

## Research Article

# An Adaptive Constraint Method for Paraunitary Filter Banks with Applications to Spatiotemporal Subspace Tracking

Scott C. Douglas

Department of Electrical Engineering, School of Engineering, Southern Methodist University, P.O. Box 750338, Dallas, TX 75275, USA

Received 1 October 2005; Revised 8 April 2006; Accepted 30 April 2006

Recommended by Vincent Poor

This paper presents an adaptive method for maintaining paraunitary constraints on direct-form multichannel finite impulse response (FIR) filters. The technique is a spatiotemporal extension of a simple iterative procedure for imposing orthogonality constraints on nearly unitary matrices. A convergence analysis indicates that it has a large capture region, and its convergence rate is shown to be locally quadratic. Simulations of the method verify its capabilities in maintaining paraunitary constraints for gradient-based spatiotemporal principal and minor subspace tracking. Finally, as the technique is easily extended to multidimensional convolution forms, we illustrate such an extension for two-dimensional adaptive paraunitary filters using a simple image sequence encoding example.

Copyright © 2007 Hindawi Publishing Corporation. All rights reserved.

## 1. INTRODUCTION

Paraunitary filters and their one-dimensional cousins, allpass filters, are important for a number of useful signal processing tasks, including coding, deconvolution and equalization, beamforming, and subspace processing [1–12]. Paraunitary filters are lossless devices, such that no spectral energy is lost or gained in any targeted spatial dimension of the multichannel input signal being filtered. The main use of paraunitary filters is to alter the phase relationships of the signals being sent through them. They are also typically used to reduce the spatial dimensionality of a multichannel signal with a minimal loss of signal power in the process.

Adaptive paraunitary filters are devices that adjust their characteristics to meet some prescribed task while maintaining paraunitary constraints on the multichannel system. For a general adaptive paraunitary filtering task, an  $n$ -input,  $m$ -output multichannel system operates on the vector input sequence  $\mathbf{x}(k) = [x_1(k) \cdots x_n(k)]^T$  to produce the output sequence

$$\mathbf{y}(k) = \sum_{p=0}^{L-1} \mathbf{W}_p \mathbf{x}(k-p), \quad (1)$$

where the  $(m \times n)$ -dimensional matrix sequence  $\{\mathbf{W}_p\}$ ,  $0 \leq p \leq L-1$ , with  $L$  odd (we choose an odd-length FIR filter structure for notational convenience) contains the coeffi-

cients of the multichannel adaptive linear system. The goal is to minimize or maximize a cost function typically depending on the sequence  $\{\mathbf{y}(k)\}$ , such as the mean-squared error  $E\{\|\mathbf{e}(k)\|^2\}$  with  $\mathbf{e}(k) = \mathbf{d}(k) - \mathbf{y}(k)$  and  $\mathbf{d}(k)$  being an  $m$ -dimensional desired response vector sequence, or the mean output power  $E\{\|\mathbf{y}(k)\|^2\}$ , while maintaining paraunitary constraints on  $\{\mathbf{W}_p\}$ . These constraints can be described in the time domain as

$$\sum_{p=\max\{0,l\}}^{\min\{L-1,L-1+l\}} \mathbf{W}_p \mathbf{W}_{p-l}^T = \mathbf{I}_m \delta_l, \quad -M \leq l \leq M, \quad (2)$$

where  $\mathbf{I}_m$  is the  $m$ -dimensional identity matrix,  $\cdot^T$  denotes the transpose operation, and  $M = (L-1)/2$  is typically chosen. Alternatively, they can be described in the frequency domain as

$$\mathcal{W}(e^{j\omega}) \mathcal{W}^T(e^{-j\omega}) = \mathbf{I}_m, \quad (3)$$

for some discrete set of frequencies  $\omega \in [-\pi, \pi]$ , where  $\mathcal{W}(z)$  is the  $z$ -transform of  $\{\mathbf{W}\}$  given by

$$\mathcal{W}(z) = \sum_{l=0}^{L-1} \mathbf{W}_l z^{-l}. \quad (4)$$

Although the constraints in (2) or (3) imply a similarity to the rows of  $\mathbf{W}_p$  or  $\mathcal{W}(z)$ , the cost function is optimized and/or the input signal statistics usually cause the parameters within these rows to converge to different, unique solutions. When  $m = n = 1$ , (3) implies that the unknown system has a unit magnitude frequency response.

Historically, there have been two basic approaches for adaptive paraunitary systems. The first approach builds the constraints defined by (2) or (3) into the system structure, such that the system is guaranteed by design to maintain the constraints. This approach uses a minimal parametrization, which is good for numerical reasons. The adaptation algorithm becomes more complicated, however, and stability monitoring may be necessary. Examples of this approach include the adaptive allpass filter described in [1] and the adaptive paraunitary filter described in [3].

The second approach chooses a convenient, potentially overparametrized structure for the adaptive system, for example, a multichannel finite-impulse response (FIR) filter, and adapts the coefficients of this structure in ways that approximately maintain allpass or paraunitary constraints on the system. These approaches are often simpler to implement due to their use of multiply-accumulates, and no stability monitoring is required for the FIR structures. Examples of such algorithms include the adaptive allpass filtering approach in [11] and the gradient-based adaptive paraunitary filtering algorithms in [12]. The overparametrized nature of their FIR-based system structure, however, means that they are prone to numerical accumulation of errors, and clever algorithm design is required to mitigate these effects in practice. In subspace tracking, numerical issues can affect the performance of subspace tracking algorithms. Such issues have made the design of minor subspace and component tracking algorithms particularly problematic in the past, leading efforts to stabilize such methods by appropriate algorithm modifications or the specification of new gradient flows [13–16]. Of course, in the simpler spatial-only case, it is possible to impose unitary constraints using a Gram-Schmidt procedure or via a symmetric square root operation, the latter of which is a projection in the Euclidean space of the vectorized system parameters [17]. For a review of such techniques, see [18]. Unfortunately, such methods are not easily extended to multichannel FIR filters, necessitating a novel approach to the task.

In this paper, we consider a third approach that might loosely be called a “step-and-constrain” method. In our procedure, the coefficients of the adaptive FIR system are adjusted to maximize or minimize a cost function, for example, by moving a small distance in the direction of the gradient of the cost, at which point the coefficients are adjusted back to the constraint space by a simple iterative procedure. Such ideas are not new in adaptive signal processing; see, for example, work on adaptation of coefficient vectors under unit-norm constraints [19] and the adaptation of unitary matrices [20]. What is new is our discovery of an iterative technique for imposing the autocorrelation constraints in (2) on a multichannel FIR system that has a number of useful properties, including fast convergence, a reasonably

large capture region, and computational simplicity. The technique is a spatiotemporal extension of a classic technique for imposing unitary constraints on close-to-unitary matrices [21]. Through frequency-domain analysis of the iterative method, we analyze the dynamics of our proposed iterative procedure, showing that convergence of the method is locally quadratic. Numerical evaluations illustrate that the technique typically converges in tens of iterations when faced with significant deviations of the multichannel system away from paraunitariness, and convergence is much faster with smaller-magnitude deviations. Moreover, when combined with existing gradient-based spatiotemporal subspace tracking algorithms, the method is observed to stabilize the numerical performance of these algorithms using only *a single iteration of the constraint update procedure* at each time instant for both principal and minor subspace tracking tasks, and it allows much larger step sizes to be used in these algorithms for faster convergence. Finally, as the technique is easily described using convolution operations, it can be extended to multidimensional signal sets, and we provide a simple image sequence coding example to show how the method might be used in such cases.

As for notation, all signals and coefficients are assumed to be real-valued, although extensions of the described method to the complex-signal case are straightforward. As a portion of our analysis is in the frequency domain, however, we will make use of complex vectors and matrices for analytical purposes.

## 2. AN ADAPTIVE ALGORITHM FOR MAINTAINING PARAUNITARY CONSTRAINTS

In this paper, our focus is on a procedure that imposes paraunitary constraints on the matrix sequence  $\{\mathbf{W}_p\}$  adaptively through its operation. Thus, the adjustment of  $\{\mathbf{W}_p\}$  by some cost-driven procedure such as a gradient maximization or minimization approach is, for the moment, implied. The technique considered in this paper would adapt  $\mathbf{W}_p = \mathbf{W}_p(t)$  iteratively for  $t = \{0, 1, 2, \dots\}$  after an update based on a cost-driven adaptive procedure has been applied, and this embedded stabilizing update would be executed for as many iterations as often as needed to impose the constraints given by (2) to an accuracy that matches the needs of the signal processing application at hand. In later sections, we will consider such an embedding for gradient-based spatiotemporal subspace analysis.

The proposed technique for imposing paraunitary constraints is

$$\mathbf{W}_p(t+1) = \frac{3}{2}\mathbf{W}_p(t) - \frac{1}{2} \sum_{l=\max\{-(L-1)/2, p-L+1\}}^{\min\{(L-1)/2, p\}} \mathbf{C}_l(t)\mathbf{W}_{p-l}(t), \quad (5)$$

where  $\mathbf{C}_l(t)$  is defined as

$$\mathbf{C}_l(t) = \begin{cases} \sum_{q=\max\{0, l\}}^{\min\{L-1, L-1+l\}} \mathbf{W}_q(t)\mathbf{W}_{q-l}^T(t) & \text{if } |l| \leq \frac{(L-1)}{2}, \\ \mathbf{0} & \text{otherwise.} \end{cases} \quad (6)$$

In both (5) and (6), the sequence  $\{\mathbf{W}_p(t)\}$  is assumed to be zero outside the interval  $p \in [0, (L-1)]$ . In order to better see the structure of this algorithm, we can use the well-known connection between polynomial multiplication and convolution to describe (5) and (6). Defining the  $z$ -transform of  $\mathbf{W}_p(t)$  as

$$\mathcal{W}_t(z) = \sum_{l=0}^{L-1} \mathbf{W}_l(t) z^{-l}, \quad (7)$$

this algorithm can be written as

$$\mathcal{W}_{t+1}(z) = \frac{3}{2} \mathcal{W}_t(z) - \frac{1}{2} \left[ [\mathcal{W}_t(z) \mathcal{W}_t^T(z^{-1})]_{-(L-1)/2}^{(L-1)/2} \mathcal{W}_t(z) \right]_0^{L-1}, \quad (8)$$

where  $[\cdot]_M^N$  denotes truncation of the polynomial to the range of powers within  $[M, N]$ .

Several initial comments about this algorithm can be made.

- (i) The technique is a spatiotemporal extension of a classic procedure for computing the best estimate of an orthogonal matrix [21], which for a  $(m \times n)$  complex-valued matrix  $\mathbf{W}(t)$  is given by

$$\mathbf{W}(t+1) = \frac{3}{2} \mathbf{W}(t) - \frac{1}{2} \mathbf{W}(t) \mathbf{W}^H(t) \mathbf{W}(t), \quad (9)$$

where  $\cdot^H$  denotes complex (Hermitian) transpose. This procedure has recently been rediscovered by the independent component analysis community as a simple method for maintaining orthogonality constraints in prewhitened blind source separation [22]. This frequency-domain connection is exact if truncation issues are ignored, or equivalently, if  $L \rightarrow \infty$ , as then we can employ the substitution  $z = e^{j\omega}$  in (8) to obtain

$$\begin{aligned} \mathcal{W}_{t+1}(e^{j\omega}) &= \frac{3}{2} \mathcal{W}_t(e^{j\omega}) - \frac{1}{2} \mathcal{W}_t(e^{j\omega}) \mathcal{W}_t^T(e^{-j\omega}) \mathcal{W}_t(e^{j\omega}). \end{aligned} \quad (10)$$

Noting that  $\mathcal{W}_t^T(e^{-j\omega}) = [\mathcal{W}_t(e^{j\omega})]^H$  for a real-valued sequence  $\mathbf{W}_p(t)$ , (10) is identical to (9) for  $\mathbf{W}(t) = \mathcal{W}_t(e^{j\omega})$ . The filter truncation employed in (5)-(6) or (8) for finite  $L$ , however, makes our proposed algorithm novel and distinct from the frequency-domain algorithm in (10).

- (ii) The technique can also be viewed as a spatiotemporal extension of a natural gradient prewhitening procedure popular for blind source separation that has been analyzed in [23, 24]. The properties of the proposed method are significantly different from these natural gradient prewhitening methods, however, because of the algorithm's large effective step size.
- (iii) The technique requires approximately  $1.25 m^2 n L^2$  multiply-accumulates at each iteration. While several iterations are typically needed to move  $\{\mathbf{W}_p(t)\}$  towards a paraunitary sequence, the number of iterations required in an online adaptive estimation setting

depends on the cost function being optimized. As we will show, in some cases a *single update* of this procedure per time instant is sufficient to maintain good overall performance.

- (iv) Since the technique involves convolution operations, fast convolution procedures can be employed to implement (5)-(6) when  $L$  is large, reducing its complexity to  $\mathcal{O}(m^2 n L \log L)$  at each iteration.

The ultimate utility of the technique in (5)-(6) depends on the theoretical and numerical properties of the update. We explore each of these issues in turn.

### 3. ALGORITHM ANALYSIS

In this section, we analyze the convergence behavior of the adaptive orthonormalization procedure given by (5)-(6). Initially, we consider the complex extension of this procedure in the single-matrix case, where  $L = 1$ . A portion of this analysis parallels that performed in [21], although we provide extensions of the results contained therein, particularly in terms of the capture region of the method. In the sequel, we extend these results for the single-matrix algorithm to the convolutional form given in (5)-(6) for an unconstrained-length (i.e., doubly infinite noncausal) paraunitary impulse response  $\{\mathbf{W}_p(t)\}$ ,  $-\infty < p < \infty$ .

Consider the update in (9) for a single  $(m \times n)$  complex-valued matrix  $\mathbf{W}(t)$ . The first three of the following four theorems pertain to this update.

**Theorem 1.** *Define a modified singular value decomposition of  $\mathbf{W}(t)$  as*

$$\mathbf{W}(t) = \mathbf{U}(t) \mathbf{\Sigma}(t) \mathbf{J}(t) \mathbf{V}^H(t), \quad (11)$$

where  $\mathbf{U}(t) \mathbf{U}^H(t) = \mathbf{U}^H(t) \mathbf{U}(t) = \mathbf{I}_m$ ,  $\mathbf{V}(t) \mathbf{V}^H(t) = \mathbf{V}^H(t) \mathbf{V}(t) = \mathbf{I}_n$ , the matrix  $\mathbf{\Sigma}(t) = \text{diag}[\sigma_1(t), \sigma_2(t), \dots, \sigma_m(t)]$  has positive real-valued unordered entries, and the matrix  $\mathbf{J}(t)$  is a diagonal matrix whose diagonal entries  $J_i(t)$  are constrained to be either (+1) or (-1). Then, it is possible to define the diagonal matrix sequences  $\mathbf{\Sigma}(t)$  and  $\mathbf{J}(t)$  such that

$$\mathbf{U}(t) = \mathbf{U}(0), \quad \mathbf{V}(t) = \mathbf{V}(0). \quad (12)$$

Equivalently, the following two relations hold:

$$\begin{aligned} \mathbf{W}(t) \mathbf{W}^H(t) &= \mathbf{U}(0) \mathbf{\Sigma}(t) \mathbf{\Sigma}^T(t) \mathbf{U}^H(0), \\ \mathbf{W}^H(t) \mathbf{W}(t) &= \mathbf{V}(0) \mathbf{\Sigma}^T(t) \mathbf{\Sigma}(t) \mathbf{V}^H(0). \end{aligned} \quad (13)$$

*Proof.* Let  $\mathbf{W}(t_0) = \mathbf{U}(t_0) \mathbf{\Sigma}(t_0) \mathbf{J}(t_0) \mathbf{V}^H(t_0)$  be the modified singular value decomposition of  $\mathbf{W}(t)$  at time  $t = t_0$ . Then, substituting for  $\mathbf{W}(t_0)$  in (9), we obtain after some simplification

$$\begin{aligned} \mathbf{W}(t_0 + 1) &= \mathbf{U}(t_0) \left[ \left( \frac{3}{2} \mathbf{\Sigma}(t_0) - \frac{1}{2} \mathbf{\Sigma}(t_0) \mathbf{\Sigma}^T(t_0) \mathbf{\Sigma}(t_0) \right) \mathbf{J}(t_0) \right] \mathbf{V}^H(t_0). \end{aligned} \quad (14)$$

Clearly, the matrix inside the large brackets on the right-hand side of (14) is diagonal, implying that

$$\begin{aligned} & \mathbf{U}^H(t_0)\mathbf{W}(t_0+1)\mathbf{V}(t_0) \\ &= \mathbf{U}^H(t_0)\mathbf{U}(t_0+1)\boldsymbol{\Sigma}(t_0+1)\mathbf{J}(t_0+1)\mathbf{V}^H(t_0+1)\mathbf{V}(t_0) \end{aligned} \quad (15)$$

is diagonal. One possible situation that guarantees the diagonal nature of  $\mathbf{U}^H(t_0)\mathbf{W}(t_0+1)\mathbf{V}(t_0)$  is  $\mathbf{U}(t_0) = \mathbf{U}(t_0+1)$  and  $\mathbf{V}(t_0) = \mathbf{V}(t_0+1)$ , such that

$$\boldsymbol{\Sigma}(t_0+1)\mathbf{J}(t_0+1) = \left( \frac{3}{2}\boldsymbol{\Sigma}(t_0) - \frac{1}{2}\boldsymbol{\Sigma}(t_0)\boldsymbol{\Sigma}^T(t_0)\boldsymbol{\Sigma}(t_0) \right) \mathbf{J}(t_0). \quad (16)$$

Define the sequences

$$\sigma_i(t_0+1) = \left| \frac{3}{2} - \frac{1}{2}\sigma_i^2(t_0) \right| \sigma_i(t_0), \quad (17)$$

$$J_i(t_0+1) = \text{sgn}(3 - \sigma_i^2(t_0))J_i(t_0). \quad (18)$$

Then, setting  $t_0 = \{0, 1, 2, \dots\}$ , the result follows.  $\square$

**Theorem 2.** *The algorithm in (9) causes the singular values of  $\mathbf{W}(t)$  to converge to unity if the following two conditions hold:*

- (1) *the singular values of  $\mathbf{W}(0)$  satisfy  $0 < \sigma_i(0) < \sqrt{3}$  or  $\sqrt{3} < \sigma_i(0) < \sqrt{5}$  for  $1 \leq i \leq m$ ;*
- (2) *none of the singular values of  $\mathbf{W}(0)$  lead to the condition  $\sigma_i(t_0) = \sqrt{3}$  for some  $t_0 \geq 1$ .*

*Proof.* Neglect the ordering of the singular values of  $\mathbf{W}(t)$ , and consider the evolution of the diagonal entries of  $\boldsymbol{\Sigma}(t)$  in (11), as defined by (17). Consider first the possibility that  $\sigma_i(0) = \sqrt{3}$ , in which case  $\sigma_i(t) = 0$  for all  $k \geq 1$ , a clearly undesirable condition. Moreover, if  $\sigma_i(t_0) = \sqrt{3}$  for some  $t_0$ , then  $\sigma_i(t) = 0$  for all  $k \geq t_0 + 1$ . Thus, values of  $\sigma_i(0)$  that lead to  $\sigma_i(t) = \sqrt{3}$  must be avoided if convergence of  $\sigma_i(t)$  to unity is desired. This verifies the second part of the theorem.

To prove the first part of the theorem, define the error criterion

$$\gamma_i(t) = \sigma_i^2(t) - 1, \quad (19)$$

such that  $\gamma_i(t) \rightarrow 0$  implies  $|\sigma_i(t)| \rightarrow 1$ . Then, (17) becomes

$$\sigma_i(t+1) = \frac{1}{2} |2 - \gamma_i(t)| \sigma_i(t). \quad (20)$$

Squaring both sides of (20), we get

$$\sigma_i^2(t+1) = \frac{1}{4} [4 - 4\gamma_i(t) + \gamma_i^2(t)] \sigma_i^2(t). \quad (21)$$

Substituting  $\sigma_i^2(t) = \gamma_i(t) + 1$ , we have after some simplification the result

$$\gamma_i(t+1) = -\frac{1}{4} [3 - \gamma_i(t)] \gamma_i^2(t). \quad (22)$$

We wish to guarantee that  $\gamma_i(t) \rightarrow 0$ , which will be the case if  $|\gamma_i(t+1)/\gamma_i(t)| < 1$  for all  $t$ . Thus, for convergence,

$$\left| \frac{\gamma_i(t+1)}{\gamma_i(t)} \right| = \frac{1}{4} |\gamma_i^2(t) - 3\gamma_i(t)| < 1. \quad (23)$$

Since  $\gamma_i(t) \geq -1$ , we can guarantee that  $|\gamma_i(t+1)/\gamma_i(t)| < 1$  if we satisfy the following two inequalities:

$$\begin{aligned} \gamma_i^2(t) - 3\gamma_i(t) &< 4 & \text{if } \gamma_i(t) \leq 0, \\ -\gamma_i^2(t) + 3\gamma_i(t) &> -4 & \text{if } \gamma_i(t) \geq 0. \end{aligned} \quad (24)$$

Employing the constraint that  $\gamma_i(t) \geq -1$ , it can be shown after further study that both inequalities are satisfied if

$$[\gamma_i(t) - 4][\gamma_i(t) + 1] < 0. \quad (25)$$

This will be the case if  $-1 < \gamma_i(t) < 4$ , which implies that

$$0 < \sigma_i(t) < \sqrt{5}. \quad (26)$$

Finally, if  $\sigma_i(0)$  satisfies (26), monotonic convergence of  $\sigma_i(t)$  to unity is guaranteed by the inequality  $|\gamma_i(t+1)/\gamma_i(t)| < 1$  over the interval  $(0, \sqrt{5})$ , so long as  $\sigma_i(t) \neq \sqrt{3}$  for any  $t$ . Thus, the first part of the theorem follows. Finally, we note that the ordering of the singular values does not affect their numerical evolutions as defined by (17), which completes the proof of the theorem.  $\square$

**Theorem 3.** *Convergence of  $\sigma_i^2(t)$  to unity is locally quadratic.*

*Proof.* This fact can be seen from the form of (22), where it can be seen for  $\gamma_i(t)$  near zero that

$$\gamma_i(t+1) \approx \frac{3}{4} \gamma_i^2(t). \quad (27)$$

$\square$

**Theorem 4.** *Define the z-transform of the sequence  $\mathbf{W}_p(t)$  as in (7). Furthermore, assume that the multichannel system function is stable, such that the multichannel system frequency response  $\mathcal{W}_t(e^{j\omega})$  satisfies  $\text{tr}[\mathcal{W}_t(e^{j\omega})\mathcal{W}_t^H(e^{-j\omega})] < \infty$ . Then, for  $L \rightarrow \infty$ , the algorithm in (5)-(6) obeys all of the results of Theorems 1, 2, and 3, namely,*

- (a) *the update in (9) only changes the singular values of  $\mathcal{W}_t(e^{j\omega})$  over time; it does not change the orientations of the left- or right-singular vectors of  $\mathcal{W}_t(e^{j\omega})$ ;*
- (b) *the singular values of  $\mathcal{W}_t(e^{j\omega})$  converge to unity as long as (i) the singular values of  $\mathcal{W}_0(e^{j\omega})$  satisfy  $0 < \sigma_i(0) < \sqrt{3}$  or  $\sqrt{3} < \sigma_i(0) < \sqrt{5}$  for  $1 \leq i \leq m$ , and (ii) none of the singular values of  $\mathcal{W}_0(e^{j\omega})$  lead to the condition  $\sigma_i(t_0) = \sqrt{3}$  for some  $t_0 \geq 1$ ;*
- (c) *convergence of  $\sigma_i^2(t)$  to unity is locally quadratic.*

*Proof.* The above results are easily seen for the case  $L \rightarrow \infty$  given the connection between (5)-(6) and (10). All that is needed is the stability of  $\mathcal{W}_t(z)$ , which is a condition given in



the statement of the theorem. In such situations, Theorems 1, 2, and 3 hold for the spatiotemporal extension in (5)-(6).  $\square$

*Remark 1.* The results of Theorems 2 and 4 indicate that the capture region of the algorithm is somewhat larger than that predicted by the analysis in [21] for the algorithm in the  $L = 1$  case, in which the constraint  $0 < \sigma_i(0) < \sqrt{3}$  was determined.<sup>1</sup> As the squares of the singular values in the spatial-only algorithm analysis correspond to the multichannel frequency response of the system  $\mathcal{W}_t(e^{j\omega})\mathcal{W}_t^T(e^{-j\omega})$ , the algorithm will remain stable and essentially monotonically convergent if

$$\lambda(\mathcal{W}_t(e^{j\omega})\mathcal{W}_t^T(e^{-j\omega})) < 5, \quad (28)$$

where  $\lambda(\mathbf{M})$  denotes the spectral radius of the Hermitian symmetric matrix  $\mathbf{M}$ . When combined with a cost-driven iterative procedure, this fact means that one should limit the step size of the cost-based portion of the overall algorithm so that the coefficients  $\{\mathbf{W}_p(t)\}$  remain in the stable capture region of the iterative procedure in (5)-(6). For gradient-based approaches, this issue is of little concern in practice, as indicated in our simulations. Explicit stabilization of the method in more aggressive adaptation scenarios is also possible. For example, if an estimate of or bound on the largest singular value  $\sigma_{\max}(0)$  of  $\mathcal{W}_0(e^{j\omega})$  is available, then one can scale all  $\mathbf{W}_p(0)$  by the inverse of this bound prior to employing the proposed iterative algorithm. An example of such a bound is

$$\sigma_{\max}(0) \leq \sqrt{\sum_{p=0}^{L-1} \text{tr}[\mathbf{W}_p(0)\mathbf{W}_p^T(0)]}, \quad (29)$$

although the computation of this bound is computationally burdensome. Simpler approaches to stabilization involving implicit coefficient normalization can be developed but will not be considered in this paper.

*Remark 2.* Many subspace tracking algorithms, including gradient-based approaches and power-iteration-based methods, are linearly convergent [18]. Thus, our proposed procedure is ideally suited for such methods, as the quadratic convergence of our method to the constraint space means that the algorithm's overall dynamics will not be limited by the adaptive procedure in (5)-(6).

*Remark 3.* Although the analytical results above justify the use of (5)-(6) as an iterative procedure for imposing paraunitary constraints on  $\{\mathbf{W}_p(t)\}$ , they do not justify the choice of impulse response truncation within the algorithm, such that

```
function [W0,Wp,W] = paraunitarytest(m,n,L,sig,numiter);
W0 = kron(eye(n,m),[zeros((L-1)/2,1);1;zeros((L-1)/2,1)]);
Wp = W0 + sig*randn(L*n,m);
W = orthW(Wp,m,n,L,numiter);

function [W] = orthW(Wp,m,n,L,numiter);

W = Wp;
for t=1:numiter
    for i=1:m
        Wt = zeros(n*L,1);
        for j=1:m
            Wt = Wt + gfun(W(:,i),W(:,j),n,L);
        end
        Wnew(:,i) = 3/2*W(:,i) - 1/2*Wt;
    end
    W = Wnew;
end

function [G,C] = gfun(U,V,n,L);

Wi = zeros(L,n);    Wi(:,i) = U;
Wj = zeros(L,n);    Wj(:,j) = V;
Ct = zeros((3*L-1)/2,1);
Z = zeros((L-1)/2,1);
ll = (L+1)/2:(3*L-1)/2;    llr = L:-1:1;
for i=1:n
    Ct = Ct + filter(Wi(llr,i),1,[Wj(:,i);Z]);
end
C = Ct(ll);
Gt = filter(C(llr),1,[Wj;zeros((L-1)/2,n)]);
Gt = Gt(ll,:);
G = Gt(:);
```

ALGORITHM 1: MATLAB implementation and testing program for the adaptive paraunitary method.

$\{C_l(t)\}$  is nonzero only for  $|l| \leq (L-1)/2$  within the update in (5). Our use of truncation is motivated by the observed performance of the procedure, in which  $\{\mathbf{W}_p(t)\}$  converges to a sequence satisfying

$$\mathbf{C}_l(t) = \mathbf{I}_m \delta_l \quad (30)$$

for  $|l| \leq (L-1)/2$  up to the numerical precision of the computing environment if it is allowed to run long enough.

Algorithm 1 provides a MATLAB implementation of the adaptive paraunitary constraint procedure. The two functions *orthW* and *gfun* apply the update in (5)-(6) to the  $(nL \times m)$  matrix  $\mathbf{W}_p$  to obtain the paraunitary system response in  $\mathbf{W}$ . The overall program *paraunitarytest* generates a perturbed paraunitary system for testing the iterative procedure in a method that we use to explore its intrinsic numerical performance in the next section.

<sup>1</sup> The condition in part 2 of Theorem 2 does not preclude the existence of a dense subset of an interval in  $(\sqrt{3}, \sqrt{5})$  such that  $\sigma_i(t) = \sqrt{3}$  for some  $k > 0$  if  $\sigma_i(0)$  belongs to this subset. Constraining  $\sigma_i(0)$  to lie in the interval  $(0, \sqrt{3})$  avoids this technical difficulty; however, numerical simulations with random initial singular values in the range  $(0, \sqrt{5})$  indicate no systemic convergence problems.

#### 4. VERIFICATION OF NUMERICAL PERFORMANCE

We now explore the behavior of the procedures in (9) and (5)-(6) via numerical simulations. The performance metric used for these simulations is the averaged value of

$$\eta(t) = \frac{\sum_{l=-(L-1)/2}^{(L-1)/2} \sum_{p=0}^{L-1} \text{tr} \left[ (\mathbf{W}_p(t) \mathbf{W}_{p+l}^T(t) - \mathbf{I}_m \delta_l)^2 \right]}{\sum_{p=0}^{L-1} \text{tr} \left[ (\mathbf{W}_p(t) \mathbf{W}_p^T(t))^2 \right]} \quad (31)$$

as computed from a set of simulation runs with different initial conditions  $\mathbf{W}(0)$  or  $\{\mathbf{W}_p(0)\}$ .

The first set of simulations is designed to verify that the convergence analysis of (9) is accurate for  $L = 1$ . For each simulation run, a ten-by-ten matrix  $\mathbf{W}(0)$  is generated with random orthonormal real-valued left and right singular vectors and a set of ten singular values uniformly distributed in the range  $(0, \sqrt{5})$ . The procedure in (9) is then applied to this initial matrix. The averaged value of the performance criterion in (31) is computed from 1000 different simulation runs of the procedure, where  $m = n = 10$ . Shown in Figure 1 is the evolution of  $E\{\eta(t)\}$  in dB, indicating that the algorithm causes  $\mathbf{W}(t)$  to converge quickly to an orthonormal matrix if the singular values of  $\mathbf{W}(0)$  lie within the algorithm's monotonic capture region.

The second set of simulations is designed to verify that the proposed spatiotemporal procedure in (5)-(6) can be used to impose paraunitary constraints on  $\{\mathbf{W}_p(t)\}$ . In these simulations,  $m = 4$ ,  $n = 7$ ,  $L = 11$ , and  $\{\mathbf{W}_p(0)\}$  is initialized as

$$\mathbf{W}_p(0) = \mathbf{I} \delta_{p-(L-1)/2} + \mathbf{N}_p, \quad (32)$$

where  $\mathbf{N}_p$  is a sequence of jointly Gaussian matrices having uncorrelated entries that were zero mean and standard deviation of either  $\text{sig} = 0.1$  or  $\text{sig} = 0.01$  (see Algorithm 1). One hundred simulation runs have been averaged to compute the performance curve shown in Figure 2. Although convergence of the performance metric is slower than that in the spatial-only case, the results show that the proposed method does cause  $\{\mathbf{W}_p(t)\}$  to converge to a paraunitary system. Moreover, if enough iterations are taken, the performance metric reaches the machine precision of the computing environment. For small initial perturbations away from paraunitariness, convergence of the algorithm is extremely fast, requiring only a few iterations to decrease the performance metric by more than 30 dB.

#### 5. APPLICATIONS TO SPATIOTEMPORAL SUBSPACE ANALYSIS

Consider a sequence of  $n$ -dimensional vectors  $\mathbf{x}(k)$  from a wide-sense stationary random process in which

$$\mathbf{R}_{\mathbf{xx}}(l) = E\{\mathbf{x}(k) \mathbf{x}^T(k-l)\} \quad (33)$$

is the autocorrelation function matrix at lag  $l$ . The goal of spatiotemporal subspace analysis is to determine an  $n$ -input,

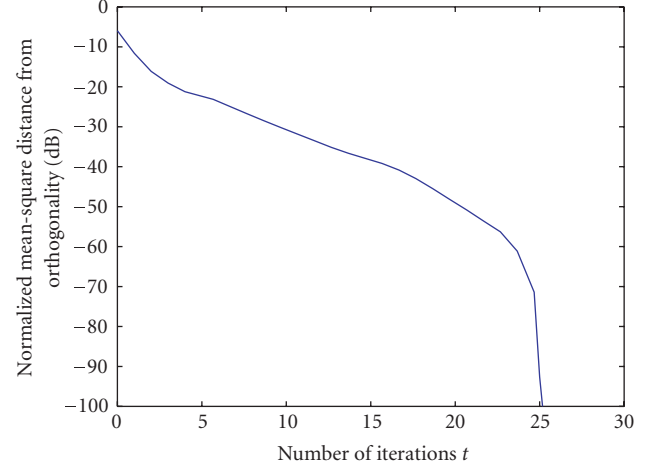


FIGURE 1: Evolution of  $E\{\eta(t)\}$  for the spatial-only unitary constraint algorithm,  $m = n = 10$ ,  $L = 1$ .

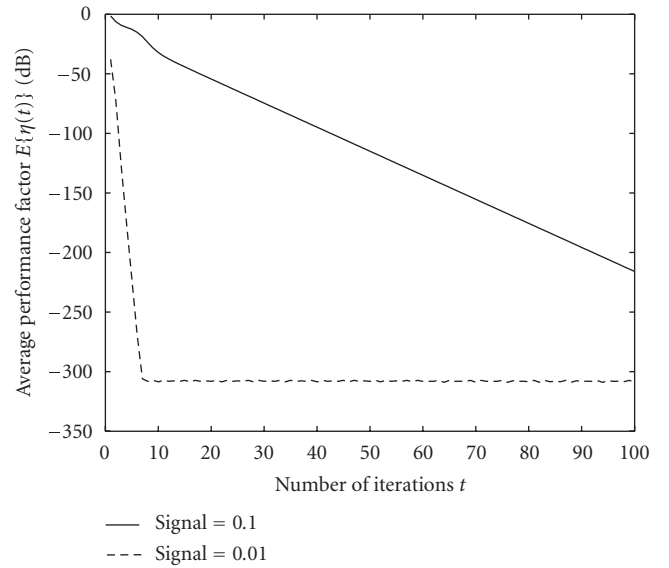


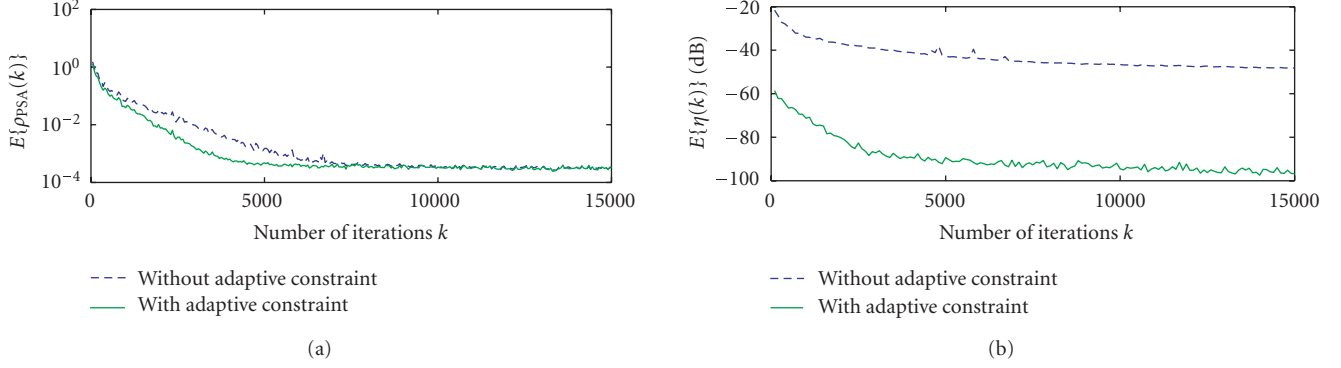
FIGURE 2: Evolution of  $E\{\eta(t)\}$  for the spatiotemporal paraunitary constraint algorithm,  $m = 4$ ,  $n = 7$ , and  $L = 11$ .

$m$ -output paraunitary system,  $m < n$ , with impulse response  $\mathbf{W}_p$  such that the output sequence

$$\mathbf{y}(k) = \sum_{p=-\infty}^{\infty} \mathbf{W}_p \mathbf{x}(k-p) \quad (34)$$

has either maximum or minimum total energy  $E\{\|\mathbf{y}(k)\|^2\}$ , where  $\|\mathbf{y}(k)\|$  denotes the  $L_2$  or Euclidean norm of  $\mathbf{y}(k)$ . If  $E\{\|\mathbf{y}(k)\|^2\}$  is maximized, then

$$\mathbf{u}(k) = \sum_{q=-\infty}^{\infty} \mathbf{W}_{-q}^T \mathbf{y}(k-q) \quad (35)$$


 FIGURE 3: Evolutions of (a)  $E\{\rho_{\text{PSA}}(k)\}$  and (b)  $E\{\eta(k)\}$  for the spatiotemporal principal subspace algorithms.

is the optimal rank- $m$  linear filtered approximation to the vector sequence  $\mathbf{x}(k)$  in a mean-square-error sense. Such techniques could be used to code multichannel signals, among other applications. Minimization of  $E\{\|\mathbf{y}(k)\|^2\}$  under paraunitary constraints yields the spatiotemporal extension of the minor subspace analysis task, which is important for direction of arrival in wideband array processing systems [2, 3, 25, 26].

In [12], simple iterative gradient-based algorithms were derived for principal and minor subspace analysis tasks. The spatiotemporal principal subspace algorithm is given by

$$\mathbf{y}(k) = \sum_{l=0}^L \mathbf{W}_l(k) \mathbf{x}(k-l), \quad (36)$$

$$\mathbf{e}(k) = \mathbf{x}(k) - \sum_{q=0}^L \mathbf{W}_{L-q}^T(k) \mathbf{y}(k-q), \quad (37)$$

$$\mathbf{W}_p(k+1) = \mathbf{W}_p(k) + \mu(k) \mathbf{y}(k-L) \mathbf{e}^T(k-p), \quad 0 \leq p \leq L, \quad (38)$$

where  $\mu(k)$  is the algorithm step size. This algorithm is the spatiotemporal extension of the well-known principal subspace rule [27]. A spatiotemporal minor subspace algorithm is also provided in [12]; it is the spatiotemporal extension of the self-stabilized algorithm in [14]. The algorithms are stochastic-gradient procedures that only approximately maintain the paraunitary constraints through their adaptive behaviors, and their ability to maintain the constraint is linked to the step size chosen for the adaptive procedure.

The proposed iterative procedure in this paper provides a potential solution to the numerical stabilization of these gradient-based algorithms, in which the imposition of the constraint is met by embedding (5)-(6) within the updates in (36)–(38). In this algorithm design, we may choose to use a limited number of iterations of (5)-(6) to improve the numerical performance of the overall algorithm, a choice that is motivated by the fast convergence of the constraint procedure. As it is now shown, *even a single iteration* of (5)-(6), when used in conjunction with (36)–(38), enables fast and accurate convergence to *either* a principal or minor subspace

estimate, depending on the sign of the step size  $\mu(k)$ . The simulations that follow explore these issues further.

Consider the example in [12], in which the following  $\mathbf{s}(k) = [s_1(k) \ s_2(k)]^T$ ,  $s_i(k)$ ,  $i \in \{1, 2\}$ , are independent zero-mean Gaussian sequences with autocorrelations  $r_{ss,i}(l) = \delta_l$ ,

$$\begin{aligned} \mathbf{x}(k) &= \hat{\mathbf{x}}(k) + \mathbf{v}(k), \\ \hat{\mathbf{x}}(k) &= \sum_{i=1}^2 \mathbf{A}_i \hat{\mathbf{x}}(k-i) + \sum_{j=0}^1 \mathbf{B}_j \mathbf{s}(k-j), \\ \mathbf{A}_1 &= \begin{bmatrix} 0.38 & 0.39 & -0.22 & 0.08 \\ 0.24 & -0.30 & -0.03 & -0.08 \\ -0.36 & -0.20 & -0.44 & 0.02 \\ -0.49 & 0.16 & 0.49 & -0.17 \end{bmatrix}, \\ \mathbf{A}_2 &= \begin{bmatrix} -0.01 & 0.01 & 0.06 & 0.06 \\ -0.05 & 0.03 & 0.04 & -0.09 \\ 0.02 & -0.06 & -0.01 & 0.02 \\ 0.05 & -0.02 & 0.01 & -0.09 \end{bmatrix}, \\ \mathbf{B}_0^T &= \begin{bmatrix} -0.02 & -0.04 & 0.07 & -0.10 \\ 0.05 & 0.09 & 0.10 & 0.06 \end{bmatrix}, \\ \mathbf{B}_1^T &= \begin{bmatrix} -0.1 & 0.0 & -0.6 & 0.3 \\ -0.4 & 0.9 & 0.5 & -0.2 \end{bmatrix}, \end{aligned} \quad (39)$$

$\mathbf{v}(k) = [v_1(k) \ v_2(k) \ v_3(k) \ v_4(k)]^T$ , and  $v_i(k)$ ,  $i \in \{1, 2, 3, 4\}$  are independent zero-mean Gaussian signals with  $r_{v_i v_i}(l) = \sigma_v^2 \delta_l$  and  $\sigma_v^2 = 10^{-4}$ . We compare the performance of (36)–(37) with and without one iteration of the adaptive constraint procedure in (5)–(6) per time instant, where  $m = 2$ ,  $n = 4$ ,  $L = 14$ ,  $\mu(k) = 0.008$  for the algorithm with the adaptive constraint method,  $\mu(k) = 0.005$  for the algorithm without the adaptive constraint method, and  $w_{ijp}(0)$  is unity if  $i = j$  and  $p = L/2$  and is zero otherwise. Note that the step size for the algorithm with the constraint method is eight times larger than that used in the simulations in [12], and the step size for the algorithm without the constraint method was chosen to obtain the fastest convergence without instability. Shown in Figures 3(a) and 3(b) are the evolutions of

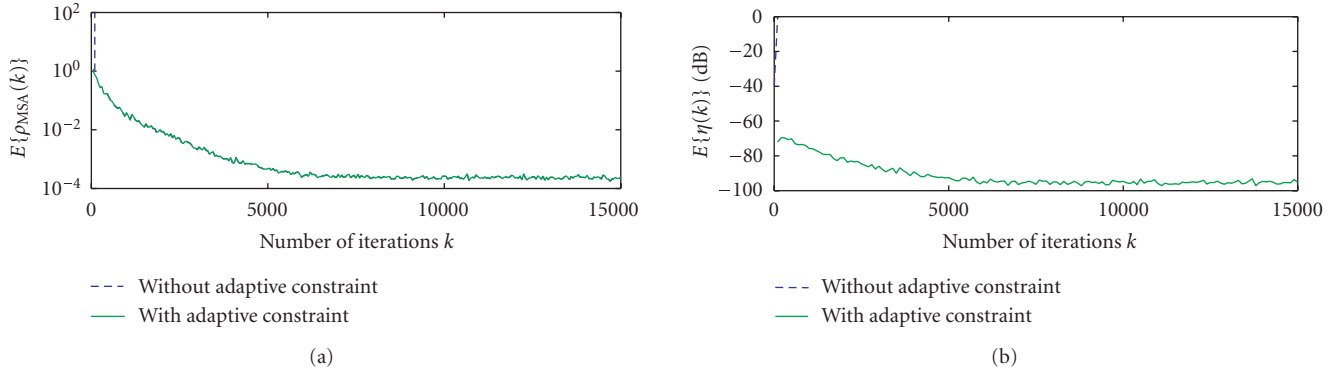


FIGURE 4: Evolutions of (a)  $E\{\rho_{\text{MSA}}(k)\}$  and (b)  $E\{\eta(k)\}$  for the spatiotemporal minor subspace algorithms.

the performance factors

$$\rho_{\text{PSA}}(k) = \|\mathbf{e}(k)\|^2 \quad (40)$$

and  $\eta(k)$  in (31), respectively, as averaged over one hundred different simulation runs. As can be seen, the proposed algorithm with a single iteration of the adaptive constraint method per time instant converges to an accurate subspace estimate that minimizes the low-rank mean-squared error criterion. The steady-state value of  $\rho_{\text{PSA}}(k)$  is approximately  $3.1 \times 10^{-4}$ , which is near the minimum value of  $2 \times 10^{-4}$  theoretically obtainable from the data model. In contrast, the original spatiotemporal principal subspace algorithm converges more slowly due to the stability limits on the algorithm step size. Larger step sizes caused this latter algorithm to diverge, that is, it could not maintain the paraunitary constraints with a larger step size despite being locally stable to the constraint space as  $\mu(k) \rightarrow 0$ . Although not easily proven, the reason for the poor performance of the original method for larger step sizes could be due to the delayed-gradient approximation employed in its derivation, in which past coefficient values appear within the coefficient updates in the error terms  $\{\mathbf{e}(k-p)\}$ . Such delayed-gradient terms are known to limit the convergence performance of filtered-gradient algorithms in multichannel active noise control systems [28]. Computing the coefficient update terms using the most recent coefficient values requires more than  $3mnL^2$  multiply-accumulates, which for  $m \ll nL^2$  is close to the complexity of a single step of the adaptive constraint procedure. Our novel adaptive projection method alleviates the convergence difficulties introduced by the delayed-gradient approximations and enables the algorithm to properly function for large step sizes.

We now explore the behavior of (36)-(37) with one iteration of the adaptive constraint procedure in (5)-(6) per time instant when applied to the spatiotemporal minor subspace analysis task, in which  $\mu(k) < 0$ . Note that, without taking any corrective measures to maintain the coefficient constraints, the update in (36)-(37) is unstable in this context as is the spatial-only principal subspace rule that is obtained when  $L = 1$  [27]. Figures 4(a) and 4(b) show the evolutions

of the performance factors

$$\rho_{\text{MSA}}(k) = \|\mathbf{y}(k)\|^2 \quad (41)$$

and  $\eta(k)$  in (31) for the same input signal model as in the previous simulation, where  $\mu(k) = -0.005$ . The algorithm without stabilization (dashed lines) quickly diverges. The algorithm with proposed stabilization method performs minor subspace analysis successfully in this situation, and its convergence speed is much faster than the self-stabilized algorithm described in [12], which requires approximately 30 000 iterations to converge under these same conditions.

## 6. MULTIDIMENSIONAL EXTENSIONS

The adaptive procedure described in (5)-(6) could be compactly and approximately defined as

$$\mathbf{W}_p(t+1) = \frac{3}{2}\mathbf{W}_p(t) - \frac{1}{2}\mathbf{W}_p(t) * \mathbf{W}_{-p}^T(t) * \mathbf{W}_p(t), \quad (42)$$

where “ $*$ ” denotes discrete-time convolution over the index  $p$  and the all-important truncation issues associated with the finite-length convolutions have been ignored. This form of the adaptive procedure inspires us to consider versions of the algorithm for higher-dimensional data, such as images, video, and hyperspectral imagery. It is reasonable to assume that, with an appropriately defined convolution operator, one could extend the procedure in (5)-(6) to these other data types. For example, consider an  $n$ -input,  $m$ -output two-dimensional (2D) FIR linear filter of the form

$$\mathbf{y}(k, l) = \sum_{p=0}^{L-1} \sum_{q=0}^{L-1} \mathbf{W}_{p,q} \mathbf{x}(k-p, l-q), \quad (43)$$

where  $\{\mathbf{W}_{p,q}\}$  contain the coefficients of the multichannel system. A multichannel 2D paraunitary filter would impose the constraints

$$\begin{aligned} & \sum_{p=\max\{0,k\}}^{\min\{L-1,L-1+k\}} \sum_{q=\max\{0,l\}}^{\min\{L-1,L-1+l\}} \mathbf{W}_{p,q} \mathbf{W}_{p-k,q-l}^T \\ & = \mathbf{I}_m \delta_k \delta_l, \quad -M \leq \{k, l\} \leq M \end{aligned} \quad (44)$$



on the coefficients of the linear system. Translating the proposed multichannel one-dimensional paraunitary constraint procedure to this two-dimensional structure, we obtain the update in polynomial form as

$$\begin{aligned} \mathcal{W}_{t+1}(z_1, z_2) &= \frac{3}{2} \mathcal{W}_t(z_1, z_2) \\ &\quad - \frac{1}{2} \left[ [\mathcal{W}_t(z_1, z_2) \mathcal{W}_t^T(z_1^{-1}, z_2^{-1})]_{-(L-1)/2}^{(L-1)/2} \mathcal{W}_t(z_1, z_2) \right]_0^{L-1}, \end{aligned} \quad (45)$$

where

$$\mathcal{W}_t(z_1, z_2) = \sum_{p=0}^{L-1} \sum_{q=0}^{L-1} \mathbf{W}_{p,q}(t) z_1^{-p} z_2^{-q} \quad (46)$$

is the 2D  $z$ -transform of  $\{\mathbf{W}_{p,q}(t)\}$  and  $[\cdot]_M^N$  here denotes truncation of its two-dimensional polynomial argument to the individual powers for  $z_1$  and  $z_2$  within the range  $[M, N]$ . We can illustrate the usefulness of this particular procedure with a simple video coding example, described using the MATLAB technical computing environment.

Consider the task of designing a three-input ( $n = 3$ ), one-output ( $m = 1$ ) paraunitary system for a set of three similar images, in which the convolution kernel for each image is of size  $(L \times L)$ , where  $L$  is odd. Let  $W1$ ,  $W2$ , and  $W3$  denote the corresponding 2D convolution kernel matrices, such that  $L = 3$ , and  $\mathcal{W}_t(z_1, z_2)$  is a  $(1 \times 3)$  vector of polynomials in  $z_1$  and  $z_2$ . Then, the following MATLAB code employing the function `filter2` can be used to impose paraunitary constraints on the filter coefficient set  $\{W1, W2, W3\}$ :

```
for t = 1:numiter
    C = filter2(W1,W1) + filter2(W2,W2)
        + filter2(W3,W3);
    W1 = 3/2*W1 - 1/2*filter2(C,W1);
    W2 = 3/2*W2 - 1/2*filter2(C,W2);
    W3 = 3/2*W3 - 1/2*filter2(C,W3);
end
```

To illustrate that this procedure works as designed, consider a simple video compression example. Given an image sequence, we first calculate a sequence of difference images. For every three difference images, we estimate a principal component image  $y(k, l)$  by maximizing the output power of the image pixels from the three-input, one-output paraunitary system while imposing a paraunitary constraint via the above adaptive procedure. In this procedure, we used a “center-spike” initialization strategy, where  $W1$  and  $W3$  were set to zero matrices and  $W2$  had one non-zero value in the center of its impulse response. We then reconstruct the first and third difference images from the single principal component image  $y(k, l)$  using  $W1$  and  $W3$ , resulting in



(a)



(b)

FIGURE 5: Reconstruction of the Cronkite sequence using 2D adaptive paraunitary filters (left-original, right-reconstructed).

the reconstructed difference images  $u_1(k, l)$  and  $u_3(k, l)$ , respectively. Finally, we use the reconstructed difference images to calculate two intermediate frames from every third “key” frame within the image sequence using adds and subtracts, respectively. The result is a compressed image sequence, because for every three frames, one only needs on average one “key” image frame, one principal component image frame  $y(k, l)$ , and the two filtering kernels  $W1$  and  $W3$  to represent three images within the sequence. Of course, such a compression scheme cannot compete with more-common motion-based image compression schemes, but the success of a 2D adaptive paraunitary filter in such an application illustrates the capability and flexibility of the proposed constraint method.

We applied the above video compression scheme to a spatially downsampled version of the Cronkite video sequence obtained from the USC SIPI database, where  $L = 3$ . In this case, the images were downsampled to size  $128 \times 128$  pixels, and a gradient-based principal component analysis procedure was used in conjunction with the adaptive 2D paraunitary constraint procedure with `numiter = 50` to maximize the output powers in the principal component images. From the sixteen-frame sequence, the ten resulting reconstructed images had an average PSNR of 26.75 dB with a standard deviation of 2.17 dB. Shown in Figure 5 are the original (left) and reconstructed (right) frames from this procedure from the eleventh (top) and twelfth (bottom) frames, respectively. As can be seen, the quality of reconstruction is high, and the proposed paraunitary constraint method can be employed to solve this approximation task.

The above paraunitary constraint procedure can be extended to the general  $N$ -dimensional filtering task. Define the sets  $\mathcal{Z}_N = \{z_1, z_2, \dots, z_N\}$  and  $\mathcal{Z}_N^{-1} = \{z_1^{-1}, z_2^{-1}, \dots, z_N^{-1}\}$ . Then, the polynomial representation of the general algorithm is

$$\begin{aligned} \mathcal{W}_{t+1}(\mathcal{Z}_N) &= \frac{3}{2} \mathcal{W}_t(\mathcal{Z}_N) \\ &\quad - \frac{1}{2} \left[ [\mathcal{W}_t(\mathcal{Z}_N) \mathcal{W}_t^T(\mathcal{Z}_N^{-1})]_{-(L-1)/2}^{(L-1)/2} \mathcal{W}_t(\mathcal{Z}_N) \right]_0^{L-1}, \end{aligned} \quad (47)$$

where

$$\mathcal{W}_t(\mathcal{Z}_N) = \sum_{i=1}^N \sum_{p_i=0}^{L-1} \mathbf{W}_{p_1, p_2, \dots, p_N}(t) \prod_{j=1}^N z_j^{-p_j} \quad (48)$$

is the  $N$ -dimensional  $z$ -transform of  $\mathbf{W}_{p_1, p_2, \dots, p_N}(t)$  and  $[\cdot]_M^P$  denotes truncation of its  $N$ -dimensional polynomial argument to the individual powers for  $z_1$  through  $z_N$  within the range  $[M, P]$ . One possible application for this method is the representation of multiple video sequences via subspace processing, a subject of current study.

## 7. CONCLUSIONS

In this paper, we have described an adaptive scheme for imposing paraunitary constraints on a multichannel linear system. The procedure is straightforward to implement, and its convergence is locally quadratic to the constraint space. We have demonstrated that the technique can be used to obtain improved convergence performance from existing simple gradient-based spatiotemporal subspace analysis methods, and we have shown how to extend the concept to higher-dimensional data sets through a simple video compression task. Extensions of these ideas are being applied to the convolutive blind source separation task; see [29] for additional details on these procedures.

## REFERENCES

- [1] B. Farhang-Boroujeny and S. Nooshfar, "Adaptive phase equalization using all-pass filters," in *Proceedings of IEEE International Conference on Communications (ICC '91)*, vol. 3, pp. 1403–1407, Denver, Colo, USA, June 1991.
- [2] P. Loubaton and P. A. Regalia, "Blind deconvolution of multivariate signals by using adaptive FIR lossless filters," in *Proceedings of the European Signal Processing Conference (EUSIPCO '92)*, pp. 1061–1064, Brussels, Belgium, August 1992.
- [3] P. A. Regalia and P. Loubaton, "Rational subspace estimation using adaptive lossless filters," *IEEE Transactions on Signal Processing*, vol. 40, no. 10, pp. 2392–2405, 1992.
- [4] T. J. Lim and M. D. Macleod, "Adaptive allpass filtering for nonminimum-phase system identification," *IEE Proceedings—Vision, Image, and Signal Processing*, vol. 141, no. 6, pp. 373–379, 1994.
- [5] M. K. Tsatsanis and G. B. Giannakis, "Principal component filter banks for optimal multiresolution analysis," *IEEE Transactions on Signal Processing*, vol. 43, no. 8, pp. 1766–1777, 1995.
- [6] P. A. McEwen and J. G. Kenney, "Allpass forward equalizer for decision feedback equalization," *IEEE Transactions on Magnetics*, vol. 31, no. 6, part 1, pp. 3045–3047, 1995.
- [7] E. Abreu, S. K. Mitra, and R. Marchesani, "Nonminimum phase channel equalization using noncausal filters," *IEEE Transactions on Signal Processing*, vol. 45, no. 1, pp. 1–13, 1997.
- [8] A. Kirac and P. P. Vaidyanathan, "Theory and design of optimum FIR compaction filters," *IEEE Transactions on Signal Processing*, vol. 46, no. 4, pp. 903–919, 1998.
- [9] P. Moulin and M. K. Mihcak, "Theory and design of signal-adapted FIR paraunitary filter banks," *IEEE Transactions on Signal Processing*, vol. 46, no. 4, pp. 920–929, 1998.
- [10] B. Xuan and R. I. Bamberger, "FIR principal component filter banks," *IEEE Transactions on Signal Processing*, vol. 46, no. 4, pp. 930–940, 1998.
- [11] X. Sun and S. C. Douglas, "Self-stabilized adaptive allpass filters for phase equalization and approximation," in *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '00)*, vol. 1, pp. 444–447, Istanbul, Turkey, June 2000.
- [12] S. C. Douglas, S.-I. Amari, and S.-Y. Kung, "Gradient adaptive paraunitary filter banks for spatio-temporal subspace analysis and multichannel blind deconvolution," *Journal of VLSI Signal Processing*, vol. 37, no. 2-3, pp. 247–261, 2004.
- [13] T.-P. Chen, S.-I. Amari, and Q. Lin, "A unified algorithm for principal and minor components extraction," *Neural Networks*, vol. 11, no. 3, pp. 385–390, 1998.
- [14] S. C. Douglas, S.-Y. Kung, and S.-I. Amari, "A self-stabilized minor subspace rule," *IEEE Signal Processing Letters*, vol. 5, no. 12, pp. 328–330, 1998.
- [15] M. A. Hasan, "Natural gradient for minor component extraction," in *Proceedings of IEEE International Symposium on Circuits and Systems (ISCAS '05)*, vol. 5, pp. 5138–5141, Kobe, Japan, May 2005.
- [16] J. H. Manton, U. Helmke, and I. M. Y. Mareels, "A dual purpose principal and minor component flow," *Systems and Control Letters*, vol. 54, no. 8, pp. 759–769, 2005.
- [17] K. Fan and A. J. Hoffman, "Some metric inequalities in the space of matrices," *Proceedings of the American Mathematical Society*, vol. 6, no. 1, pp. 111–116, 1955.
- [18] Y. Hua, "Asymptotical orthonormalization of subspace matrices without square root," *IEEE Signal Processing Magazine*, vol. 21, no. 4, pp. 56–61, 2004.
- [19] S. C. Douglas, S.-I. Amari, and S.-Y. Kung, "On gradient adaptation with unit-norm constraints," *IEEE Transactions on Signal Processing*, vol. 48, no. 6, pp. 1843–1847, 2000.
- [20] J. H. Manton, "Optimization algorithms exploiting unitary constraints," *IEEE Transactions on Signal Processing*, vol. 50, no. 3, pp. 635–650, 2002.
- [21] A. Bjorck and C. Bowie, "An iterative algorithm for computing the best estimate of an orthogonal matrix," *SIAM Journal on Numerical Analysis*, vol. 8, no. 2, pp. 358–364, 1971.
- [22] A. Hyvarinen, J. Karhunen, and E. Oja, *Independent Component Analysis*, John Wiley & Sons, New York, NY, USA, 2001.
- [23] S. C. Douglas and A. Cichocki, "Neural networks for blind decorrelation of signals," *IEEE Transactions on Signal Processing*, vol. 45, no. 11, pp. 2829–2842, 1997.
- [24] T. Chen and Q. Lin, "Dynamic behavior of the whitening process," *IEEE Signal Processing Letters*, vol. 5, no. 1, pp. 25–26, 1998.
- [25] B. Porat and B. Friedlander, "Estimation of spatial and spectral parameters of multiple sources," *IEEE Transactions on Information Theory*, vol. 29, no. 3, pp. 412–425, 1983.

- [26] B. Ottersten and T. Kailath, "Direction-of-arrival estimation for wide-band signals using the ESPRIT algorithm," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 38, no. 2, pp. 317–327, 1990.
- [27] E. Oja and J. Karhunen, "On stochastic approximation of the eigenvectors and eigenvalues of the expectation of a random matrix," *Journal of Mathematical Analysis and Applications*, vol. 106, no. 1, pp. 69–84, 1985.
- [28] S. C. Douglas, "Fast implementations of the filtered-X LMS and LMS algorithms for multichannel active noise control," *IEEE Transactions on Speech and Audio Processing*, vol. 7, no. 4, pp. 454–465, 1999.
- [29] S. C. Douglas, H. Sawada, and S. Makino, "A spatio-temporal fastica algorithm for separating convolutive mixtures," in *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP '05)*, vol. 5, pp. 165–168, Philadelphia, Pa, USA, March 2005.

**Scott C. Douglas** is an Associate Professor in the Department of Electrical Engineering at Southern Methodist University, Dallas, Tex, and is the Associate Director for the Institute for Engineering Education at SMU. He received his B.S., M.S., and Ph.D. degrees from Stanford University. Dr. Douglas is a recognized expert in the fields of adaptive filters, blind source separation, and active noise control, having authored or coauthored two books, six book chapters, and over 150 papers in journals and conference proceedings. He is a recipient of an NSF Career (Young Investigator) Award and has received significant research funding from the US Army, DARPA, other US governmental organizations, the State of Texas, and numerous companies. He is highly active in professional societies and has served as an Associate Editor for both the *IEEE Transactions on Signal Processing* and the *IEEE Signal Processing Letters*. He has served on the organizing committees of numerous international conferences and workshops as Technical Chair, Publications Chair, and Exhibits Chair, and is the General Chair of the 2010 International Conference on Acoustics, Speech, and Signal Processing. He has given many keynote and invited lectures as well as short courses on topics ranging from adaptive signal processing and control to innovative engineering education methods. Most recently, he has coauthored textbooks and developed materials and technology for the Infinity Project, a multifaceted effort to establish a United States engineering curriculum at precollege educational levels. Dr. Douglas is a frequent consultant to industry, a Senior Member of the IEEE, and a Member of both Phi Beta Kappa and Tau Beta Pi.

