

BLIND SOURCE SEPARATION AND INDEPENDENT COMPONENT ANALYSIS: A CROSSROADS OF TOOLS AND IDEAS

Scott C. Douglas

Department of Electrical Engineering
Southern Methodist University
Dallas, Texas 75275 USA

ABSTRACT

Blind source separation (BSS) and independent component analysis (ICA) are fields that have received much recent interest in various scientific and engineering communities. The numerical procedures and methods developed for BSS and ICA represent a confluence of tools and ideas whose influences extend beyond the boundaries of these two tasks. Extensions of these ideas have led to useful procedures for other problem areas. In this paper, we show how several methods and concepts in blind source separation and ICA have direct connections to one such area—adaptive signal processing—resulting in novel procedures for several tasks including phase-only adaptive filters, recursive least-squares estimation, and adaptive subspace analysis.

1. INTRODUCTION

Blind source separation (BSS) and independent component analysis (ICA) are fields that have garnered much recent interest from members of various scientific and engineering disciplines. Fields represented in the BSS and ICA community include acoustics, biology, chemistry, computer science, finance, neuroscience, medicine, physics, statistics, signal processing, and telecommunications. Several dedicate BSS and ICA workshops and symposia have been organized within these communities over the past five years to bring together researchers and practitioners to discuss the latest advancements in the field. In addition, sessions on BSS and ICA have become regular events at more-traditional topical workshops and conferences on neural networks, nonlinear systems, signal and image processing, and statistics.

With such a diverse group of participants in this research endeavor, it is challenging to identify exactly when the BSS and ICA task became popular. Both problems have strong connections to factor analysis in statistics, whose roots can be traced back to Thurstone [1]. In the signal processing community, it was the early published work of Herault and Jutten [2] and less widely-known work of Bar-Ness [3] that first established the existence of simple procedures for extracting useful signal information from spatial mixtures of this information without training data. At the same time, work on blind deconvolution and equalization of smoothed temporal sequences was yielding useful procedures for applications from geophysics to digital communications [4]–[7]. BSS and ICA received a significant boost in activity in the late 1980's and early 1990's from the work of Cardoso and Souloumiac on joint diagonalization of cumulant matrices [9], the work of Karhunen and Joutsensalo

on nonlinear principal component analysis (PCA) [10], the adaptive blind separation approaches of Cichocki and Unbehauen [11], as well as the seminal paper on cumulant-based ICA by Comon [12]. It was not until the mid-1990's, however, that an explosion of interest was generated in the two fields, led in part by two oft-cited papers on the subject [13, 14]. Since that time, hundreds of researchers have contributed their efforts.

One of the unstated goals of any great research endeavor is the development of concepts and procedures whose influences extend beyond the boundaries of the task at hand. In BSS and ICA, the development of more-general tools and ideas is enhanced through the diversity of backgrounds of the researchers working on these tasks. Many of those who work in these fields have one or more problem domains in which to apply the techniques that they have developed or learned. These researchers are in a perfect position to exploit the new ideas created by the study of BSS and ICA to solve other problems.

The goal of this paper is to illustrate the interplay between research in BSS, ICA, and other fields through particular examples drawn from the author's own research efforts in developing adaptive signal processing algorithms. While such a study is by definition not comprehensive, it shall attempt to describe the cross-fertilization process and how such connections might be made through concrete examples. Wherever possible, additional suggested efforts in the particular problem domains are described as possible starting points for future research. It is assumed that the reader is already familiar with the principles and notation of BSS and ICA as described by the many review papers and books on the subject [15]–[18]. All quantities are assumed to be real-valued. In addition, each section uses consistent mathematical notation, although slight notational differences do exist between sections.

2. BSS, ICA, AND LEAST-SQUARES ESTIMATION

2.1. Adaptive Whitening in BSS and ICA

Several iterative BSS and ICA methods require that the signal measurements be whitened prior to source or component extraction [19, 20]. In whitening, a zero-mean m -dimensional vector signal $\mathbf{x}(n) = [x_1(n) \cdots x_m(n)]^T$, $n \geq 0$ is assumed to have a stationary full-rank autocorrelation matrix

$$\mathbf{R}_{\mathbf{x}\mathbf{x}} = E\{\mathbf{x}(n)\mathbf{x}^T(n)\}, \quad (1)$$

where $E\{\cdot\}$ denotes statistical expectation. The goal is to compute an $(n \times n)$ -dimensional matrix \mathbf{P} such that

$$\mathbf{v}(n) = \mathbf{P}\mathbf{x}(n) \quad (2)$$

contains uncorrelated elements with unit variance, or

$$E\{\mathbf{v}(n)\mathbf{v}^T(n)\} = \mathbf{P}\mathbf{R}_{\mathbf{xx}}\mathbf{P}^T = \mathbf{I}. \quad (3)$$

Whitening also improves the performance of gradient-based adaptive systems in other tasks [21, 22].

When employing block processing of the measured data, the most-straightforward way to calculate \mathbf{P} is by Cholesky factorization, where a lower triangular matrix $\mathbf{R}_{\mathbf{xx}}^{1/2}$ is computed from $\mathbf{R}_{\mathbf{xx}}$ such that

$$\mathbf{R}_{\mathbf{xx}}^{1/2}\mathbf{R}_{\mathbf{xx}}^{T/2} = \mathbf{R}_{\mathbf{xx}}. \quad (4)$$

Then, \mathbf{P} can be computed as

$$\mathbf{P} = \mathbf{R}_{\mathbf{xx}}^{-1/2}, \quad (5)$$

where the inverse of $\mathbf{R}_{\mathbf{xx}}^{1/2}$ is computed via backsubstitution. The value of \mathbf{P} generated is by definition lower-triangular, although \mathbf{P} need not be constrained to be lower triangular. In fact, any matrix \mathbf{P} that satisfies $\mathbf{P}\mathbf{R}_{\mathbf{xx}}\mathbf{P}^T = \mathbf{I}$ is a valid solution to the prewhitening task. Other block-based procedures for generating \mathbf{P} employ common linear algebra calculations such as the eigenvalue and singular-value decompositions [23].

In certain applications, it is desirable to have a simple procedure for iteratively calculating the value of $\mathbf{P}(n) = \mathbf{P}$ as measurements become available. Such a procedure has been developed and popularized in the BSS and ICA communities [22, 24, 25]. This adaptive whitening procedure is given by

$$\mathbf{P}(n+1) = \mathbf{P}(n) + \mu[\mathbf{P}(n) - \mathbf{v}(n)\mathbf{v}^T(n)\mathbf{P}(n)] \quad (6)$$

where μ is a small positive step size value. It can be shown (c.f. [26]) that $E\{\mathbf{P}(n)\}$ converges to the vicinity of the solution given by

$$E\{\mathbf{P}(n)\}\mathbf{R}_{\mathbf{xx}}E\{\mathbf{P}^T(n)\} \approx \mathbf{I}, \quad (7)$$

and the speed of convergence of the algorithm depends on μ . The initial value of $\mathbf{P}(0)$ must be a full-rank matrix, and typically a diagonal matrix is chosen.

The main drawback of the procedure in (6) is that it uses gradient adaptation to achieve its goal. Gradient methods are heuristic procedures in which careful attention must be paid to the step size value μ to maintain algorithm stability. The cubic nature of the update equation in (6) means that the initial value $\mathbf{P}(0)$ and μ interact to determine how fast the estimate $\mathbf{P}(n)$ converges from its initial value. If μ is too large, the algorithm diverges. Moreover, if any of the elements of the measured vector signal $\mathbf{x}(n)$ have impulsive distributions, the estimation capabilities of the procedure will not be robust.

2.2. Relationship to Recursive Least-Squares Estimation

In adaptive signal processing, whitening is a form of signal orthogonalization that has a number of useful properties. All recursive least-squares (RLS) methods in adaptive filtering, such as the QR-decomposition-based RLS adaptive filter, least-squares lattice filters, and fast transversal filters, exploit signal orthogonalization for performance, numerical, and/or computational advantages [28]. The fact that $\mathbf{P}(n)$ is an approximate orthogonalizing linear transformation of the measured data sequence $\mathbf{x}(n)$ suggests that there may

be a connection between the procedure in (6) and RLS estimation. Exploring this connection may lead to novel RLS algorithms with interesting forms and properties.

It turns out that the adaptive whitening method in (6) does have a least-squares equivalent procedure. A simple derivation of the algorithm is given in [27]. The updates for the coefficient matrix $\mathbf{P}(n)$ are

$$\mathbf{P}(n+1) = \frac{1}{\sqrt{\lambda}} (\mathbf{P}(n) - \zeta(n)\mathbf{v}(n)\mathbf{u}^T(n)) \quad (8)$$

$$\mathbf{v}(n) = \mathbf{P}(n)\mathbf{x}(n) \quad (9)$$

$$\mathbf{u}(n) = \mathbf{P}^T(n)\mathbf{v}(n) \quad (10)$$

$$\zeta(n) = \frac{1}{\lambda + \|\mathbf{v}(n)\|^2 + \sqrt{\lambda(\lambda + \|\mathbf{v}(n)\|^2)}}, \quad (11)$$

where λ is the so-called forgetting factor that lies in the range $0 \ll \lambda \leq 1$. The update for $\mathbf{P}(n)$ in (8)–(11) has a number of advantages over that in (6):

1. *The value of $\mathbf{P}(n)$ exactly maintains the whitening property $\mathbf{P}(n)\mathbf{R}_{\mathbf{xx}}(n)\mathbf{P}^T(n) = \mathbf{I}$ at every iteration up to the numerical precision of the computing environment, where*

$$\mathbf{R}_{\mathbf{xx}}(n+1) = \lambda^n\mathbf{R}_{\mathbf{xx}}(0) + \sum_{k=0}^n \lambda^{k-n}\mathbf{x}(k)\mathbf{x}^T(k), \quad (12)$$

and $\mathbf{R}(0)$ is the initial estimate of $\mathbf{R}_{\mathbf{xx}}$. Thus, convergence of the algorithm depends on how well $\mathbf{R}_{\mathbf{xx}}(n)$ represents an accurate estimate of $\mathbf{R}_{\mathbf{xx}}$ in (1). These convergence properties are not unlike those of RLS estimation procedures. In particular, for values of λ in the range $0 < \lambda < 1$, the initial convergence of the algorithm can be made fast by setting $\mathbf{P}(0)$ to a matrix with large singular values.

2. *The update for $\mathbf{P}(n)$ is numerically-robust.* It can be shown that $\mathbf{P}(n+1)$ is related to $\mathbf{P}(n)$ and a scaled version of $\mathbf{v}(n)$ through a Householder transformation. Moreover, a study of the numerical properties of the update verifies that it maintains the whitening condition $\mathbf{P}^T(n)\mathbf{P}(n) = \mathbf{R}_{\mathbf{xx}}^{-1}(n)$ in a numerically-stable manner. These nice numerical properties are due to the fact that the procedure is the core element within a square-root RLS algorithm, and such algorithms are known to have good numerical properties [28].

3. *The algorithm is guaranteed to be stable for any value of λ in the range $0 < \lambda \leq 1$.* In contrast, the stability of the update in (6) is a complicated function of μ and the initial whitening matrix $\mathbf{P}(0)$, and its stability cannot be guaranteed for fixed step size values.

The main drawbacks of (8)–(11) as compared to the gradient adaptive approach in (6) are: i) the former is more computationally-complex, and ii) the two algorithms will likely have similar tracking capabilities. Hence, the gradient adaptive whitening procedure in (6) worth considering if tracking performance is of paramount concern and if computational resources are at a premium.

While the connections between (6) and RLS estimation led this author to the discovery of the procedure in (8)–(11), it was originally discovered and used for RLS estimation by Rontogiannis and Theodoridis [29], who call it the Householder RLS algorithm. The conference publication [27] provides some additional extensions, including an algorithmic derivation that is easier to follow, a technique for iteratively updating $\mathbf{P}^{-1}(n)$ such that $\mathbf{P}^{-1}(n)\mathbf{P}^{-T}(n) = \mathbf{R}_{\mathbf{xx}}(n)$, a

simplified version of the algorithm in [29], and two additional RLS estimation procedures based on the whitening method. The fact remains, however, that the connection between (6) and (8)–(11) would likely have taken longer to discover if one were not working across the boundaries of the BSS, ICA, and adaptive signal processing fields.

For those who wish to study the Householder RLS algorithm, a version is included in a set of adaptive filtering routines within the Filter Design Toolbox for the MATLAB technical computing software package [30].

2.3. Other Adaptive Whitening Methods

Having made a connection between adaptive whitening and recursive least-squares estimation, we should consider this relationship more carefully to see if it can be exploited for further developments. We now show how adaptive signal processing can motivate new adaptive whitening procedures that may be better for BSS and ICA than (6) or (8)–(11).

All recursive least-squares estimation procedures calculate the same coefficient estimates in an evolving linear regression task. These procedures differ in the way they parametrize the problem. The best solutions use just enough unique parameters to solve the task. The procedures in (6) and (8)–(11) calculate a square ($m \times m$) matrix even though (3) and its least-squares counterpart specify only $m(m+1)/2$ unique constraints. Hence, both systems are over-parametrized. Numerical and statistical effects could cause the columns of $\mathbf{P}(n)$ to rotate or wander over time in both cases, such that $\mathbf{P}(n)$ never converges to a unique solution. For RLS estimation, these effects are more of a nuisance than a problem, because they are unobservable in the Kalman gain vector. For whitening in BSS and ICA, however, these effects are problematic, because the source separation procedure would have to track these changes using the higher- or lower-order statistics of $\mathbf{v}(n)$.

Fortunately, researchers in both gradient and RLS algorithms have identified procedures that do not pose this problem. These methods constrain $\mathbf{P}(n)$ to be lower- or upper-triangular so that it has exactly the right degrees of freedom to solve (3) or its least-squares equivalent. The most well-known RLS algorithm with this structure is the Gentleman-Kung systolic array, the details of which were published over 20 years ago [31]. A sequential gradient-based approach has also been developed [21].

Given that (6) and (8)–(11) are related, it should be possible to find or develop simple gradient adaptive whitening procedures that impose a lower-triangular structure on $\mathbf{P}(n)$ as inspired by existing methods in the literature. We now give three such simple whitening procedures, where $\mathbf{P}(n)$ is a lower-triangular matrix, $\mathbf{M}(n)$ is a lower triangular matrix with ones along the diagonal, $\mathbf{G}(n)$ is a diagonal matrix, $\text{tril}[\mathbf{v}\mathbf{v}^T]$ denotes the strictly lower triangular part of $\mathbf{v}\mathbf{v}^T$, and $\text{diag}[\mathbf{v}\mathbf{v}^T]$ is a diagonal matrix whose diagonal entries are the same as $\mathbf{v}\mathbf{v}^T$.

1. *Inverse Cholesky Whitening Algorithm:* After computing $\mathbf{v}(n)$ in (9),

$$\begin{aligned} \mathbf{P}(n+1) &= \mathbf{P}(n) + \mu (\mathbf{I} - \text{diag}[\mathbf{v}(n)\mathbf{v}^T(n)]) \mathbf{P}(n) \\ &\quad - 2\mu \text{tril}[\mathbf{v}(n)\mathbf{v}^T(n)] \mathbf{P}(n). \end{aligned} \quad (13)$$

2. *Inverse LU Whitening Algorithm:* Set $\mathbf{P}(n) = \mathbf{G}(n)\mathbf{M}(n)$ with

$$\mathbf{M}(n+1) = \mathbf{M}(n) - 2\mu \text{tril}[\mathbf{v}_M(n)\mathbf{v}_G^T(n)] \mathbf{M}(n) \quad (14)$$

$$\mathbf{G}(n+1) = (1 + \mu) \mathbf{G}(n) - \mu \text{diag}[\mathbf{v}(n)\mathbf{v}_G^T(n)] \quad (15)$$

$$\mathbf{v}_M(n) = \mathbf{M}(n)\mathbf{x}(n) \quad (16)$$

$$\mathbf{v}(n) = \mathbf{G}(n)\mathbf{v}_M(n) \quad (17)$$

$$\mathbf{v}_G(n) = \mathbf{G}(n)\mathbf{v}(n). \quad (18)$$

3. *Simplified Adaptive Escalator Algorithm:* This algorithm is a modification of the approach in [21] such that no divides are required; see [21] for the relationship between the parameters $\{a_{ij}(n)\}$ and $\mathbf{P}(n)$. Setting $v_{j1}(n) = x_j(n)$, $a_{ij}(0) = 0$, and $g_{jj}(0) > 0$ for $1 \leq j < i \leq m$,

for $j = 1$ to m do

$$v_{M,j}(n) = v_{jj}(n) \quad (19)$$

$$v_j(n) = g_{jj}(n)v_{M,j}(n) \quad (20)$$

$$v_{G,j}(n) = g_{jj}(n)v_j(n) \quad (21)$$

for $i = (j+1)$ to m do

$$v_{i(j+1)}(n) = v_{ij}(n) + a_{ij}(n)v_{M,j}(n) \quad (22)$$

$$a_{ij}(n+1) = a_{ij}(n) - 2\mu v_{G,j}(n)y_{i(j+1)}(n) \quad (23)$$

end

$$g_{jj}(n+1) = (1 + \mu)g_{jj}(n) - \mu y_{G,j}(n)y_j(n) \quad (24)$$

end

The primary advantage of the above approaches over (6) and (8)–(11) is the computational simplicity of each approach. Table 1 shows the total number of multiply/adds for each method, along with that of (6). The proposed methods all require fewer multiply/adds than (6) for $m \geq 2$.

2.4. Analysis and Simulations

It can be shown that the inverse Cholesky and inverse LU algorithms globally converge to a whitening solution satisfying (3). An analysis based upon the ordinary differential equation (ODE) method is provided in Appendix A. The analysis is similar to that used in [32] to analyze (6).

All of the proposed algorithms have better performance than that of (6). To illustrate this fact, we generated 100 sequences of $m = 6$ -dimensional jointly-Gaussian random vectors whose autocorrelation statistics are described by the eigenvalues $\{0.01, 0.1, 0.3, 1, 1.001, 5\}$ and random orthogonal eigenvectors. Each algorithm was applied to this data, where $\mathbf{W}(0) = \mathbf{M}(0) = \mathbf{G}(0) = \mathbf{I}$ and $\mu = 0.01$. For comparison, we also implemented a brute-force estimation method using

$$\hat{\mathbf{R}}_{\mathbf{x}\mathbf{x}}(n+1) = (1 - \mu)\hat{\mathbf{R}}_{\mathbf{x}\mathbf{x}}(n) + \mu \mathbf{x}(n)\mathbf{x}^T(n) \quad (25)$$

from which $\mathbf{W}(n)$ was computed as the inverse Cholesky factor of $\hat{\mathbf{R}}_{\mathbf{x}\mathbf{x}}(k)$ using MATLAB's `chol` function. The average values of the performance factors

$$\rho(n) = \frac{\|\text{diag}[\mathbf{W}(n)\mathbf{R}_{\mathbf{x}\mathbf{x}}\mathbf{W}^T(n)] - \mathbf{W}(n)\mathbf{R}_{\mathbf{x}\mathbf{x}}\mathbf{W}^T(n)\|_F^2}{2\|\text{diag}[\mathbf{W}(n)\mathbf{R}_{\mathbf{x}\mathbf{x}}\mathbf{W}^T(n)]\|_F^2} \quad (26)$$

$$\eta(n) = \frac{\|\mathbf{I} - \text{diag}[\mathbf{W}(n)\mathbf{R}_{\mathbf{x}\mathbf{x}}\mathbf{W}^T(n)]\|_F^2}{\|\text{diag}[\mathbf{W}(n)\mathbf{R}_{\mathbf{x}\mathbf{x}}\mathbf{W}^T(n)]\|_F^2} \quad (27)$$

were then computed for each estimate using the 100 sequences. Shown in Fig. 1 are the behaviors of $E\{\rho(n)\}$ and $E\{\eta(n)\}$ for each algorithm, in which the inverse Cholesky, simplified escalator, and brute-force methods all exhibit similar behavior. The inverse LU algorithm performs the best in this situation. All of the proposed methods are simpler than (6) or the brute-force method.

Table 1: Adaptive whitening algorithm complexities (number of multiply/adds per update)

Natural Gradient	Inverse Cholesky	Inverse LU	Simplified Escalator
$4m^2 + m$	$2m^2 + 2m$	$1.5m^2 + 4.5m$	$m^2 + 4m$

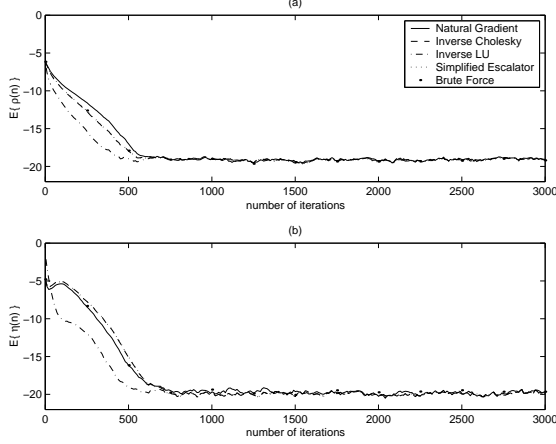


Fig. 1: Evolution of (a) $E\{\rho(n)\}$ and (b) $E\{\eta(n)\}$ for the five algorithms in the example.

2.5. Summary

The adaptive whitening task that is a precursor to many BSS and ICA approaches has direct connections with other adaptive signal processing tasks and recursive least-squares estimation in particular. This connection has allowed us to discover novel algorithms for RLS estimation as well as simplified adaptive whitening approaches for BSS and ICA. Simulations show that the new simplified whitening approaches are competitive with existing methods.

3. BSS, ICA, AND ADAPTIVE SUBSPACE ANALYSIS

3.1. Equivariant Adaptive Separation by Independence

One of the seminal and oft-cited works on blind source separation is that of Cardoso and Laheld [25] who examine procedures that combine adaptive whitening with contrast optimization [12]. These authors recognized that the whitening algorithm in (6) is over-parametrized and thus could be modified to produce a simple BSS and ICA procedure: Equivariant Adaptive Separation via Independence (EASI).

The EASI algorithm is first described. Let $\mathbf{x}(n)$ be an m -dimensional measured linear mixture of m sources. A demixing matrix $\mathbf{B}(n)$ is used to estimate the m -element output signal vector

$$\mathbf{y}(n) = \mathbf{B}(n)\mathbf{x}(n). \quad (28)$$

The matrix $\mathbf{B}(n)$ is then updated as

$$\mathbf{B}(n+1) = \mathbf{B}(n) + \mu_1(n)[\mathbf{I} - \mathbf{y}(n)\mathbf{y}^T(n)]\mathbf{B}(n) + \mu_2(n)[\mathbf{y}(n)\mathbf{f}^T(\mathbf{y}(n)) - \mathbf{f}(\mathbf{y}(n))\mathbf{y}^T(n)]\mathbf{B}(n), \quad (29)$$

where $\mu_1(n)$ and $\mu_2(n)$ are time-varying step sizes, $\mathbf{f}(\mathbf{y}) = [f_1(y_1) \cdots f_m(y_m)]^T$ is a vector-valued nonlinearity, and each scalar nonlinearity $f_i(y_i)$ depends on the statistics of the extracted source in $y_i(n)$. By proper choice of each $f_i(\mathbf{y})$, (29) adjusts $\mathbf{W}(n)$ such that each $y_i(n)$ contains a single independent source from a linear mixture without replacement. When the probability density functions

(p.d.f.'s) $p_{s_j}(s_j)$ of the sources are known and differentiable, then the best choice for each $f_i(\mathbf{y})$ is $-p'_{s_j}(\mathbf{y})/p_{s_j}(\mathbf{y})$, where $p'_{s_j}(\mathbf{y})$ is the derivative of $p_{s_j}(\mathbf{y})$ and $y_i(n)$ contains a scaled estimate of the source $s_j(n)$ at convergence. Other choices for the nonlinearities $f_i(\mathbf{y})$ can also separate the sources so long as the proper signs of the nonlinearities are chosen. In very few cases, however, is a separating result guaranteed.

3.2. The EASI Algorithm and Constrained Adaptation

The EASI algorithm contains two terms on the right-hand side of (29) that determine the algorithm's evolutionary behavior. These terms have special constraint properties that are worth exploring. We now provide a description of the algorithm that illustrates the special properties of these terms. This description employs the ordinary differential equation (ODE) form of the update in (29), as given by

$$\frac{d\mathbf{B}}{dt} = [\mathbf{I} - \mathbf{B}\mathbf{R}_{\mathbf{x}\mathbf{x}}\mathbf{B}^T]\mathbf{B} + [\overline{\mathbf{G}}\mathbf{B}^T - \mathbf{B}\overline{\mathbf{G}}^T]\mathbf{B}, \quad (30)$$

$$\overline{\mathbf{G}} = -E\{\mathbf{f}(\mathbf{y})\mathbf{x}^T\} = \frac{\partial}{\partial \mathbf{B}} \sum_{i=1}^m E\{g_i(y_i)\} \quad (31)$$

and $\partial g_i(\mathbf{y})/\partial \mathbf{y} = \mathbf{f}_i(\mathbf{y})$. Factor \mathbf{B} into two matrices \mathbf{W} and \mathbf{P} such that

$$\mathbf{B} = \mathbf{W}\mathbf{P}. \quad (32)$$

Assume without loss of generality that \mathbf{W} is initially orthogonal; *i.e.* $\mathbf{W}^T(0)\mathbf{W}(0) = \mathbf{W}(0)\mathbf{W}^T(0) = \mathbf{I}$. The following theorem and corollary describe the properties of \mathbf{P} and \mathbf{W} , the proofs of which can be found in Appendix B.

Theorem 1: *The ODE in (30) can be expressed as $\mathbf{B} = \mathbf{W}\mathbf{P}$, where \mathbf{W} and \mathbf{P} evolve as*

$$\frac{d\mathbf{P}}{dt} = \mathbf{P} - \mathbf{P}\mathbf{R}_{\mathbf{x}\mathbf{x}}\mathbf{P}^T\mathbf{P} \quad (33)$$

$$\frac{d\mathbf{W}}{dt} = \mathbf{G}\mathbf{W}^T\mathbf{W} - \mathbf{W}\mathbf{G}^T\mathbf{W} \quad (34)$$

and $\mathbf{G} = \overline{\mathbf{G}}\mathbf{P}^T$.

Corollary 1.1: *$\mathbf{W}(t)$ is orthonormal for all $t \geq 0$.*

These results show that the EASI algorithm in (29) is asymptotically-equivalent to a cascade of two subsystems. The first subsystem performs whitening using (6) with step size sequence $\mu_1(n)$, yielding the whitened signal vector sequence $\mathbf{v}(n)$. This sequence is then processed by the second subsystem, in which $\mathbf{y}(n) = \mathbf{W}(n)\mathbf{v}(n)$ and the coefficient matrix $\mathbf{W}(n)$ is updated as

$$\mathbf{W}(n+1) = \mathbf{W}(n) + \mu_2(n)[\mathbf{y}(n)\mathbf{f}^T(\mathbf{y}(n)) - \mathbf{f}(\mathbf{y}(n))\mathbf{y}^T(n)]\mathbf{W}(n). \quad (35)$$

3.3. Relationship to Adaptive Subspace Analysis

The update for $\mathbf{W}(n)$ in (35) and its associated ODE is closely related to recent work in linear algebra dealing with the estimation of eigenvectors and subspaces [33, 34]. These efforts have focused on the underlying structure of

the parameter spaces and on developing simple but powerful methods for computing gradient and Newton updates within these parameter spaces. From this work, the following two gradient-based differential updates have been developed:

- *Gradient update on the Stiefel manifold:*

$$\frac{d\mathbf{W}}{dt} = \mathbf{W}\mathbf{W}^T\mathbf{G} - \mathbf{W}\mathbf{G}^T\mathbf{W}, \quad (36)$$

where $\mathbf{G} = \partial\mathcal{J}(\mathbf{W})/\partial\mathbf{W}$ is the gradient of an inhomogeneous cost function $\mathcal{J}(\mathbf{W})$ satisfying $\mathcal{J}(\mathbf{W}) \neq \mathcal{J}(\mathbf{Q}\mathbf{W})$ for an arbitrary $(p \times p)$ orthogonal matrix \mathbf{Q} .

- *Gradient update on the Grassmann manifold:*

$$\frac{d\mathbf{W}}{dt} = \mathbf{W}\mathbf{W}^T\mathbf{G} - \mathbf{G}\mathbf{W}^T\mathbf{W}, \quad (37)$$

where $\mathbf{G} = \partial\mathcal{J}(\mathbf{W})/\partial\mathbf{W}$ is the gradient of a homogeneous cost function $\mathcal{J}(\mathbf{W})$ satisfying $\mathcal{J}(\mathbf{W}) = \mathcal{J}(\mathbf{Q}\mathbf{W})$ for all $(p \times p)$ orthogonal matrices \mathbf{Q} .

These updates are named after the researchers who first studied the corresponding parameter spaces and their geometric properties [35, 36]. It is easy to show that both (36) and (37) maintain the orthonormality of $\mathbf{W}(t)$ if $\mathbf{W}(0)$ is orthonormal, because $d[\mathbf{W}\mathbf{W}^T]/dt = \mathbf{0}$ for both updates. These algorithms compute gradient flows on spaces of orthonormal matrices without requiring a minimal parametrization of the matrix estimate.

The updates in (36) and (37) are useful starting points for algorithm design, and existing approaches can often be easily related to them. For example, the principal subspace rule of Oja and Karhunen [37] updates $\mathbf{W}(n)$ as

$$\begin{aligned} \mathbf{W}(n+1) &= \mathbf{W}(n) + \mu\mathbf{W}(n)\mathbf{x}(n)\mathbf{x}^T(n)[\mathbf{I} \\ &\quad - \mathbf{W}^T(n)\mathbf{W}(n)]. \end{aligned} \quad (38)$$

This coefficient update has the corresponding ODE

$$\frac{d\mathbf{W}}{dt} = \mathbf{G} - \mathbf{G}\mathbf{W}^T\mathbf{W}, \quad (39)$$

where $\mathbf{G} = E\{\mathbf{W}\mathbf{x}\mathbf{x}^T\}$ is the gradient of the cost function $\mathcal{J}(\mathbf{W}) = 0.5\|\mathbf{W}\mathbf{x}\|^2$. The ODE in (39) is identical to (37) except for the omission of the leading $\mathbf{W}\mathbf{W}^T$ matrix on the first term of (37). Since $\mathbf{W}\mathbf{W}^T$ is constant, the behaviors of both ODEs are identical.

When we compare the gradient update on the Stiefel manifold in (36) with the ODE in (34), we notice that they are strikingly similar. It is straightforward to show that (34) maintains $\mathbf{W}^T\mathbf{W}$, and not $\mathbf{W}\mathbf{W}^T$, at a constant value. If $\mathbf{W}(0)$ is also orthonormal and square, the two differential updates produce the same matrix flows.

3.4. Numerical Stability Issues in Prewhitened BSS

All practical adaptive algorithms are written in terms of finite-difference equations or updates and not differential equations. The classic way of turning differential equations into useful coefficient updates is by finite differences, *i.e.* by replacing $d\mathbf{W}/dt$ terms with $(\mathbf{W}(n+1) - \mathbf{W}(n))/\mu$ terms. For (34), (36), and (37), however, such approximations can lead to numerical difficulties. Accumulation of numerical errors in $\mathbf{W}(n)$ from the discretization process can cause the value of $\mathbf{W}(n)$ to “wander away” from its orthonormal

constraint space. Over time, the performance of the adaptive procedure can be compromised, and it can cause the algorithm to fail catastrophically. This numerical behaviors of these discrete-time updates can only be studied when a particular gradient matrix $\mathbf{G}(n)$ is chosen for the updates, as $\mathbf{G}(n)$ usually depends on $\mathbf{W}(n)$. Hence, the analysis is problem-specific.

In [38], discrete-time versions of the update in (36) are studied in the case where $\mathbf{G} = E\{\mathbf{f}(\mathbf{y})\mathbf{v}^T\}$ and $\mathbf{y} = \mathbf{W}\mathbf{v}$ is chosen for the prewhitened BSS task. It is shown that the update is marginally-stable within the constraint space, such that $\mathbf{W}\mathbf{W}^T$ slowly diverges from orthonormality over time. Hence, this update in its unmodified form is not appropriate for BSS and ICA tasks. Similarly, the update in (35) is not appropriate when used to adapt $\mathbf{W}(n)$ separately from $\mathbf{P}(n)$.

To fix these numerical problems, we can explore variations to the marginally-stable updates in (34), (36), and (37) that have identical behavior in their differential form but better numerical performance when they are discretized. A comparison between (37) and (39) for the Grassmann manifold suggest a possible cure:

When $\mathbf{W}(0)$ is orthonormal, the matrix $\mathbf{W}\mathbf{W}^T$ (and $\mathbf{W}^T\mathbf{W}$ if \mathbf{W} is square ($p = m$)) can be added to or removed from terms within the updates without changing the behaviors of the associated ODEs.

Since $\mathbf{W}\mathbf{W}^T = \mathbf{I}$, the values of these terms will be unchanged after the modifications. We can develop numerous equivalent ODEs for the purposes of algorithm design using this idea. These ODEs include

$$\frac{d\mathbf{W}}{dt} = \mathbf{G} - \mathbf{W}\mathbf{G}^T\mathbf{W} \quad (40)$$

$$\frac{d\mathbf{W}}{dt} = \mathbf{W}\mathbf{W}^T\mathbf{W}\mathbf{W}^T\mathbf{G} - \mathbf{W}\mathbf{G}^T\mathbf{W} \quad (41)$$

if \mathbf{W} is non-square. If \mathbf{W} is square, then other variants include

$$\frac{d\mathbf{W}}{dt} = \mathbf{W}\mathbf{W}^T\mathbf{G}\mathbf{W}^T\mathbf{W} - \mathbf{W}\mathbf{G}^T\mathbf{W} \quad (42)$$

$$\frac{d\mathbf{W}}{dt} = \mathbf{G}\mathbf{W}^T\mathbf{W}\mathbf{W}^T\mathbf{W} - \mathbf{W}\mathbf{G}^T\mathbf{W} \quad (43)$$

In [38], several of these algorithm variants are considered for the prewhitened BSS task where \mathbf{W} is square, where it is shown that the design of the algorithm is linked to the interaction between the amplitude statistics of the sources being extracted and the nonlinearities contained with $\mathbf{f}(\mathbf{y})$. In particular, when each $f_i(\mathbf{y})$ satisfies the sign constraint $\text{sgn}[f_i(\mathbf{y})] = \text{sgn}[y]$, the sign of the term

$$\kappa_i = E\{y_i(k)f_i(y_i(k))\} - E\{y_i^2(k)\}E\{f_i'(y_i(k))\} \quad (44)$$

determines which ODE should be chosen for discretization. If $\kappa_i > 0$ for all $1 \leq i \leq m$, then (40) is an appropriate choice. Similarly, if $\kappa_i < 0$, then (41), (42), or (43) can be chosen. The final choice of algorithm form can be motivated by other concerns, such as overall computational complexity.

3.5. Numerical Stability Issues in Adaptive Subspace Analysis

The BSS and ICA methods presented above also have yielded novel methods for minor subspace analysis. The

goal of minor subspace analysis is to minimize the average energy in the vector sequence $\mathbf{W}(n)\mathbf{x}(n)$ subject to the orthogonality constraint $\mathbf{W}(n)\mathbf{W}^T(n) = \mathbf{I}$. Such algorithms are useful for array processing, parameter estimation, and harmonic retrieval tasks in signal processing [39, 40, 41]. The design of algorithms for the minor subspace analysis task have proven to be problematic due to numerical difficulties associated with maintaining the orthonormality of the subspace matrix estimate [23]. The design of such algorithms should be motivated by the same design constraints that make (38) useful for the principal subspace analysis task:

1. The algorithm should be computationally-simple, involving multiplies and adds that scale according to the size of the matrix $\mathbf{W}(n)$.
2. The algorithm should not require an orthogonalization step or a normalization step; it should maintain $\mathbf{W}(n)\mathbf{W}^T(n) \approx \mathbf{I}$ through its adaptive behavior
3. The algorithm should converge to a solution satisfying the minor subspace analysis task.

These design constraints were employed in [42] to develop the following minor subspace analysis algorithm:

$$\mathbf{W}(n+1) = \mathbf{W}(n) + \mu[\mathbf{y}(n)\mathbf{y}^T(n)\mathbf{W}(n) - \mathbf{W}(n)\mathbf{W}^T(n)\mathbf{W}(n)\mathbf{W}^T(n)\mathbf{y}(n)\mathbf{x}^T(n)] \quad (45)$$

$$\mathbf{y}(n) = \mathbf{W}(n)\mathbf{x}(n), \quad (46)$$

where $\mu > 0$. It can be shown via analysis that

- the update converges to the p -dimensional minor subspace of $\mathbf{R}_{\mathbf{x}\mathbf{x}}$,
- the update is locally-stable to the minor subspace; and
- the update maintains $\mathbf{W}(n)\mathbf{W}^T(n) \approx \mathbf{I}$ in a numerically-stable manner.

It uses about 7 multiply/adds for each element in $\mathbf{W}(n)$.

It is this author's conjecture that the algorithm in (45)–(46) represents the simplest minor subspace analysis algorithm that does not require any operations beyond multiplications and additions to implement. It has already appeared in other publications for performance comparison purposes [43, 44]. These comparisons are similar to those in adaptive filtering, where all researchers are motivated to compare their adaptive procedures with the ubiquitous least-mean-square (LMS) algorithm due to the latter algorithm's simplicity and robustness. The design of this minor subspace analysis rule would not have happened without the connections made between ideas in BSS [12, 19, 25] and in subspace analysis [34, 37].

3.6. Alternative Constraints

The algorithmic ideas described above offer new and largely-unexplored opportunities for the design of adaptive signal processing algorithms. One such extension is now considered.

The ODEs in (36) and (37) satisfy $d[\mathbf{W}\mathbf{W}^T]/dt = \mathbf{0}$, as previously mentioned. Hence, they maintain $\mathbf{W}(t)\mathbf{W}^T(t) = \mathbf{W}(0)\mathbf{W}^T(0)$ for all $t \geq 0$. These constraints are independent of the initial matrix $\mathbf{W}(0)$. If $\mathbf{W}(0)$ is chosen to be orthogonal, then $\mathbf{W}(t)$ is orthogonal. This choice is not unique, however, and *any* initial value $\mathbf{W}(0)$ could be chosen. For example, if

$$\mathbf{W}(t)\mathbf{W}^T(t) = \mathbf{C} \quad (47)$$

is desired for some $(p \times p)$, non-singular, symmetric constraint matrix \mathbf{C} , we only need to choose $\mathbf{W}(0)$ such that $\mathbf{W}(0)\mathbf{W}^T(0) = \mathbf{C}$. The ODEs will then implicitly maintain (47) throughout adaptation.

The above idea allows for the design of adaptive algorithms that maintain arbitrary constraints on the matrix $\mathbf{W}(n)\mathbf{W}^T(n)$ or $\mathbf{W}^T(n)\mathbf{W}(n)$ through their adaptation. It is unclear how useful such algorithms might be, as non-orthogonality constraints are not normally considered in signal processing tasks. Even so, it shows the richness of the theory and possible avenues for exploration should such constraints arise naturally within a particular problem. Extensions of existing subspace analysis algorithms to these non-orthogonal constraints are relatively simple to develop. For example, a modified version of the principal subspace rule in (38) that maintains $\mathbf{W}(n)\mathbf{W}^T(n) = \mathbf{C}$ is given by

$$\mathbf{W}(n+1) = \mathbf{W}(n) + \mu[\mathbf{C}\mathbf{y}(n)\mathbf{x}^T(n) - \mathbf{y}(n)\mathbf{y}^T(n)\mathbf{W}] \quad (48)$$

where $\mathbf{y}(n) = \mathbf{W}(n)\mathbf{x}(n)$. Informal simulations of the above algorithm for various sequences $\mathbf{x}(n)$ with different auto-correlation statistics show that the rows of $\mathbf{W}(n)$ converge to the p -dimensional principal subspace of $\mathbf{x}(n)$ in a stable manner, and $\mathbf{W}(n)\mathbf{W}^T(n) \approx \mathbf{C}$ at convergence. The above algorithm also contains key ideas that lead to novel approaches for phase-only adaptive finite-impulse-response (FIR) filters [45, 46], as discussed in the next section.

3.7. Summary

The EASI algorithm combines a well-known signal whitening algorithm with an update that maintains the orthogonality of a square matrix through its adaptive behavior. Similar ideas have appeared in techniques for adaptively estimating eigenvectors and subspaces for array processing tasks. A careful study of these methods indicate that their numerical behaviors may differ in practical implementations, but they can be modified to prevent the resulting instabilities. A novel minor subspace analysis algorithm has been developed as a result, and adaptive algorithms involving non-orthogonal matrix constraints can also be derived.

4. BSS, ICA, AND PHASE-ONLY ADAPTIVE FILTERING

4.1. Spatio-temporal Extensions of BSS and ICA

One of the oft-cited potential applications of BSS and ICA is the so-called ‘‘cocktail party problem,’’ which is a term coined from the cocktail party effect in hearing research [47]. Human listeners have an uncanny ability to focus their attention on the speech of a chosen talker amongst several talkers within a reverberant environment. It is well-known that using more than one sensor—our two ears—is an important component of this ability. Our brains use the time-of-arrival and amplitude differences in the signals received by our ears to decode the information and understand the speech in the midst of jamming signals—other talkers—in the room. One can test this fact simply by covering up one ear while attempting to listen to a talker in a noisy environment; speech understanding becomes increasingly more difficult. Those who have any hearing loss in either ear will have difficulty in understanding others unless an assistive device is used.

Speech signals have statistical properties that vary from time instant to time instant for a single talker and from talker to talker. Over reasonably short time frames, say from the length of a phoneme (~ 0.2 s) to an entire phrase (~ 4 s), speech signals from two different talkers usually have

no significant statistical relationship. These facts suggest that BSS and ICA could be used to separate speech signals. In this case, the mixing system is determined by the multipath propagation of the room from each talker to several microphones located in the room. The signals from the microphones would be processed to obtain the speech of each individual talker.

The original formulation to BSS and ICA involves spatial mixtures in which no structural relationship between the individual signals is assumed. In other words, the mixing matrix is arbitrary. The acoustic mixing scenario described above involves sources that have certain temporal relationships due to the facts that (a) speech signals are correlated in time and (b) the room imposes a structural relationship on the reflection times-of-arrival. Because normal sound propagation is linear, these mixing conditions can be represented by a multichannel convolution process. It is beneficial to maintain such structural relationships in the separation system by using a multichannel convolution system of the form

$$\mathbf{y}(n) = \sum_{l=0}^L \mathbf{W}_l(n) \mathbf{x}(n-l) \quad (49)$$

where $\{\mathbf{W}_l(n)\}$, $0 \leq l \leq m$ is a set of L ($m \times m$) demixing coefficient matrices. Here, we have assumed a causal FIR demixing model for reasons of implementation. The goal is to adjust each $\mathbf{W}_l(n)$ over time so that the sequence $\mathbf{y}(n)$ contains samples from each speech signal in the environment, with perhaps a delay and an arbitrary filtering operation on each sequence $y_i(n)$.

The connection between standard BSS and ICA tasks and the speech separation task has led researchers to look for extensions of BSS and ICA approaches to solve the speech separation task. Any one of several design procedures could be used. For example, the problem could be studied in the frequency domain, in which the separation system is parametrized by $m^2(L+1)$ bin values corresponding to the multichannel FIR system in (49) in the discrete Fourier transform (DFT) domain [48, 49, 50]. Alternatively, a time-domain adjustment procedure could be developed, in which a logical extension of the single-matrix updates of existing BSS and ICA algorithms are extended to the coefficients of the FIR demixing system. In the latter case, several researchers have established rules by which such extensions can be made, as starting points for algorithm design [48, 51]. These rules can be paraphrased as follows:

- Matrix multiplication in the spatial BSS or ICA task is equivalent to multichannel convolution of matrix sequences in the spatio-temporal BSS task.
- Matrix addition in the spatial BSS or ICA task is equivalent to element-by-element addition of matrix sequences in the spatio-temporal BSS task.
- Matrix transposition in the spatial BSS or ICA task is equivalent to transposition and time-reversal of matrix sequences in the spatio-temporal BSS task.

It should be noted that these rules generate algorithms that are merely starting points for a working solution, and the behaviors of the algorithms so obtained must be studied to determine their convergence, stability, and statistical behaviors.

4.2. Extensions to Phase-Only Adaptive Filters

The procedures by which spatial-only BSS algorithms are extended to spatio-temporal BSS tasks have a number of useful and interesting applications in non-blind adaptive signal processing. One such application is in phase-only adaptive filtering. In this task, the goal is to adjust the coefficients of a single-channel FIR filter such that

- the magnitude response of the filter remains constant, and
- the phase response of the filter is adjusted to meet a fidelity criterion.

In essence, the proposed procedure would allow one to decouple the way in which the magnitude response and the phase response of an adaptive filter is specified, making only the phase response adaptive.

Such procedures could have a number of applications. For example, in almost every communications task, the medium or channel used to carry the signals of interest spreads out the waveforms in time. Usually, one is only interested in signals over a specified frequency range. A phase-only adaptive filter could be used to phase-align the signals at the receiver only over the frequency band of interest. This solution may be more appropriate than a traditional equalizer that compensates for both amplitude and phase distortions, because (i) traditional equalizers require that the signals being sent have equal energy across the frequency band of interest, and (ii) traditional equalizers attempt to compensate for the entire frequency band and not just a band of interest.

The phase-only adaptive filtering task appears to have connections to the constrained adaptation procedures exemplified by the EASI algorithm in (29) as well as the Stiefel and Grassmann manifold updates in (36) and (37). It is beneficial, therefore, to consider extensions of one or more of these methods to the single-channel filtering task, in which a single-input, single-output system with scalar coefficients $\{w_l\}$ is defined. By applying the three above rules to the Stiefel manifold update, the following generic ODE is obtained:

$$\frac{dw_l}{dt} = w_l * w_{-l} * g_l - w_l * g_{-l} * w_l \quad (50)$$

where ‘*’ denotes the discrete-time convolution operation

$$w_l * g_l = \sum_{i=-\infty}^{\infty} w_l g_{l-i}. \quad (51)$$

In this algorithm, a doubly-infinite impulse response for w_l is assumed, and filter truncation is required in order to approximately implement it for FIR systems [52, 45, 46].

The sequence g_l can be the scaled gradient $\mu \partial \mathcal{J}(\{w_l\}) / \partial w_l$ of any appropriately-chosen ensemble-averaged cost function for the parameter sequence $\{w_l\}$. Two possibilities for the phase only adaptive filtering task assume that a desired response signal $d(n)$ is available for training purposes. They are

- *Error minimization:*

$$\mathcal{J}(\{w_l\}) = E\{[d(n) - y(n)]^2\} \quad (52)$$

where $y(n)$ is the adaptive filter output given by

$$y(n) = \sum_{l=-\infty}^{\infty} w_l x(n-l) \quad (53)$$

and $\mu < 0$.

- *Correlation maximization:*

$$\mathcal{J}(\{w_l\}) = E\{d(n)y(n)\}, \quad (54)$$

where $\mu > 0$.

The differential update in (50) has a number of interesting properties that are analogous to those of the Stiefel manifold ODE in (36):

1. When $w_l(t)$ is chosen to be an *allpass* impulse response satisfying

$$\sum_{i=-\infty}^{\infty} w_i(t)w_{l+i}(t) = \delta_l \quad (55)$$

for $t = 0$, where δ_l is the Kronecker impulse function, (50) maintains the constraint in (55) for all $t \geq 0$. Thus, the adaptive filter maintains an allpass filter characteristic $|W(\omega)| = 1$, where $W(\omega)$ is the discrete-time Fourier transform of w_l .

2. If $w_l(t)$ initially satisfies

$$\sum_{i=-\infty}^{\infty} w_i(t)w_{l+i}(t) = c_l \quad (56)$$

for $t = 0$, where c_l is any symmetric impulse response, (50) maintains the constraint in (56) for all $t \geq 0$. Thus, the adaptive filter maintains the magnitude response characteristics defined by $|W(\omega)|^2 = C(\omega)$, where $C(\omega)$ is the discrete-time Fourier transform of c_l .

As in previous cases, the ODE in (50) represents a starting point for design, and suitable approximations and modifications are required to obtain a working and implementable adaptive filtering update.

The algorithms so obtained from these design efforts provide unique capabilities that heretofore have not been available in the adaptive signal processing community. An example drawn from [46] illustrates this fact. Suppose one wishes to adapt the coefficients of a filter such that (a) its magnitude response matches that of a target filter with impulse response f_l , and (b) its phase response matches the phase relationship between two statistically-stationary random signals $d(n)$ and $x(n)$. Define

$$d_f(n) = f_n * d(n) \quad (57)$$

$$x_f(n) = f_n * x(n) \quad (58)$$

Then, a numerically-stable gradient-based algorithm for solving this task can be derived from the generic update in (50) and is given by

$$w_l(n+1) = w_l(n) + \mu[d_f(n-L)x_f(n-L-l) - y(n-L)u(n-l)] \quad (59)$$

$$y(n) = \sum_{i=0}^L w_i(n)x(n-l) \quad (60)$$

$$u(n) = \sum_{q=0}^L w_{L-q}(n)d(n-q) \quad (61)$$

where $\mu > 0$. Further details about this algorithm, as well as its complex-valued extension, can be found in [46].

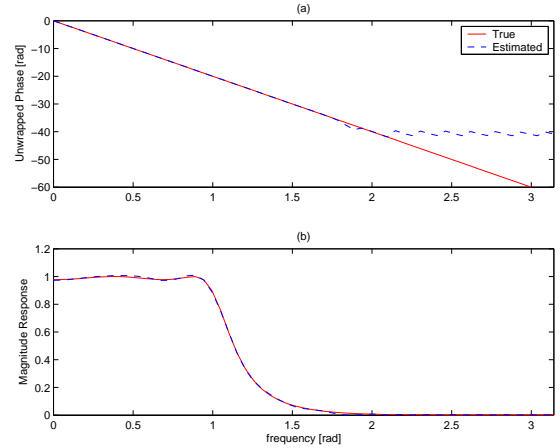


Fig. 2: Comparison of the (a) unwrapped phases and (b) magnitude responses for the true and estimated filters.

4.3. Simulations

To show that the above algorithm works as designed, we consider a simulation example drawn from [46]. In these simulations, the filter $F(\omega)$ is a fourth-order Chebyshev Type 1 lowpass filter with a bandwidth of 0.15π and 0.2dB of passband ripple. We let $x(n)$ and $\nu(n)$ be zero-mean uncorrelated Gaussian signals with $E\{|x(n)|^2\} = 1$ and $E\{|\nu(n)|^2\} = 0.01$, and define $d(n) = x(n-D) + \nu(n)$, where $D = 20$. With these choices, the goal is to adapt $\{w_l(n)\}$ such that the filter's steady-state magnitude response matches the Chebyshev lowpass filter, whereas its phase response is linear. The adaptive filter's parameters are $L = 39$, $\mu = 0.001$, and $w_l(0) = 0$. Fig. 2(a) and (b) show the magnitude and phase responses of the filter after a single simulation run of 30000 iterations. As can be seen, $\angle W(\omega)$ is linear over the passband, and $|W(\omega)| \approx |F(\omega)|$ over the entire frequency range. Monte Carlo simulations of the procedure verify that this convergence behavior is typical of the algorithm in this case.

4.4. Summary

Spatio-temporal extensions of BSS and ICA methods have led to novel procedures for speech separation in reverberant room environments. The methods used in these extensions can be used to develop novel adaptive filtering algorithms in other contexts. As an example, a novel phase-only adaptive FIR filtering algorithm is described that has been derived from a procedure for gradient adaptation on the Stiefel manifold. Simulations show that the adaptive filter's phase response can be adapted independently from its magnitude response using a time-domain algorithm.

5. CONCLUSIONS

Blind source separation and independent component analysis are two research areas in which novel ideas and techniques continue to be developed and tested. The influence of this work, however, is not limited just to BSS and ICA problems. This paper provides several examples of adaptive signal processing procedures whose development was spurred by BSS and ICA research. The diversity in backgrounds of those currently working in the BSS and ICA fields suggests that such cross-fertilization of tools and ideas will continue well into the future.

6. APPENDIX A

In this appendix, we prove that the inverse Cholesky and inverse LU whitening algorithms in (13) and (14)–(18) are globally convergent using the ODE method [32]. To analyze (13), we study its associated averaged ODE

$$\frac{d\mathbf{P}}{dt} = \mathbf{P} - \text{diag}[\mathbf{P}\mathbf{R}_{\mathbf{xx}}\mathbf{P}^T]\mathbf{P} - 2\underline{\text{tril}}[\mathbf{P}\mathbf{R}_{\mathbf{xx}}\mathbf{P}^T]\mathbf{P}. \quad (62)$$

The following theorem describes the convergence properties of (62).

Theorem 2: *The ODE in (62) converges to the inverse Cholesky factor satisfying*

$$\lim_{t \rightarrow \infty} \mathbf{P}(t)\mathbf{R}_{\mathbf{xx}}\mathbf{P}^T(t) = \mathbf{I} \quad (63)$$

for an arbitrary non-singular lower-triangular matrix $\mathbf{P}(0)$ and symmetric positive definite $\mathbf{R}_{\mathbf{xx}}$.

Proof: The proof of the relation is by induction. Consider the matrix $\mathbf{C}(t) = \mathbf{P}(t)\mathbf{R}_{\mathbf{xx}}\mathbf{P}^T(t)$ whose entries are $\{c_{ij}(t)\}$. Because $\mathbf{C}(t)$ is symmetric, $c_{ij}(t) = c_{ji}(t)$ for all $1 \leq j < i \leq m$, such that we only need to consider the convergence of $\text{tril}[\mathbf{C}(t)]$. Since $\mathbf{R}_{\mathbf{xx}}$ is positive definite, $c_{ii}(t) > 0$ for an arbitrary non-singular lower-triangular matrix $\mathbf{P}(t)$. The evolution of $\mathbf{C}(t)$ is described by

$$\frac{d\mathbf{C}}{dt} = \frac{d\mathbf{P}}{dt}\mathbf{R}_{\mathbf{xx}}\mathbf{P}^T + \mathbf{P}\mathbf{R}_{\mathbf{xx}}\frac{d\mathbf{P}^T}{dt} \quad (64)$$

$$= 2\mathbf{P}\mathbf{R}_{\mathbf{xx}}\mathbf{P}^T - (\text{diag}[\mathbf{P}\mathbf{R}_{\mathbf{xx}}\mathbf{P}^T]\mathbf{P}\mathbf{R}_{\mathbf{xx}}\mathbf{P}^T + \mathbf{P}\mathbf{R}_{\mathbf{xx}}\mathbf{P}^T \text{diag}[\mathbf{P}\mathbf{R}_{\mathbf{xx}}\mathbf{P}^T]) - 2\underline{\text{tril}}[\mathbf{P}\mathbf{R}_{\mathbf{xx}}\mathbf{P}^T]\mathbf{P}\mathbf{R}_{\mathbf{xx}}\mathbf{P}^T - 2\mathbf{P}\mathbf{R}_{\mathbf{xx}}\mathbf{P}^T (\underline{\text{tril}}[\mathbf{P}\mathbf{R}_{\mathbf{xx}}\mathbf{P}^T])^T \quad (65)$$

$$= 2\mathbf{C} - (\text{diag}[\mathbf{C}]\mathbf{C} + \mathbf{C}\text{diag}[\mathbf{C}]) - 2\underline{\text{tril}}[\mathbf{C}]\mathbf{C} - 2\mathbf{C}(\underline{\text{tril}}[\mathbf{C}])^T. \quad (66)$$

The evolution of $c_{11}(t)$ is described by the following ODE:

$$\frac{dc_{11}}{dt} = 2c_{11} - 2c_{11}^2 \quad (67)$$

The closed-form solution to (67) can be shown to be:

$$c_{11}(t) = (1 - e^{-2t}\{1 - c_{11}^{-1}(0)\})^{-1}. \quad (68)$$

Hence, $\lim_{t \rightarrow \infty} c_{11}(t) = 1$. Next, assume that all $c_{kl}(t)$ for all $1 \leq l \leq k \leq j-1$ have converged to

$$\lim_{t \rightarrow \infty} c_{kl}(t) = \begin{cases} 1 & \text{if } l = k \\ 0 & \text{if } l < k \end{cases}. \quad (69)$$

Then, from (66), it can be shown that the evolution of $c_{ij}(t)$ for $1 \leq j < i$ is described by

$$\frac{dc_{ij}}{dt} = -(1 + c_{ii})c_{ij}. \quad (70)$$

Since $c_{ii}(t) > 0$ by construction, (70) guarantees that $\lim_{t \rightarrow \infty} c_{ij}(t) = 0$ for $1 \leq j < i$. Assuming that this convergence has occurred, the evolution of $c_{ii}(t)$ is described by the ODE

$$\frac{dc_{ii}}{dt} = 2c_{ii} - 2c_{ii}^2, \quad (71)$$

which has the solution

$$c_{ii}(t) = (1 - e^{-2t}\{1 - c_{ii}^{-1}(0)\})^{-1} \quad (72)$$

Hence, $\lim_{t \rightarrow \infty} c_{ii}(t) = 1$. Setting $i = m$ proves the theorem.

We now show that the ODE in (62) also describes the inverse LU algorithm's behavior through the relation $\mathbf{P} = \mathbf{GM}$. The associated ODEs for the updates of $\mathbf{M}(n)$ and $\mathbf{G}(n)$ in (14) and (15), respectively, are

$$\frac{d\mathbf{M}}{dt} = -2\underline{\text{tril}}[\mathbf{M}\mathbf{R}_{\mathbf{xx}}\mathbf{M}^T\mathbf{G}^T]\mathbf{M} \quad (73)$$

$$\frac{d\mathbf{G}}{dt} = \mathbf{G} - \text{diag}[\mathbf{G}\mathbf{M}\mathbf{R}_{\mathbf{xx}}\mathbf{M}^T\mathbf{G}^T]\mathbf{G}. \quad (74)$$

Since $\mathbf{P} = \mathbf{GM}$, the equivalent ODE for \mathbf{P} can be described in terms of the ODEs in (73) and (74) using

$$\frac{d\mathbf{P}}{dt} = \frac{d\mathbf{G}}{dt}\mathbf{M} + \mathbf{G}\frac{d\mathbf{M}}{dt}. \quad (75)$$

Substituting (73) and (74) into the right-hand sides of (75) and simplifying results in the ODE for \mathbf{P} in (62). Hence, the inverse LU algorithms' behavior is asymptotically identical to that of the inverse Cholesky algorithm as $\mu \rightarrow 0$.

7. APPENDIX B

This appendix provides proofs of Theorem 1 and Corollary 1.1 for the EASI algorithm. We begin with a proof of the first corollary. The evolution of the matrix product $\mathbf{W}^T\mathbf{W}$ is given by

$$\frac{d[\mathbf{W}^T\mathbf{W}]}{dt} = \frac{d\mathbf{W}^T}{dt}\mathbf{W} + \mathbf{W}^T\frac{d\mathbf{W}}{dt} \quad (76)$$

$$= \mathbf{W}^T\mathbf{W}\overline{\mathbf{G}}^T\mathbf{W} - \mathbf{W}^T\overline{\mathbf{G}}\mathbf{W}^T\mathbf{W} + \mathbf{W}^T\overline{\mathbf{G}}\mathbf{W}^T\mathbf{W} - \mathbf{W}^T\mathbf{W}\overline{\mathbf{G}}^T\mathbf{W} \quad (77)$$

$$= \mathbf{0} \quad (78)$$

where we have used (34) on the right-hand side of (76) to obtain (77). Since $d[\mathbf{W}^T\mathbf{W}]/dt = \mathbf{0}$, the value of $\mathbf{W}^T(t)\mathbf{W}(t)$ does not change with time. Therefore, if $\mathbf{W}(0)$ is orthonormal, $\mathbf{W}^T(t)\mathbf{W}(t) = \mathbf{I}$ for all $t \geq 0$.

Now, consider the evolutionary behavior of $\mathbf{B}(t)$. Using (32), it can be shown that

$$\frac{d\mathbf{B}}{dt} = \mathbf{W}\frac{d\mathbf{P}}{dt} + \frac{d\mathbf{W}}{dt}\mathbf{P} \quad (79)$$

$$= \mathbf{W}\mathbf{P} - \mathbf{W}\mathbf{P}\mathbf{R}_{\mathbf{xx}}\mathbf{P}^T\mathbf{P} + \mathbf{G}\mathbf{P}^T\mathbf{W}^T\mathbf{W}\mathbf{P} - \mathbf{W}\mathbf{P}\mathbf{G}^T\mathbf{W}\mathbf{P}. \quad (80)$$

where we used the relationships in (33) and (34) to obtain (80) from (79). Since $\mathbf{W}(t)$ is orthogonal for all $t \geq 0$, we can insert the term $\mathbf{W}^T\mathbf{W}$ into the last term of (80) to get

$$\frac{d\mathbf{B}}{dt} = [\mathbf{I} - \mathbf{W}\mathbf{P}\mathbf{R}_{\mathbf{xx}}\mathbf{P}^T\mathbf{W}^T]\mathbf{W}\mathbf{P} + [\mathbf{G}\mathbf{P}^T\mathbf{W}^T - \mathbf{W}\mathbf{P}\mathbf{G}^T]\mathbf{W}\mathbf{P}. \quad (81)$$

Substituting $\mathbf{W}\mathbf{P} = \mathbf{B}$ for all terms on the right-hand side of (81) gives the update in (30).

REFERENCES

- [1] L.L. Thurstone, "Multiple factor analysis," *Psychol. Rev.*, vol. 38, pp. 406-427, 1931.
- [2] C. Jutten and J. Héroult, "Blind separation of sources: An adaptive algorithm based on a neurnimetic architecture," *Signal Processing*, vol. 24, pp. 1-10, 1991.
- [3] A. Dinc and Y. Bar-Ness, "Bootstrap: A fast blind adaptive signal separator," *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing*, San Francisco, CA, vol. 2, pp. 325-328, Mar. 1992.
- [4] Y. Sato, "Two extensional applications of the zero-forcing equalization method," *IEEE Trans. Comm.*, vol. COM-23, pp. 684-687, June 1975.
- [5] E.A. Robinson and S. Treitel, *Geophysical Signal Analysis* (Englewood Cliffs, NJ: Prentice-Hall, 1980).
- [6] D.N. Godard, "Self-recovering equalization and carrier tracking in two-dimensional data communication systems," *IEEE Trans. Comm.*, vol. COM-28, pp. 1867-1875, Nov. 1980.
- [7] J.R. Treichler and B.G. Agee, "A new approach to multipath correction of constant modulus signals," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-31, pp. 349-372, Apr. 1983.
- [8] A. Benveniste and M. Goursat, "Blind equalizers," *IEEE Trans. Comm.*, vol. ACOM-32, pp. 871-883, Aug. 1984.
- [9] J.-F. Cardoso and A. Souloumias, "Blind beamforming for non-Gaussian signals," *Proc. IEE*, pt. F, vol. 140, pp. 362-370, Dec. 1993.
- [10] J. Karhunen and J. Joutsensalo, "Representation and separation of signals using nonlinear PCA type learning," *Neural Networks*, vol. 7, pp. 113-127, 1993.
- [11] A. Cichocki, R. Unbehauen, and E. Rummert, "Robust learning algorithm for blind separation of signals," *Electron. Lett.*, vol. 30, pp. 1386-1387, Aug. 1994.
- [12] P. Comon, "Independent component analysis: A new concept?" *Signal Processing*, vol. 36, pp. 287-314, Apr. 1994.
- [13] A.J. Bell and T.J. Sejnowski, "An information maximization approach to blind separation and blind deconvolution," *Neural Computation*, vol. 7, pp. 1129-1159, Nov. 1995.
- [14] S. Amari, A. Cichocki, and H.H. Yang, "A new learning algorithm for blind signal separation," *Adv. Neural Inform. Proc. Sys. 8* (Cambridge, MA: MIT Press, 1996), pp. 757-763.
- [15] J.-F. Cardoso, "Blind signal separation: Statistical principles," *Proc. IEEE*, vol. 86, pp. 2009-2025, Oct. 1998.
- [16] S. Haykin, ed., *Unsupervised Adaptive Filtering, Vol. 1: Blind Source Separation* (New York: Wiley, 2000).
- [17] A. Hyvärinen, J. Karhunen, and E. Oja, *Independent Component Analysis* (New York: Wiley, 2001).
- [18] A. Cichocki and S. Amari, *Adaptive Blind Signal Processing: Learning Algorithms and Applications* (New York: Wiley, 2002).
- [19] N. Delfosse and P. Loubaton, "Adaptive blind separation of independent sources: a deflation approach," *Signal Processing*, vol. 45, no. 1, pp. 59-83, July 1995.
- [20] A. Hyvärinen and E. Oja, "A fast fixed-point algorithm for independent component analysis," *Neural Computation*, vol. 9, no. 7, pp. 1483-1492, Oct. 1997.
- [21] N. Ahmed and D. H. Youn, "On a realization and related algorithm for adaptive prediction," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 28, pp. 493-497, Oct. 1980.
- [22] F.M. Silva and L.B. Almeida, "A distributed decorrelation algorithm," in *Neural Networks: Advances and Applications*, ed. E. Gelenbe, (Amsterdam: Elsevier Science, 1991), pp. 145-163.
- [23] K.I. Diamantaras and S.-Y. Kung, *Principal Component Neural Networks: Theory and Applications* (New York: Wiley, 1996).
- [24] J.J. Atick and A.N. Redlich, "Convergent algorithm for sensory receptive field development," *Neural Computation*, vol. 5, no. 1, pp. 45-60, 1993.
- [25] J.-F. Cardoso and B. Laheld, "Equivariant adaptive source separation," *IEEE Trans. Signal Processing*, vol. 44, pp. 3017-3030, Dec. 1996.
- [26] S.C. Douglas and A. Cichocki, "Neural networks for blind decorrelation of signals," *IEEE Trans. Signal Processing*, vol. 45, pp. 2829-2842, Nov. 1997.
- [27] S.C. Douglas, "Numerically-robust $\mathcal{O}(N^2)$ RLS algorithms using least-squares prewhitening," *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing*, Istanbul, Turkey, vol. I, pp. 412-415, June 2000.
- [28] S. Haykin, *Adaptive Filter Theory*, 4th ed. (Upper Saddle River, NJ: Prentice-Hall, 2002).
- [29] A.A. Rontogiannis and S. Theodoridis, "Inverse factorization adaptive least-squares algorithms," *Signal Processing*, vol. 52, no. 1, pp. 35-47, July 1996.
- [30] S.C. Douglas and R. Losada, "Adaptive filters in MATLAB: From novice To expert," *Proc. IEEE Signal Processing Education Workshop*, Pine Mountain, GA, Paper 4.9, Oct. 2002.
- [31] W. M. Gentleman and H. T. Kung, "Matrix triangularization by systolic arrays," *Proc. SPIE Int. Soc. Opt. Eng.*, vol. 298, pp. 19-26, 1982.
- [32] T.-P. Chen and Q. Lin, "Dynamic behavior of the whitening process," *IEEE Signal Processing Lett.*, vol. 5, pp. 25-26, Jan. 1998.
- [33] U. Helmke and J.B. Moore, *Optimization and Dynamical Systems* (New York: Springer-Verlag, 1994).
- [34] A. Edelman, T. Arias, and S.T. Smith, "The geometry of algorithms with orthogonality constraints," *SIAM J. Matrix Anal. Appl.*, vol. 20, pp. 303-353, Apr. 1998.
- [35] H.G. Grassman, *Die Ausdehnungslehre*, (Berlin: Enslin, 1862).
- [36] E. Stiefel, "Richtungsfelder und fernparallelismus in n -dimensionalem mannigfaltigkeiten," *Commentarii Math. Helvetici*, vol. 8, pp. 305-353, 1935-1936.
- [37] E. Oja and J. Karhunen, "On stochastic approximation of the eigenvectors and eigenvalues of the expectation of a random matrix," *J. Math. Anal. Appl.*, vol. 106, pp. 69-84, 1985.
- [38] S.C. Douglas, "Self-stabilized gradient algorithms for blind source separation with orthogonality constraints," *IEEE Trans. Neural Networks*, vol. 11, pp. 1490-1497, Nov. 2000.
- [39] P.A. Thompson, "An adaptive spectral analysis technique for unbiased frequency estimation in the presence of white noise," *Proc. 13th Asilomar Conf. Circ., Syst., Comput.*, Pacific Grove, CA, pp. 529-533, Nov. 1979.
- [40] S.C. Douglas and M. Rupp, "On bias removal and unit norm constraints in equation-error adaptive IIR filters," *Proc. 30th Asilomar Conf. Sig., Syst., Comput.*, Pacific Grove, CA, vol. 2, pp. 1093-1097, Nov. 1996.
- [41] V.U. Reddy, B. Egardt, and T. Kailath, "Least squares type algorithm for adaptive implementation of Pisarenko's harmonic retrieval method," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 30, pp. 399-405, June 1982.
- [42] S.C. Douglas, S.-Y. Kung, and S. Amari, "A self-stabilized minor subspace rule," *IEEE Signal Processing Lett.*, vol. 5, no. 12, pp. 328-330, Dec. 1998.
- [43] K. Abed-Meraim, S. Attallah, A. Chkeif, and Y. Hua, "Orthogonal Oja algorithm," *IEEE Signal Processing Lett.*, vol. 7, pp. 116-119, May 2000.
- [44] P. Strobach, "Square-root QR inverse iteration for tracking the minor subspace," *IEEE Trans. Signal Processing*, vol. 48, pp. 2994-2999, Nov. 2000.
- [45] X. Sun and S.C. Douglas, "Phase estimation using adaptive allpass filters," *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing*, Istanbul, Turkey, vol. I, pp. 444-447, June 2000.
- [46] J.D. Norris and S.C. Douglas, "A gradient algorithm for phase-only adaptive FIR filtering," to be presented at *IEEE Int. Conf. Acoust., Speech, Signal Processing*, Hong Kong, April 2003.
- [47] C. Cherry, "Some experiments on the recognition of speech with one and two ears," *J. Acoust. Soc. Am.*, vol. 25, pp. 975-981, 1953.
- [48] R.H. Lambert, "Multichannel blind deconvolution: FIR matrix algebra and separation of multipath mixtures," Ph.D. dissertation, Univ. Southern California, Los Angeles, CA, May 1996.
- [49] R.H. Lambert and A.J. Bell, "Blind separation of multiple speakers in a multipath environment," *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing*, Munich, Germany, vol. 1, pp. 423-426, Apr. 1997.
- [50] T.-W. Lee, A. Ziehe, R. Orglmeister, and T. Sejnowski, "Combining time-delayed decorrelation and infomax: Towards solving the cocktail party problem," *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing*, Seattle, WA, vol. 2, pp. 1089-1092, May 1998.
- [51] S.C. Douglas and S. Haykin, "Relationships between blind deconvolution and blind source separation," in *Unsupervised Adaptive Filtering, Vol. II: Blind Deconvolution*, S. Haykin, ed. (New York: Wiley, 2000), pp. 113-145.
- [52] S.C. Douglas, S. Amari, and S.-Y. Kung, "Adaptive paraunitary filter banks for spatio-temporal principal and minor subspace analysis," *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing*, Phoenix, AZ, vol. 2, pp. 1089-1092, Mar. 1999.