e(i), \\ &\quad d = 1, \ldots, D. \end{aligned} \tag{18}$$

By equating (18) to zero, and solving for $\mathbf{s}_d(i), d = 1, \ldots, D$, we have

$$\begin{aligned} \mathbf{s}_d(i) = \mathbf{R}_d^{-1}(i) \Bigg(& \frac{\bar{w}_d^*(i)}{|\bar{w}_d(i)|^2} \mathbf{u}_d(i) - \sum_{\substack{e=1 \\ e\neq d}}^{D} \frac{\bar{w}_e(i)\bar{w}_d^*(i)}{|\bar{w}_d(i)|^2} \\ & \times \mathbf{V}_{d,e}(i)\mathbf{s}_e(i) \Bigg) \end{aligned} \tag{19}$$

where $\mathbf{R}_d(i) = \sum_{n=1}^{i} \alpha^{i-n}\mathbf{x}_d(n)\mathbf{x}_d^H(n)$, $\mathbf{u}_d(i) = \sum_{n=1}^{i} \alpha^{i-n} \mathbf{x}^H(n)\mathbf{a}_a(\theta_0)\mathbf{x}_d(n)$, and $\mathbf{V}_{d,e}(i) = \sum_{n=1}^{i} \alpha^{i-n}\mathbf{x}_d(n)\mathbf{x}_e^H(n)$. We note that (19) requires the application of a matrix inverse operation to form $\mathbf{R}_d^{-1}(i)$. To reduce the computational complexity, we apply the matrix inversion lemma [4] and obtain

$$\mathbf{k}_d(i) = \frac{\mathbf{R}_d^{-1}(i-1)\mathbf{x}_d(i)}{\alpha + \mathbf{x}_d^H(i)\mathbf{R}_d^{-1}(i-1)\mathbf{x}_d(i)} \tag{20}$$

$$\mathbf{R}_d^{-1}(i) = \alpha^{-1} \left(\mathbf{R}_d^{-1}(i-1) - \mathbf{k}_d(i)\mathbf{x}_d^H(i)\mathbf{R}_d^{-1}(i-1) \right) \tag{21}$$

which can be iteratively updated.

To derive the iterative update procedure of the reduced-rank filter $\bar{\mathbf{w}}(i)$, we rearrange the cost function as

$$\begin{aligned} J(\bar{\mathbf{w}}(i), \mathbf{s}_d(i)) &= \sum_{n=1}^{i} \alpha^{i-n} |y(n)|^2 \\ &= \sum_{n=1}^{i} \alpha^{i-n} \left| \mathbf{a}_a^H(\theta_0)\mathbf{x}(n) - \bar{\mathbf{w}}^H(i)\bar{\mathbf{x}}(n) \right|^2 \end{aligned} \tag{22}$$

where $\bar{\mathbf{x}}(n) = \sum_{d=1}^{D} \mathbf{s}_d^H(n)\mathbf{x}_d(n)\mathbf{q}_d$, which is now quadratic in the unknown weight vector $\bar{\mathbf{w}}(i)$. By fixing $\{\mathbf{s}_d(i)\}$ and taking the gradient of (22) with respect to $\bar{\mathbf{w}}^*(i)$ and equating it to zero, after further manipulations, we obtain

$$\bar{\mathbf{w}}(i) = \mathbf{Q}^{-1}(i)\mathbf{z}(i) \tag{23}$$

where $\mathbf{Q}(i) = \sum_{n=1}^{i} \alpha^{i-n}\bar{\mathbf{x}}(n)\bar{\mathbf{x}}^H(n)$ and $\mathbf{z}(i) = \sum_{n=1}^{i} \alpha^{i-n}\bar{\mathbf{x}}(n)\mathbf{x}^H(n)\mathbf{a}_a(\theta_0)$. We use the matrix inversion lemma again and obtain the following two equations:

$$\mathbf{g}(i) = \frac{\mathbf{Q}^{-1}(i-1)\bar{\mathbf{x}}(i)}{\alpha + \bar{\mathbf{x}}^H(i)\mathbf{Q}^{-1}(i-1)\bar{\mathbf{x}}(i)} \tag{24}$$

$$\mathbf{Q}^{-1}(i) = \alpha^{-1} \left(\mathbf{Q}^{-1}(i-1) - \mathbf{g}(i)\bar{\mathbf{x}}^H(i)\mathbf{Q}^{-1}(i-1) \right). \tag{25}$$

Using (23)–(25), as well as the recursion $\mathbf{z}(i) = \alpha\mathbf{z}(i-1) + \bar{\mathbf{x}}(i)\mathbf{x}^H(i)\mathbf{a}_a(\theta_0)$, we obtain

$$\bar{\mathbf{w}}(i) = \bar{\mathbf{w}}(i-1) + \mathbf{g}(i)\xi^*(i) \tag{26}$$

where $\xi(i)$ denotes the prior estimation error, with expression $\xi(i) = \mathbf{a}_a^H(\theta_0)\mathbf{x}(i) - \bar{\mathbf{w}}^H(i-1)\bar{\mathbf{x}}(i)$. The detailed ALRD reduced-rank algorithm is summarized in Table I for reference, where $\mathbf{1}$ denotes a $D \times 1$ all ones vector. The algorithm trades off a full-rank adaptive beamformer with a total of ML coefficients against D basis vectors with dimension $I \times 1$, with the update recursions given in (19)–(21), plus one $D \times 1$ reduced-rank adaptive receive filter $\bar{\mathbf{w}}(i)$, with recursive update given in (24)–(26). This alternating scheme for updating and exchanging information between the two subsets of parameters can lead to a better performance of the adaptive beamformer, as verified in our simulations. In order to further improve the performance of the proposed adaptive ALRD reduced-rank algorithm, we can also use the automatic parameter selection scheme proposed in [11] to determine the parameter I and D. Due to the space limitation, we do not incorporate it in this paper.

C. Computational Complexity Analysis

In Table II, we analyze the computational complexity of

TABLE I
Proposed ALRD Reduced-Rank Algorithm

Initialization:

$\mathbf{R}_d^{-1}(0) = \delta_d \mathbf{I}_I, \mathbf{s}_d(0) = [1, 0, \ldots, 0]^T, \mathbf{u}_d(0) = \mathbf{0}$

$\mathbf{V}_{d,e}(0) = \mathbf{0}_I, \quad d = 1, \ldots, D, \quad e = 1, \ldots, D$

$\tilde{\mathbf{Q}}^{-1}(0) = \delta \mathbf{I}_D, \bar{\mathbf{w}}(0) = \mathbf{1}/\sqrt{D}$

For the ith snapshot $i = 1, 2, \ldots, N$

$\mathbf{x}_B(i) = \mathbf{B}\mathbf{x}(i)$, $\mathbf{x}_d(i) = \mathbf{C}_x^T(i)\mathbf{d}_d$

Update the basis vector $\mathbf{s}_d(i), d = 1, \ldots, D$:

for $e = 1, \ldots, D, \quad e \neq d$:

$\mathbf{V}_{d,e}(i) = \alpha \mathbf{V}_{d,e}(i-1) + \mathbf{x}_d(i)\mathbf{x}_e^H(i)$,

$\mathbf{k}_d(i) = \frac{\mathbf{R}_d^{-1}(i-1)\mathbf{x}_d(i)}{\alpha + \mathbf{x}_d^H(i)\mathbf{R}_d^{-1}(i-1)\mathbf{x}_d(i)}$,

$\mathbf{R}_d^{-1}(i) = \alpha^{-1}\left(\mathbf{R}_d^{-1}(i-1) - \mathbf{k}_d(i)\mathbf{x}_d^H(i)\mathbf{R}_d^{-1}(i-1)\right)$,

$\mathbf{u}_d(i) = \alpha \mathbf{u}_d(i-1) + \mathbf{x}^H(i)\mathbf{a}_a(\theta_0)\mathbf{x}_d(i)$,

$\mathbf{s}_d(i) = \mathbf{R}_d^{-1}(i)\left(\frac{\bar{w}_d^*(i)}{|\bar{w}_d(i)|^2}\mathbf{u}_d(i) - \sum_{\substack{e=1 \\ e \neq d}}^{D} \frac{\bar{w}_e(i)\bar{w}_d^*(i)}{|\bar{w}_d(i)|^2}\mathbf{V}_{d,e}(i)\mathbf{s}_e(i)\right)$.

Update the reduced-rank filter $\bar{\mathbf{w}}(i)$:

$\bar{\mathbf{x}}(i) = \sum_{d=1}^{D} \mathbf{s}_d^H(i)\mathbf{x}_d(i)\mathbf{q}_d$,

$\mathbf{g}(i) = \frac{\mathbf{Q}^{-1}(i-1)\bar{\mathbf{x}}(i)}{\alpha + \bar{\mathbf{x}}^H(i)\mathbf{Q}^{-1}(i-1)\bar{\mathbf{x}}(i)}$,

$\xi(i) = \mathbf{a}_a^H(\theta_0)\mathbf{x}(i) - \bar{\mathbf{w}}^H(i-1)\bar{\mathbf{x}}(i)$,

$\mathbf{Q}^{-1}(i) = \alpha^{-1}\left(\mathbf{Q}^{-1}(i-1) - \mathbf{g}(i)\bar{\mathbf{x}}^H(i)\mathbf{Q}^{-1}(i-1)\right)$,

$\bar{\mathbf{w}}(i) = \bar{\mathbf{w}}(i-1) + \mathbf{g}(i)\xi^*(i)$.

TABLE II
Computational Complexity for ALRD and Other Algorithms

Algorithms	Number of operations per symbol	
	Multiplications	Additions
ALRD	$3(DI)^2 + D^2I + DI^2$ $+5DI + 4D^2 + 5D$ $+DM + 2M$	$2(DI)^2 + DI^2 + 2DI$ $+2D^2 + D + DM$ $+2M - 1$
MSWF	$(3D+2)(ML)^2 + 3D^2$ $(3D+1)(ML) + 4D$	$(3D+1)(ML)^2$ $+2D^2$
AVF	$2(ML)^2D + 2(ML)^2$ $+3MLD + 2MD$	$2(ML)^2D + (ML)^2$ $+MLD + 2MD - 3D$
FR	$3(ML)^2 + 4ML + 2M$	$2(ML)^2 + 2ML + 2M - 1$

TABLE III
Computational Complexity for a Specific System Configuration

Algorithms	Number of operations per symbol	
	Multiplications	Additions
ALRD	8124	5435
MSWF	360544	332832
AVF	258080	231188
FR	77480	51559

the proposed ALRD reduced-rank algorithm with the GSC structure, where the complexity is evaluated in terms of the number of multiplications and additions for each new input vector $\mathbf{x}(i)$ in (3) with dimension ML. For comparison, we also analyze the complexity of the existing MSWF [8], AVF [9], and full-rank algorithms [4]. ST processing often leads to a high-dimensional input vector, which may drastically increase the computational complexity of the full-rank or traditional reduced-rank algorithms. However, for our proposed scheme, the parameters D and I are often chosen such that $DI \ll ML$, which will save a substantial number of calculations.

In particular, for a specific configuration with $M = 20$, $L = 8$, $D = 4$, and $I = 12$, the numbers of multiplications and additions for all these analyzed algorithms are listed in Table III. It can be seen that the proposed ALRD reduced-rank algorithm significantly reduces the computational complexity.

D. Convergence Analysis

Next, the convergence behavior of the proposed ALRD adaptive algorithm, which jointly computes D basis vectors plus one reduced-rank receive filter in an adaptive manner. Note that a sufficient, but not necessary, condition for global convergence is the convexity of the cost function, which is verified when the associated Hessian matrix is positive semidefinite. In our proposed scheme, the original cost function can be written as

$$J(\bar{\mathbf{w}}(i), \mathbf{s}_d(i)) = \mathbb{E}\left\{|y(i)|^2\right\} = \mathbb{E}\left\{\left|\mathbf{a}_a^H(\theta_0)\mathbf{x}(i) - \sum_{d=1}^{D}\mathbf{s}_d^H(i)\mathbf{x}_d(i)\bar{w}_d^*(i)\right|^2\right\}. \quad (27)$$

Considering the case where only one parameter vector $\mathbf{s}_d(i)$, $d = 1, \ldots, D$ or $\bar{\mathbf{w}}(i)$ is variable while the remaining parameters are all given, the convexity of the above-mentioned optimization problem has been proven in [19]. However, when all these $D + 1$ vectors are allowed to vary in a joint optimization, the convexity is no longer assured and the optimization analysis of (27) is an interesting open problem. Here, we adopt conceptually pragmatic point of view to clarify its convergence. The recursive updating procedure (19)–(21) and (24)–(26) for $\mathbf{s}_d(i)$ and $\bar{\mathbf{w}}(i)$, respectively, implies that from the $(i-1)$th time instant to the ith time instant, every single adjustment corresponds to a small displacement from the specific vector location, which will tend to decrease the cost function as it is oriented in a direction opposite to its local gradient [4]. In effect, each parameter vector update further depends on other vectors that have been previously updated, and this information exchange tends to further accelerate the decrease of the cost function. Finally, after a number of time iterations using a sequence of received data vectors, the proposed alternating algorithm will tend converge to the desired solution, which is also verified by our simulations.

IV. SIMULATIONS

In this section, we assess proposed ALRD reduced-rank algorithm, which is applied to jammer suppression in a navigation receiver. For comparison, we also implement the commonly used adaptive interference suppression algorithms in such systems, which include the existing MSWF [8], AVF [9], and full-rank algorithms [4]. Monte Carlo simulations are carried out to obtain the desired results. In our simulations, we assume that the ST ULA is equipped with $M = 8$ antenna elements, and for each element, a tapped delay line with $L = 20$ coefficients is employed, thereby providing an augmented data vector with a total dimension of $ML = 160$. We utilize the coarse/acquisition

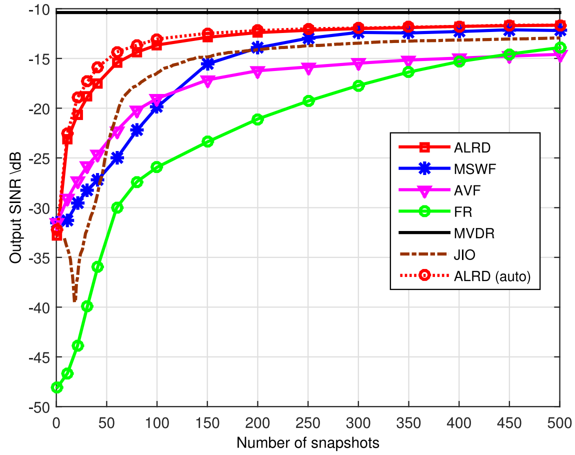

Fig. 3. Output SINR convergence performance for the proposed and existing algorithms.

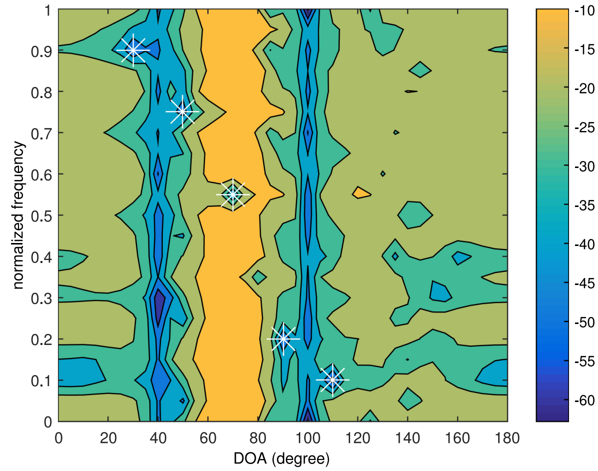

Fig. 4. Beampattern of the proposed ALRD-reduced-rank processor.

code to modulate the navigation signal of a navigation system with SNR -23 dB, which is received from a satellite with DOA $\theta_0 = 70^\circ$. Two wideband jammer signals that cover the whole frequency band impinge on the array with DOAs of $[40^\circ, 100^\circ]$. For each wideband jammer, the corresponding jammer-to-noise ratio (JNR) is set as 30 dB. In addition, there are five narrowband jammers with DOAs of $[110^\circ, 90^\circ, 70^\circ, 50^\circ, 30^\circ]$ and normalized frequencies $[0.1, 0.2, 0.55, 0.75, 0.9]$. The corresponding JNRs are [20, 30, 20, 25, 25] dB, respectively.

A. Comparison Between the Proposed and Existing Algorithms

In Fig. 3, we investigate the output signal-to-interference-plus-noise ratio (SINR) convergence versus the number of snapshots. The performance of the optimum minimum variance distortionless response (MVDR) spatial filter is provided as a reference. For comparison, the existing full-rank and reduced-rank algorithms, i.e., the full-rank adaptive RLS [4], JIO [11], MSWF [8], and AVF [9] based reduced-rank algorithms are considered. In Fig. 3, the corresponding curves are labeled using the acronyms ALRD, FR, JIO, MSWF, AVF, and MVDR, respectively. Clearly, the proposed ALRD reduced-rank algorithm exhibits faster convergence performance and higher steady-state SINR with a smaller gap to the MVDR solution in comparison to the other adaptive algorithms. In this simulation, we use $D = 7$ for the conventional MSWF and AVF reduced-rank algorithms in order to generate the best performance for comparison. In addition, we also compare the ALRD-based reduced-rank algorithm with the automatic model selection [11] to that with fixed parameters. We set $D_{\min} = I_{\min} = 2$ and $D_{\max} = I_{\max} = 15$ for the proposed automatic parameter selection scheme. For the fixed parameter scheme, we tuned $I = 12$ and $D = 4$ based on experience. From the results, the ALRD-based reduced-rank algorithm with the automatic parameter selection scheme provides better performance than that with fixed parameters.

B. Beampattern and the Despreading Performance

The beampattern produced by the proposed ST adaptive processing algorithm is depicted in Fig. 4, which shows the beamformer output power versus the DOA and normalized frequency. This beampattern shows that the adaptive array can yield a high processing gain toward the direction of the navigation satellite while creating nulls at the interference locations. Specifically, nulls corresponding to the wideband jammers appear as stripes parallel to the frequency axis, whereas nulls corresponding to the narrowband jammers appear as narrow and deep holes, whose positions are explicitly indicated with a star for clarity. The designed ST adaptive receiver achieves its intended goal of jammer suppression, but also provides more precise information about narrowband jammers' DOA and frequency than the conventional spatial filter. It can even cancel narrowband jammers coming from the same direction as the navigation satellite signal (as shown by the star on the vertical stripe with DOA $= 70^\circ$).

The last experiment in Fig. 5 illustrates the despreading performance of the various adaptive receivers under consideration, where the navigation system employs the principles of spread spectrum communications. After the ST processing for canceling strong jamming signals, a despreading operation is cascaded for synchronization between the receiver and the satellite. The synchronization is conducted based on the cross correlation between the ST processor outputs and a locally stored spreading sequence. Fig. 5 (a)–(d) exhibits the correlation results of the ALRD, MSWF, AVF, and the FR algorithms with different local time offsets (at the chip rate) on the x-axis, where the number of snapshots for these algorithms is set as $i = 200$. It can be seen that the ALRD reduced-rank algorithm shows an explicit correlation peak at the correct time offset with a relatively lower output noise power as compared with other analyzed algorithms. Basically, the proposed scheme can effectively cancel the jammers and, thus, achieves a better performance in synchronization.

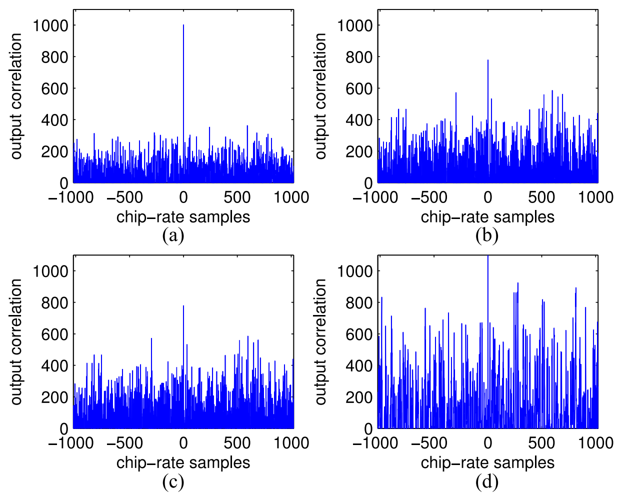

Fig. 5. Despreading performance for the various algorithms under study. (a) ALRD. (b) MSWF. (c) AVF. (d) FR.

V. CONCLUSION

In this paper, we have proposed a low-complexity ALRD algorithm based on the GSC structure for ST adaptive interference suppression in navigation receivers. In this approach, large-dimensional received data vector is processed by a rank reduction matrix with structured basis vectors that performs dimensionality reduction and a reduced-rank receive filter successively. Specifically, the number of nonzero elements for each basis vector is restricted in an efficient way for the sake of simplicity. An RLS-based adaptive algorithm has been devised to jointly estimate the projection vectors and the reduced-rank receive filter using an alternating optimization strategy. The computational complexity analysis has been provided to compare the proposed algorithm with the existing adaptive algorithms. Simulation results have shown that the proposed algorithm achieves a superior performance than existing full-rank and reduced-rank algorithms.

YUNLONG CAI
XIAOMIN WU
MINJIAN ZHAO
Zhejiang University, Hangzhou, China

RODRIGO C. DE LAMARE
CETUC-PUC-Rio, Rio de Janeiro, Brazil and also University of York, York, U.K.

BENOIT CHAMPAGNE
McGill University, Montreal, Canada

REFERENCES

[1] E. D. Kaplan
Understanding GPS: Principles and Applications. Norwood, MA, USA: Artech House, 1996.

[2] B. Parkinson, J. Spilker, Jr., P. Axelrad, and P. Enge
Global Positioning System: Theory and Application, vol. 1. New York, NY, USA: American Institute of Aeronautics and Astronautics, 1995.

[3] M. S. Grewal, L. R. Weill, and A. P. Andrews
Global Positioning Systems, Inertial Navigation, and Integration. New York, NY, USA: Wiley, 2001.

[4] S. Haykin
Adaptive Filter Theory, 4th ed. Englewood Cliffs, NJ, USA: Prentice-Hall, 2002.

[5] A. M. Haimovich and Y. Bar-Ness
An eigenanalysis interference canceler
IEEE Trans. Signal Process., vol. 39, no. 1, pp. 76–84, Jan. 1991.

[6] J. S. Goldstein and I. S. Reed
Reduced-rank adaptive filtering
IEEE Trans. Signal Process., vol. 45, no. 2, pp. 492–496, Feb. 1997.

[7] J. S. Goldstein, I. S. Reed, and L. L. Scharf
A multistage representation of the Wiener filter based on orthogonal projections
IEEE Trans. Inf. Theory, vol. 44, no. 11, pp. 2943–2959, Nov. 1998.

[8] M. L. Honig and J. S. Goldstein
Adaptive reduced-rank interference suppression based on the multistage Wiener filter
IEEE Trans. Commun., vol. 50, no. 6, pp. 986–994, Jun. 2002.

[9] D. A. Pados and G. N. Karystinos
An iterative algorithm for the computation of the MVDR filter
IEEE Trans. Signal Process., vol. 49, no. 2, pp. 290–300, Feb. 2001.

[10] R. C. de Lamare and R. Sampaio-Neto
Reduced-rank adaptive filtering based on joint iterative optimization of adaptive filters
IEEE Signal Process. Lett., vol. 14, no. 12, pp. 980–983, Dec. 2007.

[11] R. C. de Lamare and R. Sampaio-Neto
Reduced-rank space-time adaptive interference suppression with joint iterative least squares algorithms for spread-spectrum systems
IEEE Trans. Veh. Technol., vol. 59, no. 3, pp. 1217–1228, Mar. 2010.

[12] R. C. de Lamare and R. Sampaio-Neto
Adaptive reduced-rank equalization algorithms based on alternating optimization design techniques for MIMO systems
IEEE Trans. Veh. Technol., vol. 60, no. 6, pp. 2482–2494, Jul. 2011.

[13] L. Wang, R. C. de Lamare, and M. Haardt
Direction finding algorithms based on joint iterative subspace optimization
IEEE Trans. Aerosp. Electron. Syst., vol. 50, no. 4, pp. 2541–2553, Oct. 2014.

[14] N. Song, W. U. Alokozai, R. C. de Lamare, and M. Haardt
Adaptive widely linear reduced-rank beamforming based on joint iterative optimization
IEEE Signal Process. Lett., vol. 21, no. 3, pp. 265–269, Mar. 2014.

[15] Y. Cai and R. C. de Lamare
Adaptive linear minimum BER reduced-rank interference suppression algorithms based on joint and iterative optimization of filters
IEEE Commun. Lett., vol. 17, no. 4, pp. 633–636, Apr. 2013.

[16] R. C. de Lamare and R. Sampaio-Neto
Adaptive reduced-rank processing based on joint and iterative interpolation, decimation, and filtering
IEEE Trans. Signal Process., vol. 57, no. 7, pp. 2503–2514, Jul. 2009.

[17] Y. Cai, R. C. de Lamare, B. Champagne, B. Qin, and M. Zhao
Adaptive reduced-rank receive processing based on minimum symbol-error-rate criterion for large-scale multiple-antenna systems
IEEE Trans. Commun., vol. 63, no. 11, pp. 4185–4201, Nov. 2015.

[18] Y. Cai, B. Qin, and H. Zhang
An improved adaptive constrained constant modulus reduced-rank algorithm with sparse updates for beamforming
Multidimensional Syst. Signal Process., vol. 27, pp. 321–340, 2016.

[19] F. Rui, R. C. de Lamare, and L. Wang
Reduced-rank STAP schemes for airborne radar based on switched joint interpolation, decimation and filtering algorithm
IEEE Trans. Aerosp. Electron. Syst., vol. 58, no. 8, pp. 4182–4194, Aug. 2010.

[20] T.-K. Woo
HRLS: A more efficient RLS algorithm for adaptive FIR filtering
IEEE Commun. Lett., vol. 5, no. 3, pp. 81–84, Mar. 2001.

[21] S. Werner, M. With, and V. Koivunen
Householder multistage wiener filter for space-time navigation receivers
IEEE Trans. Aerosp. Electron. Syst., vol. 43, no. 3, pp. 975–988, Jul. 2007.

[22] M. L. Honig, U. Madhow, and S. Verdu
Blind adaptive multiuser detection
IEEE Trans. Inf. Theory, vol. 41, no. 4, pp. 944–960, Jul. 1995.

[23] M. L. Wyrick, J. S. Goldstein, and M. D. Zoltowski
Low complexity anti-jam space-time processing for GPS
In *Proc. Int. Conf. Acoust., Speech, Signal Process.*, May 2001, vol. 4, pp. 2233–2236.

[24] Z. Xu and K. T. Michail
Blind adaptive algorithms for minimum variance CDMA receivers
IEEE Trans. Commun., vol. 49, no. 1, pp. 180–194, Jan. 2001.

[25] J. S. Goldstein and I. S. Reed
Theory of partially adaptive radar
IEEE Trans. Aerosp. Electron. Syst., vol. 33, no. 4, pp. 1309–1325, Oct. 1997.

[26] G. H. Golub and C. F. Van Loan
Matrix Computations. 3rd ed. Baltimore, MD, USA: The Johns Hopkins Univ. Press, 1996.