EURASIP Journal on Applied Signal Processing 2002:5, 538–551

Deterministic Blind Subspace MIMO Equalization

Balaji Sampath
Electrical and Computer Engineering Department and Institute for Systems Research, University of Maryland, College Park, MD 20742, USA
Email: bsampath@eng.umd.edu

K. J. Ray Liu
Electrical and Computer Engineering Department and Institute for Systems Research, University of Maryland, College Park, MD 20742, USA
Email: kjrliu@eng.umd.edu

Ye (Geoffrey) Li
School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA 30332, USA
Email: liye@ece.gatech.edu

Received 25 October 2001 and in revised form 14 February 2002

A subspace based approach for the blind multiple signal separation and recovery for MIMO systems is proposed in this paper. Instead of using the statistics of the received signal, the proposed algorithm exploits the received signal structure and the finite alphabet property of the desired signals. The finite alphabet property is used to remove the unknown unitary matrix that is associated with most of the statistics-based MIMO system identification algorithms. The proposed algorithm also incorporates an error-correcting procedure; therefore, it has more accuracy than the existing algorithms. Computer simulation results demonstrate that the algorithm can detect the signals and estimate channel parameters accurately with very few symbols, even under high noise and bad channel conditions.

Keywords and phrases: blind signal processing, subspace algorithm, MIMO system, equalization.

1. INTRODUCTION

Multiple transmit and receive antennas can be used in wireless communications to form *multiple-input and multiple-output* (MIMO) systems to improve transmission capacity and performance. The first problem that we have to address before using MIMO communication systems is to identify and equalize MIMO systems, that is, to find system parameters, separate and recover signals. In this paper, we present blind subspace algorithm for MIMO system equalization. Since almost all MIMO systems can be modeled or approximated as FIR systems, we limit ourselves to linear FIR systems.

A number of algorithms have been proposed for blind identification and equalization of channels with only one input. Traditionally, most blind equalization algorithms for *single-input and single-output* (SISO) systems have been based on higher-order statistics [1, 2, 3, 4]. In the last few years, a number of second-order statistics based algorithms have been proposed in [5, 6, 7] to exploit the cyclostationarity of oversampled *continuous* SISO communication systems. Since an oversampled continuous SISO system is equivalent to a discrete *single-input and multiple-output* (SIMO) system, these algorithms can be regarded as addressing the SIMO system identification problem as well. Recently, a number of subspace-based algorithms have been proposed in [8, 9, 10, 11, 12] for blind system identification. In particular, we have proposed a new deterministic subspace-based algorithm in [11, 12] which addresses blind equalization of oversampled continuous SISO (or equivalently discrete SIMO) systems. Compared with other algorithms in the literature, this algorithm has much better performance. However, it requires that the number of outputs equal to the FIR channel length, which is very restrictive in reality. In Section 2, we generalize this algorithm, remove this restriction, and study the performance of the algorithm under different oversampling rates. The approach used to generalize the SIMO identification algorithms can be also used to generalize the MIMO identification algorithm developed in Section 3.

Compared with the number of algorithms that exist for the single-input systems, there are very few algorithms which address the MIMO channel equalization. This is because the

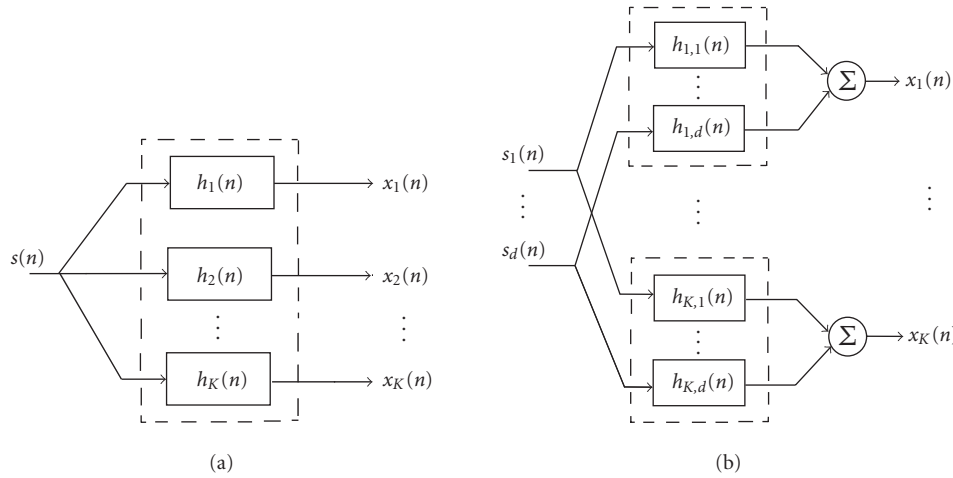


Figure 1: (a) SIMO System, (b) MIMO System.

MIMO problem is much more difficult to deal with. For example, it can be shown that by using only the structure and no other extra information (like quantized inputs or knowledge of the statistics), we can only identify the system up to a unitary matrix. Nevertheless, a few algorithms that equalize MIMO systems have been discovered [10, 13, 14]. In particular, [10, 14] are subspace-based methods which use the ILSP method described in [15] and are iterative. The statistics-based algorithm in [13], like other statistics-based algorithms, needs a very long observation interval to equalize the system and is therefore not suited to fast changing environments, such as wireless communications. It is becoming greater to find MIMO system blind equalization algorithms that use smaller observation intervals, work well at high noise conditions, and give more accurate estimates. In Section 3, we develop an algorithm for MIMO systems that requires a small observation interval. The effectiveness of the proposed algorithms is demonstrated through computer simulation in Section 4.

2. GENERALIZED SUBSPACE ALGORITHM FOR SIMO SYSTEMS

In [11, 12], we presented an algorithm for blind identification of oversampled continuous SISO systems under certain conditions. However, that algorithm requires that the oversampling rate should be exactly the same as the length of the FIR channel. Although in most cases this constraint will not create any problems. In some cases, too high oversampling rates will cause the impulse response matrix to become very ill-conditioned, which leads to poor channel estimates. After describing the mathematical model of SIMO systems, we will present a generalized subspace algorithm that does not require the constraint.

2.1. Problem statement

The mathematical model of 1-input and K-output systems can be illustrated as in Figure 1a. The sequence $s(n)$ is sent through K linear channels $h_i(n)$, $i = 1, \ldots, K$. In the noiseless case, the channel outputs $x_i(n)$, $1 \le i \le K$, can be expressed as

$$x_i(n) = \sum_{j=0}^{J-1} s(n-j)h_i(j), \tag{1}$$

where J is the maximum length of $h_i(\cdot)$'s. The problem we propose to tackle here is the blind identification and equalization of SIMO systems, that is, to find an algorithm to estimate the sequence $s(n)$ given only the outputs $x_i(n)$. The solution we presented in [11, 12] assumed that $K = J$. Here we will derive a more general algorithm without using the assumption.

For a given n, define for $1 \le i \le K$,

$$\begin{aligned} \mathbf{x}_i(p) &= \left[x_i(p) \;\; x_i(p+1) \;\; \cdots \;\; x_i(p+n-1)\right]^T, \\ \mathbf{s}(p) &= \left[s(p) \;\; s(p+1) \;\; \cdots \;\; s(p+n-1)\right]^T, \\ \mathbf{h}_i &= \left[h_i(0) \;\; h_i(1) \;\; \cdots \;\; h_i(J-1)\right]^T. \end{aligned} \tag{2}$$

With the above definitions, equation (1) can be expressed as,

$$\mathbf{x}_i(p) = \left[\mathbf{s}(p) \;\; \mathbf{s}(p-1) \;\; \cdots \;\; \mathbf{s}(p-J+1)\right] \mathbf{h}_i, \tag{3}$$

or

$$\begin{aligned} &\left[\mathbf{x}_1(p) \;\; \mathbf{x}_2(p) \;\; \cdots \;\; \mathbf{x}_K(p)\right] \\ &\quad = \left[\mathbf{s}(p) \;\; \mathbf{s}(p-1) \;\; \cdots \;\; \mathbf{s}(p-J+1)\right] \mathbf{H}, \end{aligned} \tag{4}$$

where $\mathbf{H}$ is a $J \times K$ matrix defined as

$$\mathbf{H} = \left[\mathbf{h}_1 \;\; \mathbf{h}_2 \;\; \cdots \;\; \mathbf{h}_K\right]. \tag{5}$$

When developing the subspace algorithms in [11, 12], we have assumed that the matrix $\mathbf{H}$ is square (i.e., $J = K$) and invertible.

Unfortunately, $\mathbf{H}$ need not be a square matrix (and hence will not be invertible). Therefore, we are going to introduce a generalized algorithm to deal with the situation.

2.2. Generalized subspace algorithm

The algorithm developed in [11, 12] requires that $\mathbf{H}$ be a square matrix. When $\mathbf{H}$ is not square, the generalized algorithm is to modify this matrix to get a square, invertible matrix, without substantially changing (4). Without loss of generality, we assume $K \leq J$.

Let $\mathbf{O}_{j\times k}$ be a $j \times k$ matrix of zeros. Define $\hat{\mathbf{H}}_1 = \mathbf{H}$ and $\hat{\mathbf{H}}_{j+1}$ recursively as,

$$\hat{\mathbf{H}}_{j+1} = \begin{bmatrix} \hat{\mathbf{H}}_j & \mathbf{O}_{j\times K} \\ \mathbf{O}_{1\times jK} & \mathbf{H} \end{bmatrix}. \tag{6}$$

From the definition, $\hat{\mathbf{H}}_j$ is a $(J + j - 1) \times jK$ matrix and it has more columns than rows when $j \geq (J - 1)/(K - 1)$. Let $j' = \lceil (J - 1)/(K - 1) \rceil$ and $\hat{\mathbf{H}}$ be a $(J + j' - 1) \times (J + j' - 1)$ square matrix consisting of the first $(J + j' - 1)$ columns of $\hat{\mathbf{H}}_{j'}$.

Now, define the following:

$$\begin{aligned} X(p) &= \begin{bmatrix} \mathbf{x}_1(p) & \mathbf{x}_2(p) & \cdots & \mathbf{x}_K(p) \end{bmatrix}, \\ \hat{X}(p - j' + 1) &= \begin{bmatrix} \mathbf{x}_1(p-j'+1) & \mathbf{x}_2(p-j'+1) & \cdots & \mathbf{x}_{K+(J-1)-(K-1)j'}(p-j'+1) \end{bmatrix}, \\ \mathbf{X}(p) &= \begin{bmatrix} X(p) & \cdots & X(p - j' + 2) & \hat{X}(p - j' + 1) \end{bmatrix}, \\ \mathbf{S}(p) &= \begin{bmatrix} \mathbf{s}(p) & \cdots & \mathbf{s}(p - J + 1) & \cdots & \mathbf{s}\big(p - J + 1 - (j' - 1)\big) \end{bmatrix}. \end{aligned} \tag{7}$$

With the above definitions, we have the following equation:

$$\mathbf{X}(p) = \mathbf{S}(p)\hat{\mathbf{H}}. \tag{8}$$

Now define the matrix $\mathbf{\Phi}$ as

$$\mathbf{\Phi} = \begin{bmatrix} \mathbf{X}(p+1) & -\mathbf{X}(p) & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{X}(p+1) & -\mathbf{X}(p) & \ddots & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{X}(p+1) & \ddots & \mathbf{0} & \mathbf{0} \\ \vdots & \ddots & \ddots & \ddots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{X}(p+1) & -\mathbf{X}(p) \end{bmatrix} \tag{9}$$

with the number of block-rows being $J + j' - 1$ and $\mathbf{0}$ representing a matrix of zeros.

We can show that $\mathbf{\Phi}$ can be factored as

$$\mathbf{\Phi} = \mathbf{\Psi}\tilde{\mathbf{H}}, \tag{10}$$

where

$$\tilde{\mathbf{H}} = \begin{bmatrix} \hat{\mathbf{H}} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \hat{\mathbf{H}} & \ddots & \mathbf{0} \\ \vdots & \ddots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \cdots & \hat{\mathbf{H}} \end{bmatrix},$$

$$\mathbf{\Psi} = \begin{bmatrix} \mathbf{S}(p+1) & -\mathbf{S}(p) & 0 & \cdots & 0 & 0 \\ 0 & \mathbf{S}(p+1) & -\mathbf{S}(p) & \ddots & 0 & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & \mathbf{S}(p+1) & -\mathbf{S}(p) \end{bmatrix}. \tag{11}$$

From [11, 12, Lemma 3.2], we know that the probability that $\mathbf{\Psi}$ has a one-dimensional null-space tends to 1 exponentially with increasing n. We also know that the one-dimensional null space is given by $\mathcal{N}(\mathbf{\Psi})$,

$$\mathcal{N}(\mathbf{\Psi}) = \left\{ c \begin{bmatrix} \mathbf{e}_1 & \mathbf{e}_2 & \cdots & \mathbf{e}_{J+j'-1} \end{bmatrix}^T : c \in \Re \right\}, \tag{12}$$

where $\mathbf{e}_i$ is a $1 \times (J + j' - 1)$ vector given by

$$\mathbf{e}_i = \begin{bmatrix} 0 \cdots 0 \underbrace{1}_{i\text{th}} 0 \cdots 0 \end{bmatrix}. \tag{13}$$

So for all practical purposes we can assume that $\mathbf{\Psi}$ has a one-dimensional null space. If we further assume that $\hat{\mathbf{H}}$ is invertible, it is easily seen that $\mathbf{\Phi}$ has a one-dimensional null-space and that the null space of $\mathbf{\Phi}$ is given by

$$\mathcal{N}(\mathbf{\Phi}) = \left\{ c \begin{bmatrix} \hat{\mathbf{H}}^{-1}\mathbf{e}_1^T \\ \hat{\mathbf{H}}^{-1}\mathbf{e}_2^T \\ \vdots \\ \hat{\mathbf{H}}^{-1}\mathbf{e}_{J+j'-1}^T \end{bmatrix} : c \in \Re \right\}. \tag{14}$$

Note that $\mathbf{\Phi}$ is a matrix whose elements are the actual channel output samples—so we can apply some of the standard algorithms to find its null space—let the null space be

$$\mathcal{N}(\mathbf{\Phi}) = \left\{ c \begin{bmatrix} \boldsymbol{\lambda}_1 & \boldsymbol{\lambda}_2 & \cdots & \boldsymbol{\lambda}_{J+j'-1} \end{bmatrix}^T : c \in \Re \right\}, \tag{15}$$

where $\boldsymbol{\lambda}_i$ is a $1 \times (J + j' - 1)$ vector. We therefore have,

$$\hat{\mathbf{H}}^{-1}\begin{bmatrix} \mathbf{e}_1^T & \mathbf{e}_2^T & \cdots & \mathbf{e}_{J+j'-1}^T \end{bmatrix} = \hat{\mathbf{H}}^{-1} = c\begin{bmatrix} \boldsymbol{\lambda}_1^T & \boldsymbol{\lambda}_2^T & \cdots & \boldsymbol{\lambda}_{J+j'-1}^T \end{bmatrix}. \tag{16}$$

From the null space of $\mathbf{\Phi}$ we can get $\hat{\mathbf{H}}$ and hence $\mathbf{H}$, the channel response matrix, up to a multiplication factor. Thus, in the noiseless case we have identified the channel exactly.

The same algorithm can be used for noisy case. Instead of finding a null vector of $\mathbf{\Phi}$, we find the smallest singular vector of $\mathbf{\Phi}$. To obtain a better estimate of the channel, we can also apply the *error-correcting least-squares* (ECLS)

algorithm discussed in [11, 12]. Here we have a choice of several sampling rates (or equivalently several channels) unlike [11, 12] where the sampling rate was fixed at J—by choosing an appropriate sampling rate, it may be possible to overcome the problem of ill-conditionedness that is always faced in blind equalization algorithms. In [11, 12], the ill-conditionedness problem was overcome by choosing the effective channel length to be smaller than J. Here the ill-conditionedness can be overcome by reducing the sampling rate itself. In Section 4, we present some simulation results comparing the performance of this algorithm at different sampling rates. From the simulations we can see that at least for very ill-conditioned matrices, the performance becomes better as the rate of over-sampling decreases. The reason for this counter-intuitive result[1] is that, in very ill-conditioned matrices, increasing the sampling rate does not really increase the information content in the samples.

3. THE SUBSPACE ALGORITHM FOR THE MIMO SYSTEMS

In Section 2, we looked at the SIMO channel equalization problem. In this section, we tackle the MIMO channel case. The mathematical model of d-input/K-output MIMO systems can be illustrated as in Figure 1b. The d sequences $s_1(n), \ldots, s_d(n)$ are sent through linear channels $h_{ij}(n)$ for $i = 1, \ldots, K$ and $j = 1, \ldots, d$ and hence, the channel outputs $x_k(n)$, $1 \le k \le K$, can be expressed as

$$x_k(n) = \sum_{i=1}^{d} \sum_{m=0}^{J} h_{k,i}(m) \cdot s_i(n-m), \tag{17}$$

where J is the maximum length of $h_{i,j}(\cdot)$. The solution we present below assumes that $K = Jd$. By using the approach that was presented in Section 2 for the SIMO case, this algorithm can also be extended to $K \neq Jd$ case.

3.1. The MIMO subspace approach

First we define some matrices and derive some basic relations between them, and then prove Theorem 1. Based on these relations and Theorem 1, we then derive the MIMO subspace algorithm.

For a given n, for $1 \le i \le d$ and $1 \le j \le Jd$, define

$$\begin{aligned}
\mathbf{s}_i(p) &= \left[s_i(p) \;\; s_i(p+1) \;\; \cdots \;\; s_i(p+n-1)\right]^T, \\
\mathbf{S}_i(p) &= \left[\mathbf{s}_i(p) \;\; \mathbf{s}_i(p-1) \;\; \cdots \;\; \mathbf{s}_i(p-J+1)\right]^T, \\
\mathbf{S}(p) &= \left[\mathbf{S}_1^T(p) \;\; \mathbf{S}_2^T(p) \;\; \cdots \;\; \mathbf{S}_d^T(p)\right]^T, \\
\mathbf{x}_j(p) &= \left[x_j(p) \;\; x_j(p+1) \;\; \cdots \;\; x_j(p+n-1)\right]^T, \\
\mathbf{X}(p) &= \left[\mathbf{x}_1(p) \;\; \mathbf{x}_2(p) \;\; \cdots \;\; \mathbf{x}_{dJ}(p)\right], \\
\mathbf{h}_{i,j} &= \left[h_{i,j}(0) \;\; h_{i,j}(1) \;\; \cdots \;\; h_{i,j}(J-1)\right]^T,
\end{aligned}$$

$$\mathbf{H} = \begin{bmatrix} \mathbf{h}_{1,1} & \mathbf{h}_{2,1} & \cdots & \mathbf{h}_{dJ,1} \\ \mathbf{h}_{1,2} & \mathbf{h}_{2,2} & \cdots & \mathbf{h}_{dJ,2} \\ \vdots & \vdots & \vdots & \vdots \\ \mathbf{h}_{1,d} & \mathbf{h}_{2,d} & \cdots & \mathbf{h}_{dJ,d} \end{bmatrix}. \tag{18}$$

We then have the following equation:

$$\mathbf{X}(p) = \mathbf{S}(p)\mathbf{H}. \tag{19}$$

Now construct the matrix $\mathbf{\Phi}$ as

$$\mathbf{\Phi} = \begin{bmatrix} \mathbf{X}(p+1) & -\mathbf{X}(p) & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{X}(p+1) & -\mathbf{X}(p) & \ddots & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{X}(p+1) & \ddots & \mathbf{0} & \mathbf{0} \\ \vdots & \ddots & \ddots & \ddots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{X}(p+1) & -\mathbf{X}(p) \end{bmatrix} \tag{20}$$

with the number of block-rows being J, 0 representing a matrix of zeros. Using (18) and (19), it is easy to see that the following identity holds:

$$\mathbf{\Phi} = \mathbf{\Psi}\tilde{\mathbf{H}}, \tag{21}$$

where

$$\mathbf{\Psi} = \begin{bmatrix} \mathbf{S}(p+1) & -\mathbf{S}(p) & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{S}(p+1) & -\mathbf{S}(p) & \ddots & \mathbf{0} & \mathbf{0} \\ \vdots & \ddots & \ddots & \ddots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{S}(p+1) & -\mathbf{S}(p) \end{bmatrix}, \tag{22}$$

and $\tilde{\mathbf{H}} = \mathrm{diag}(\mathbf{H}, \mathbf{H}, \ldots, \mathbf{H})$ and the number of block matrices $\mathbf{H}$ is J.

Below we prove that $\mathbf{\Psi}$ has a d-dimensional null space.

Theorem 1. *If the vectors $\mathbf{s}_i(j)$ for $1 \le i \le d$ and $(p - J + 1) \le j \le (p+1)$ are independent, then $\mathbf{\Psi}$ has the d-dimensional null space given by*

$$\mathcal{N}(\mathbf{\Psi}) = \left\{ c_1 \begin{bmatrix} \mathbf{e}_1 \\ \mathbf{e}_2 \\ \vdots \\ \mathbf{e}_J \end{bmatrix} + c_2 \begin{bmatrix} \mathbf{e}_{J+1} \\ \mathbf{e}_{J+2} \\ \vdots \\ \mathbf{e}_{2J} \end{bmatrix} + \cdots + c_d \begin{bmatrix} \mathbf{e}_{(d-1)J+1} \\ \mathbf{e}_{(d-1)J+2} \\ \vdots \\ e_{dJ} \end{bmatrix} : c_k \in \Re \right\}, \tag{23}$$

where $\mathbf{e}_i = [0 \cdots 0 \underbrace{1}_{ith} 0 \cdots 0]$.

The proof of the theorem will use the following lemma, which is proved in the appendix.

[1] Intuitively, higher sampling rate implies more information, which in general should lead to better performance.

Lemma 1. *If the $J+1$ vectors $\mathcal{B}_0, \mathcal{B}_1, \ldots, \mathcal{B}_J$ are linearly independent, the $n \times n$ matrix, $\mathbf{\Gamma}$, with zero elements everywhere except the following elements:*

$$\begin{aligned} \mathbf{\Gamma}_{ii} &= \mathcal{B}_1 \mathcal{B}_2 \cdots \mathcal{B}_J, \\ \mathbf{\Gamma}_{i,i+1} &= \mathcal{B}_0 \mathcal{B}_1 \cdots \mathcal{B}_{J-1} \end{aligned} \tag{24}$$

has a one-dimensional null space and

$$\mathcal{N}(\mathbf{\Gamma}) = \left\{ c \begin{bmatrix} \mathbf{e}_1 & \mathbf{e}_2 & \mathbf{e}_3 & \cdots & \mathbf{e}_J \end{bmatrix}^T : c \in \Re \right\}. \tag{25}$$

With the above lemma, we can now prove the theorem.

Proof of Theorem 1. Let $\mathbf{O}_{n\times k}$ be an $n \times k$ matrix of zeros, $\hat{\mathbf{S}}_i(p) = \begin{bmatrix} \mathbf{O}_{n\times(i-1)J} & \mathbf{S}_i(p+1) & \mathbf{O}_{n\times(d-i)J} \end{bmatrix}$, and

$$\mathbf{\Psi}_i = \begin{bmatrix} \hat{\mathbf{S}}_i(p+1) & -\hat{\mathbf{S}}_i(p) & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \hat{\mathbf{S}}_i(p+1) & -\hat{\mathbf{S}}_i(p) & \ddots & \mathbf{0} & \mathbf{0} \\ \vdots & \ddots & \ddots & \ddots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \hat{\mathbf{S}}_i(p+1) & -\hat{\mathbf{S}}_i(p) \end{bmatrix}. \tag{26}$$

Now, $\mathbf{\Psi}$ can be written as

$$\mathbf{\Psi} = \sum_{i=1}^{d} \mathbf{\Psi}_i. \tag{27}$$

Let

$$\boldsymbol{\lambda} = \begin{bmatrix} \boldsymbol{\lambda}_{1,1}^T & \cdots & \boldsymbol{\lambda}_{d,1}^T & \boldsymbol{\lambda}_{1,2}^T & \cdots & \boldsymbol{\lambda}_{d,2}^T & \cdots & \boldsymbol{\lambda}_{1,J}^T & \cdots & \boldsymbol{\lambda}_{d,J}^T \end{bmatrix}^T \tag{28}$$

be a $dJ^2 \times 1$ vector in the null space of $\mathbf{\Psi}$ (where $\boldsymbol{\lambda}_{i,j}$ is a $J \times 1$ vector). Now for $1 \le i \le d$, define β_i as the vector obtained by replacing all elements in $\boldsymbol{\lambda}$, except $\boldsymbol{\lambda}_{i,1}, \boldsymbol{\lambda}_{i,2}, \ldots, \boldsymbol{\lambda}_{i,J}$, by zero. Therefore, β_i is given by

$$\beta_i = \begin{bmatrix} \boldsymbol{\theta}_{i,1} & \boldsymbol{\theta}_{i,2} & \boldsymbol{\theta}_{i,3} & \cdots & \boldsymbol{\theta}_{i,J} \end{bmatrix}^T, \tag{29}$$

where $\boldsymbol{\theta}_{i,k} = \begin{bmatrix} \mathbf{O}_{1\times(i-1)J} & \boldsymbol{\lambda}_{i,k}^T & \mathbf{O}_{1\times(d-i)J} \end{bmatrix}$.

It can be seen that the following relation holds:

$$\mathbf{\Psi}_i \beta_i = \mathbf{0} \quad \text{for } 1 \le i \le d. \tag{30}$$

Using Lemma 1 it can be seen that (30) implies that

$$\beta_i = c_i \begin{bmatrix} \mathbf{e}_{(i-1)J+1}^T & \mathbf{e}_{(i-1)J+2}^T & \mathbf{e}_{(i-1)J+3}^T & \cdots & \mathbf{e}_{iJ}^T \end{bmatrix}^T \tag{31}$$

for some $c_i \in \Re$. Since $\boldsymbol{\lambda} = \sum_{i=1}^{d} \beta_i$, the theorem follows. □

Assuming further that $\mathbf{H}$ is invertible, we can conclude that $\mathbf{\Phi}$ also has a d-dimensional null space if the vectors $\mathbf{s}_i(j)$, for $1 \le i \le d$, $(p - J + 1) \le j \le (p + 1)$, are independent. But it is easy to see that as n increases, the probability that these vectors are not independent goes to zero exponentially [11, 12]. Therefore, we conclude that as n increases, the probability that $\mathbf{\Phi}$ has a d-dimensional null space increases exponentially to 1.

Also from the expression for the null space of $\mathbf{\Psi}$ in Theorem 1, it follows that the null space of $\mathbf{\Phi}$ can be written as

$$\mathcal{N}(\mathbf{\Phi}) = \sum_{i=1}^{d} c_i \mathbf{b}_i, \tag{32}$$

where the basis vectors $\mathbf{b}_i$ are

$$\mathbf{b}_i = \begin{bmatrix} \mathbf{e}_{(i-1)J+1}^T \mathbf{\Lambda}^T & \mathbf{e}_{(i-1)J+2}^T \mathbf{\Lambda}^T & \mathbf{e}_{(i-1)J+3}^T \mathbf{\Lambda}^T & \cdots & \mathbf{e}_{iJ}^T \mathbf{\Lambda}^T \end{bmatrix}^T, \tag{33}$$

where $\mathbf{\Lambda} = \mathbf{H}^{-1}$.

Note that the d basis vectors $\{\mathbf{b}_i\}$ have all the required information about $\mathbf{H}$ as well as about the transmitted sequences, for

$$\mathbf{\Lambda} = \begin{bmatrix} \mathbf{\Lambda}\mathbf{e}_1 & \mathbf{\Lambda}\mathbf{e}_2 & \mathbf{\Lambda}\mathbf{e}_3 & \cdots & \mathbf{\Lambda}\mathbf{e}_{dJ} \end{bmatrix}, \tag{34}$$

$$\mathbf{S}(p) = \mathbf{X}(p)\mathbf{\Lambda}. \tag{35}$$

Another important observation is that just the knowledge of $\mathbf{b}_i$ is enough to determine the ith transmitted sequence. The reason is that the J columns (from columns $(i-1)J+1$ to iJ) of $\mathbf{\Lambda}$ are from $\mathbf{b}_i$ and the information about the ith transmitted sequence can be obtained by post-multiplying $\mathbf{X}(p)$ by these J columns of $\mathbf{\Lambda}$.

3.2. The problem caused by the d-dimensionality of the null space

From the above relations, it is clear that if we have the basis vectors, then we have solved the MIMO problem. Unfortunately, the basis vectors are unknown. In the SIMO case, this problem does not arise since the null space is one-dimensional. Here the null-space is d-dimensional and standard algorithms (like the SVD algorithm) can give us the null space, but not the basis vectors that we need. Just arbitrary d basis vectors for the null-space will not be enough—the basis vectors we need are special as given by (33). If we just choose a basis for the null space, what we end up with is a linear combination of the basis vectors, that is, vectors of the form $\sum_{i=1}^{d} k_i \mathbf{b}_i$, where $k_i \in \Re$. The problem that we therefore have to address can be stated as follows: *given the null space (and some basis for it), find the d basis vectors* $\{\mathbf{b}_i\}$.

To solve this problem, we now make use of the discrete nature of the transmitted sequence. In what follows, for the sake of simplicity in presentation, we focus only on the $d = 2$ case. The points discussed below can be easily generalized to arbitrary d.

Our two basis vectors $\mathbf{b}_1$ and $\mathbf{b}_2$ lie in the null space. So if we have any other two independent vectors which belong to the null space, then our basis vectors are just linear combinations of those two vectors. We write this mathematically. Let γ_1 and γ_2 be any two vectors in the null space obtained from $\mathbf{\Phi}$, then

$$\mathbf{b}_1 = x\gamma_1 + y\gamma_2, \qquad \mathbf{b}_2 = z\gamma_1 + w\gamma_2, \tag{36}$$

for some $x, y, z, w \in \Re$. The problem now reduces to finding x, y, z, and w.

Table 1: Procedure for finding x and y.

Step 1. Choose two independent vectors, γ_1 and γ_2, in the null space.
Step 2. Choose some x and y.
Step 3. Substitute these values of x, y, γ_1, and γ_2 in (36). This gives us an estimate of $\mathbf{b}_1$.
Step 4. Using this estimated $\mathbf{b}_1$ and (35) obtain an estimate for the transmitted sequence.

Table 2: MIMO-subspace algorithm (without error correction).

Step 1. Construct the matrix $\mathbf{\Phi}$.
Step 2. Find any two linearly independent vectors, γ_1 and γ_2, in the null space of $\mathbf{\Phi}$.
Step 3. Using the procedure described above and using the *proper transmitted sequence* criterion for correctness, search for the correct x and y over all possible values they can take.

We had previously made the observation that just the knowledge of $\mathbf{b}_1$ is enough to determine the first sequence. This now boils down to the statement that just the knowledge of x and y, the weighting factors of γ_1 and γ_2, is enough to determine the first sequence. We will make use of this observation below to solve for x and y and hence the transmitted sequence. Table 1 describes the procedure to find x and y.

The big question is of course: what if the estimate of x and y is wrong? What criterion should be used to find out whether our estimate for x and y is correct? One reasonable criterion is the following: *whenever the transmitted sequence obtained from x and y is a proper transmitted sequence, then we can assume that the estimate of x and y is correct*. Note that we do not know the exact transmitted sequence but we do know the symbol constellation of the transmitted sequence. So we can define a *proper transmitted sequence* to be any sequence with elements from the signal constellation.

Using the idea presented above to check whether any assumed x and y is correct, we have Table 2 for estimating the transmitted sequence (and the channel responses).

Here are some comments on Table 2.

(1) To find x and y in Step 3 in the algorithm, two-dimensional search over real numbers is required. The *proper transmitted sequence* criterion defines the correct estimate by the fact that at the correct estimates of x and y, all the elements of the estimated transmitted sequence will lie in the signal constellation. In a noisy situation, instead of the correct estimates of x and y, we will have the optimum estimates of x and y and the condition that the estimated transmitted sequence should lie in the signal constellation will be replaced by the following: *denote the estimated transmitted sequence to be* $\mathbf{v}$. *Find the constellation-sequence* $\mathbf{u}$ *closest*[2] *to* $\mathbf{v}$, *that is, to find x and y that minimize* $\|\mathbf{u} - \mathbf{v}\|$.

(2) If signal constellation is the same for both the sequences, there will be two sets of x and y at which the estimated transmitted sequence will lie in (or in the noisy case be very close to a sequence with elements in) the signal constellation. These represent the two transmitted sequences and with one search we can find both the transmitted sequences.

(3) As we can see in Section 3.3, the two-dimensional search can actually be reduced to a one-dimensional search.

[2]We can choose any sensible criterion for closeness—for most practical cases, we can either use the nearest neighbor rule or more simply just perform an element by element quantization. In our simulations we have adopted the later approach of element by element quantization.

3.3. Complexity reduction

To reduce the computational complexity, it would be better if we could manage with just a one-dimensional search instead of a two-dimensional search over reals. It is in fact possible to reduce the two-dimensional search to a one-dimensional search. Furthermore, it does not involve any loss in accuracy of estimate.

There are many ways in which this can be done. For example, we can just set $x = 1$ and search over all values of y from $-\infty$ to ∞. We can normalize each vector obtained and this gives all possible[3] normalized vectors in the null space. But we end up with problems if for example $\mathbf{b}_1 = \gamma_2$, for then y will have to be ∞ which is not practically reachable through any search algorithm. So from a practical point of view this approach and other similar approaches are not very effective. We found the following procedure most advantageous from a computational angle.

(1) Perform Gram-Schmidt orthonormalization procedure on the two vectors, γ_1 and γ_2 and obtain the two orthonormal vectors δ_1 and δ_2. Let τ_1 and τ_2 be the first $2J$ elements of δ_1 and δ_2, respectively.

(2) Minimize the following over $0 \le \theta \le 2\pi$,

$$\|\mathbf{a} - \text{quant}(\mathbf{a})\|, \tag{37}$$

where $\mathbf{a}$ is the estimate of the transmitted sequence given by

$$\mathbf{a} = \mathbf{X}(p+1)\big(\cos(\theta)\boldsymbol{\tau}_1 + \sin(\theta)\boldsymbol{\tau}_2\big) \tag{38}$$

and quant($\mathbf{x}$) is the element by element quantization of $\mathbf{x}$, where the quantization maps each element of $\mathbf{x}$ to its nearest point in the signal constellation. Figure 2 shows a sample plot of $\|\mathbf{a} - \text{quant}(\mathbf{a})\|$ versus θ (in the noisy case).

The actual estimates of the two transmitted sequences are the quantized versions of the estimated sequences obtained from the two values of θ at the two minima.

3.4. Error-correcting least squares MIMO algorithm

We have so far made substantial use of the structure inherent in the output samples. Although we have also used the fact that the transmitted sequence is quantized, we have not

[3]It is important not to lose track of the essence behind all these manipulations—our aim is to search over all the vectors in the null space—because we know that one (or more accurately one pair) of the vectors is the correct one. All these tricks we are employing are just ways of making this search systematic.

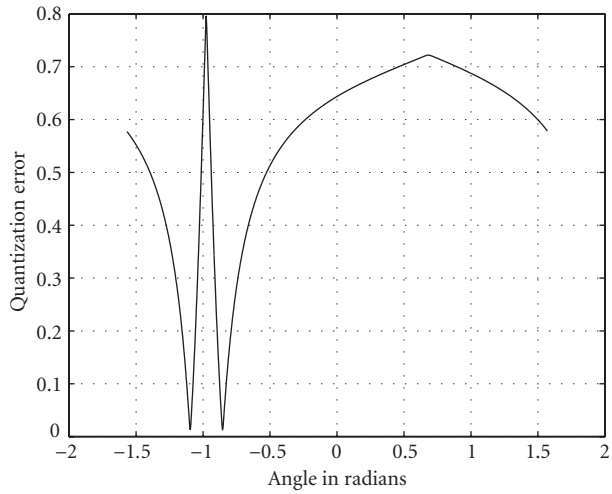


Figure 2: One sample plot of $\|\mathbf{a} - \text{quant}(\mathbf{a})\|$ versus $\boldsymbol{\theta}$.

exploited it fully. The ECLS algorithm discussed in [11, 12] does this very effectively and we can apply it here as well.

The crucial point in the ECLS algorithm is the observation that in almost all practical cases, if the initial estimate is close enough to the desired optimum, then we can reach the optimum by searching over a smaller area around initial estimate. The great reduction in the computational complexity comes from the fact that n small searches is just linear complexity whereas one search over the whole area[4] will have a complexity of n^p, where p is the dimension in which we are searching.

By applying this same idea here, we can reduce the search complexity greatly. The only problem is that we have two estimated sequences rather than one. So there are many ways in which we can do a smaller search. We can at each stage search over all possible one-element deviations for each transmitted sequence and then choose those one-element deviations for each sequence which corresponds to the overall optimum. The algorithm which does this is shown in Table 3.

But since we are only dealing with finite sequences, there is an even more simpler way of using error-correction. Concatenate the two sequences and apply the error correcting procedure on the combined 1D sequence. This is clearly only a subset of the previously described algorithm and so the performance of this will be degraded compared to the previous. But the importance of this approach lies in the fact that it reduces the complexity substantially. In the case that the two transmitted sequences are independent, the assumption that we can search over the two sequences independently (which is what we are doing here) instead of a joint search for the best sequence-pair, is quite reasonable. Because of these advantages, in our simulations we use this approach. Although the ECLS-MIMO algorithm improves the performance in all cases, the improvement is especially important when the matrix $\mathbf{H}$ is ill-conditioned because the channel and sequence estimates are not very accurate.

[4]The word area has been used as a general term for the region of search, not to denote the two-dimensionality of the search region.

Table 3: ECLS-MIMO algorithm.

Step 1. First let the search area be all the sequence-pairs which differ from the *current sequence-pair* in at most one position for each sequence in the pair.
Step 2. Find the *best* sequence-pair in this area. The best sequence-pair is the one which has the smallest least-squares error, that is, minimizes $\|\hat{\mathbf{S}}(p)\mathbf{H} - \mathbf{X}(p)\|$ over all possible $\mathbf{H}$ matrices, where $\hat{\mathbf{S}}(p)$ is the $\mathbf{S}(p)$-matrix that would be used in (21) if the actual transmitted sequences were the sequence-pairs in question.
Step 3. If this sequence is the same as the *current sequence-pair* then stop and decide that this is the *best* estimate.
Step 4. If this sequence is not the same as the *current sequence-pair* then replace the *current sequence-pair* by this sequence-pair and start all over again from Step 1.

4. SIMULATION RESULTS

In this section, we present some simulation results for the two algorithms presented in this paper—the generalized SIMO and MIMO-subspace algorithms—and compare it with other algorithms, under the same channel and noise conditions.

4.1. Simulations results for the generalized SIMO algorithm

First, we present the results for the generalized SIMO-subspace algorithm and compare it with the results in [8, 9, 10]. Here we do not present the results for [11, 12] since the algorithm in [11, 12] assumes that the "numerical" channel length is smaller than the actual channel length and uses the shorter channel length for its computation which leads to its better performance. This approach to overcome the ill-conditionedness of the channel is complimentary to the lower sampling rate approach we have proposed in this paper and both the approaches can be used simultaneously to even further improve the performance.

For these simulations, we used a channel with $J = 4$. Oversampling it at four times the baud rate, the channel matrix $\mathbf{H}$ which we used is given by

$$\mathbf{H} = \begin{bmatrix} 0.04142 & 0.0216 & -0.01959 & -0.06035 \\ -0.07025 & -0.0241 & 0.08427 & 0.2351 \\ 0.3874 & 0.4931 & 0.5167 & 0.4494 \\ 0.3132 & 0.152 & 0.01383 & -0.06754 \end{bmatrix}. \quad (39)$$

To quantify the ill-conditionedness or well-conditionedness of a matrix we will use RCOND($\mathbf{X}$). RCOND($\mathbf{X}$) is an estimate for the reciprocal of the condition of X in the 1-norm. If X is well conditioned, RCOND($\mathbf{X}$) is near 1.0. If X is badly conditioned, RCOND($\mathbf{X}$) is near 0.0. For the matrix $\mathbf{H}$ above, RCOND($\mathbf{H}$) = 0.0027. It is therefore clearly a

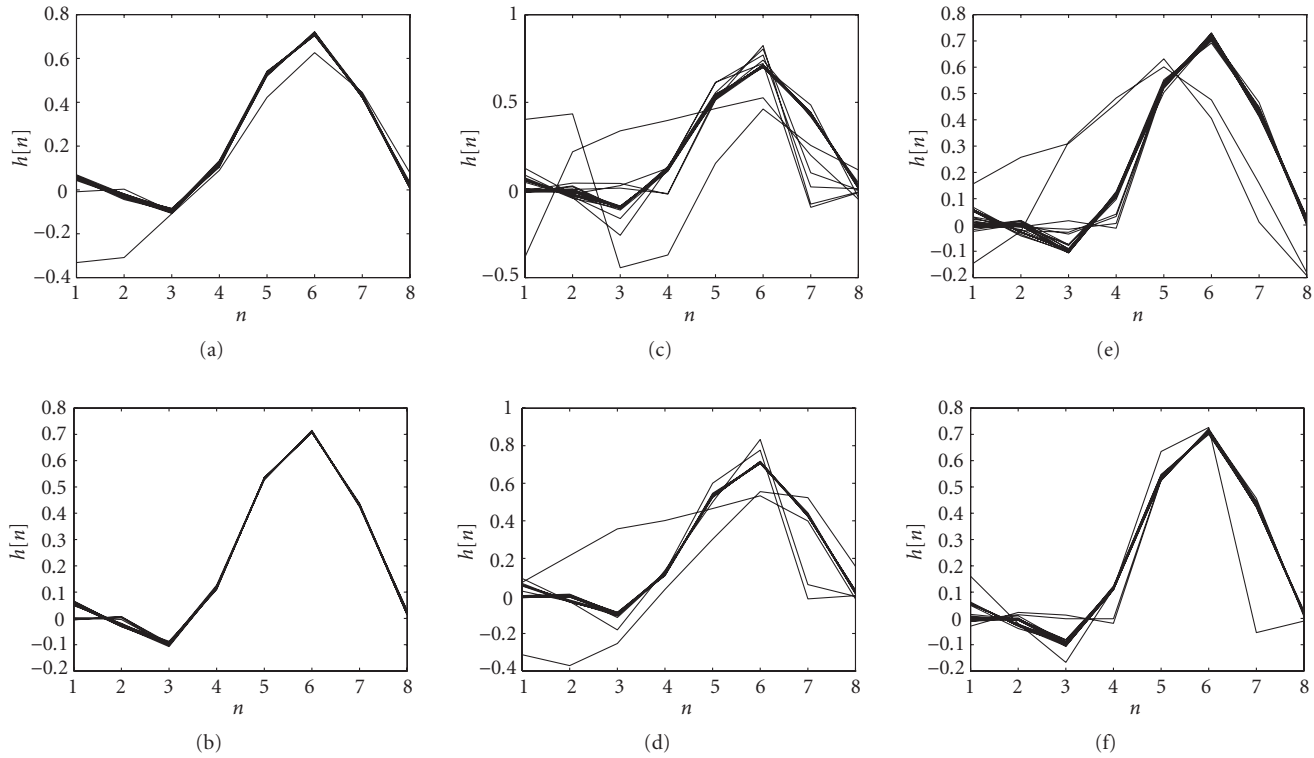


FIGURE 3: Generalized subspace algorithm, (a), (b) 2-oversampling, (c), (d) 3-oversampling, and (e), (f) 4-oversampling. SNR = 25 dB. (a), (c), and (e) 40 symbols, (b), (d), and (f) 100 symbols.

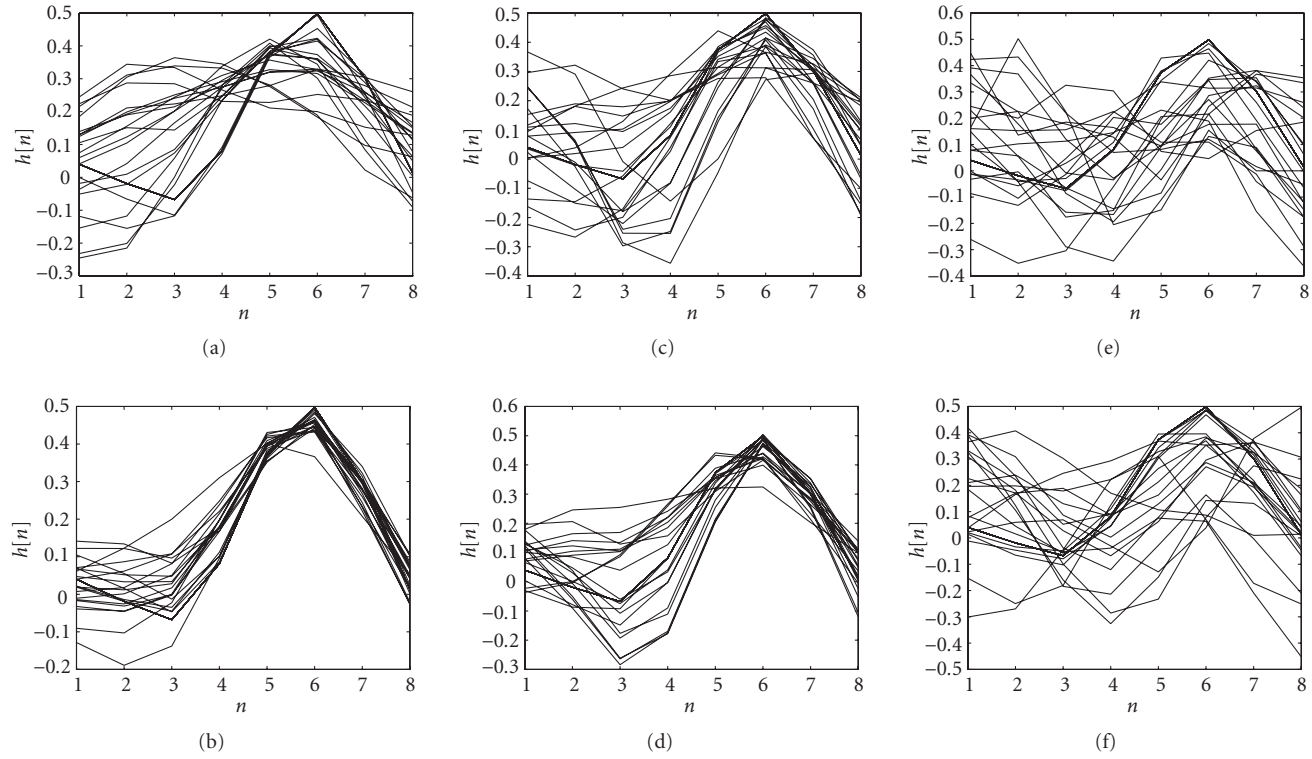


FIGURE 4: Algorithms (a), (b) in [8], (c), (d) in [9], and (e), (f) in [10]. SNR = 25 dB. (a), (c), and (e) 40 symbols, (b), (d), and (f) 100 symbols.

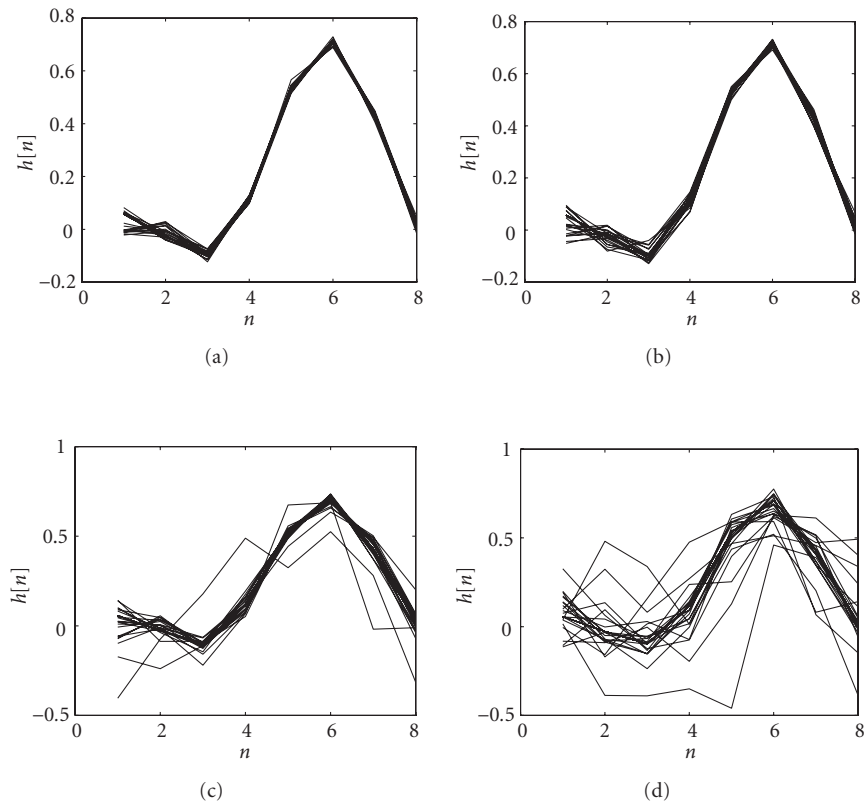


Figure 5: 20 estimates of the channel for (a) SNR = 20 dB, (b) SNR = 15 dB, (c) SNR = 10 dB, and (d) SNR = 5 dB for the oversampling twice-subspace algorithm. 40 symbols.

very ill-conditioned matrix. If we oversample the channel at only twice the baud rate (as opposed to the four times oversampling done above) and create a square matrix using the method presented in Section 2, the RCOND of the matrix thus obtained is 0.0123. This is still an ill-conditioned matrix but the condition number is five times better. By choosing the sampling instants appropriately we can even make the RCOND = 0.0186—but since we have no hold over the sampling instants, for our simulations we assume a sampling such that we get a lower condition number. Similarly, depending on the sampling instants, sampling at three times the baud rate can give us square matrices with RCOND numbers 0.0021, 0.0046, or 0.0134. For our simulations we have assumed the sampling rates to be such that the RCOND number is the worst of the possible values, that is, RCOND = 0.0021. Since the matrices are so ill-conditioned, it is necessary to use error-correction and all results presented below are with error correction.

To obtain a performance measure of the channel estimation, the normalized root-mean-square error (NRMSE) of the estimator is defined by

$$\text{NRMSE} = \frac{1}{\|\mathbf{h}\|}\sqrt{\frac{1}{M}\sum_{i=1}^{M}\|\hat{\mathbf{h}}_{(i)} - \mathbf{h}\|^2}, \tag{40}$$

where M is the number for independent trials, and $\hat{\mathbf{h}}_{(i)}$ is the estimate of the channel from the ith trial. Figure 6a shows the NRMSE versus SNR for different algorithms. In Figure 6b we have shown the bit error rate (BER) in estimating the transmitted sequences for the different algorithms. Figure 6c shows the computational complexity of different algorithms (measured by the number of floating point operations—FLOPS in MATLAB). As we can see from these figures, the BER is quite small and drops very sharply for all the subspace algorithm with an increase in the SNR. Also we can clearly see that the values of both the NRMSE and the BER for the generalized subspace algorithms presented here are lower than for other algorithms in the literature in the low SNR situation. In particular, the twice baud rate sampling case consistently outperforms the rest.

Figures 3a and 3b show 20 channel estimates obtained by oversampling at twice the baud rate (the observation interval is 40 and 100 symbols for the two figures and SNR in both cases is 25 dB). Figures 3c, 3d, 3e, and 3f show 20 estimates obtained by oversampling at 3 and 4 times the baud rate. Figures 4a, 4b, 4c, 4d, 4e, and 4f show the 20 channel estimates obtained using the algorithms in [8, 9, 10]. From the figures we can see that the generalized subspace algorithms presented in this paper perform better than the other algorithms in the literature. Figures 5a, 5b, 5c, and 5d show 20

channel estimates using the 2-times oversampling subspace algorithm with only 40 symbols for 20, 15, 10, and 5 dB SNR, respectively.

The figures clearly demonstrate how as the condition number of the matrix becomes better (with lower sampling the condition number can get better), the performance improves. Sampling at twice the baud rate improves the condition number and hence the performance as compared to sampling at 4 times the baud rate. Also, just reducing the sampling rate also improves the performance—even though we chose the worst cases of sampling at thrice baud rate, still the performance is better than the 4-times baud rate sampling case.

Figure 6c is the computational complexity plot. It shows that the complexity of the generalized subspace algorithm presented here is comparable to those of the other algorithms in the literature. As should be expected, lowering the sampling increases the complexity—the reason being that the lower the sampling rate, the bigger the effective $\mathbf{H}$ (with zero-paddings) becomes. Since the complexity of the subspace approach (because of the singular value decomposition for finding the null space) depends on the size of this matrix, as this size increases, we can expect the complexity to also increase.

4.2. Simulations results for the MIMO algorithm

We will now present the results for the MIMO-subspace algorithm and compare it with the MIMO equalization algorithm presented in [14]. We will present the results first without using error-correction and then with error-correction and demonstrate the algorithm's effectiveness in both cases.

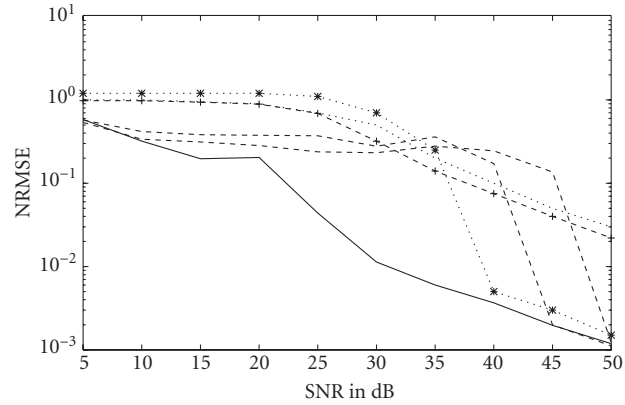
The simulation study presented here is for the case when $d = 2$, $K = 4$, and $J = 2$. The impulse responses used in the simulation for both the algorithms is given as

$$\mathbf{H} = \begin{bmatrix} 0.1667 & 0.1057 & -0.0665 & 0.3232 \\ 0.4813 & 0.5333 & 0.4369 & 0.3846 \\ 0.2804 & -0.0505 & 0.2208 & -0.0067 \\ 0.3730 & 0.4744 & 0.4296 & 0.5090 \end{bmatrix}. \tag{41}$$

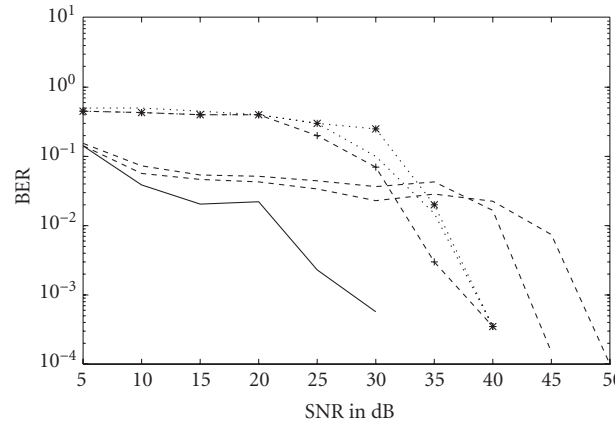
Figures 7a and 7b show the estimates[5] of the channel response using the algorithm in [14] for SNR = 25 dB. In both cases, the length of the observation interval covers 100 symbols. Figures 7c and 7d show the estimates obtained by using the subspace algorithm without error correction and using an observation interval of only 50 symbols. The SNR in this case is also 25 dB. These figures show that the subspace algorithm (even without error correction) performs much better than the algorithm in [14].

Figures 8a and 8b show the NRMSE of the algorithm versus SNR for the two input sequences. In Figures 9a and 9b we have shown the BER in estimating the two input sequences. As can be seen from the figure, the BER drops very

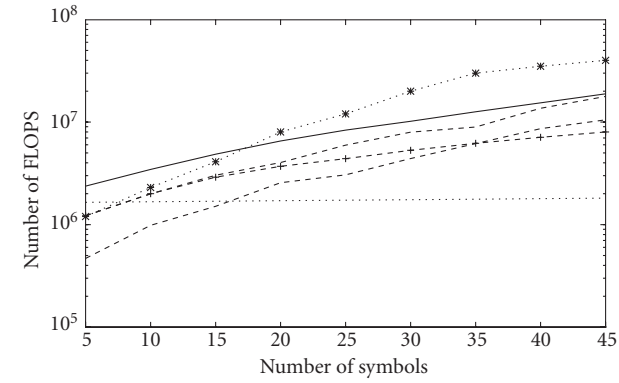
[5]We have combined the channel responses of the four channels $h_{i,1}(n)$, $1 \le i \le 4$, and the four channels $h_{i,2}(n)$, $1 \le i \le 4$, to form the two channel responses shown in the figures.



(a)



(b)



(c)

SIMO algorithm, twice baud rate sampling
SIMO algorithm, thrice baud rate sampling
SIMO algorithm, four baud rate sampling
Algorithm in [9]
Algorithm in [11]
Algorithm in [10]

Figure 6: (a) NRMSE versus SNR, 35 symbols. (b) BER versus SNR, 35 Symbols. (c) Computational complexity, SNR = 30 dB.

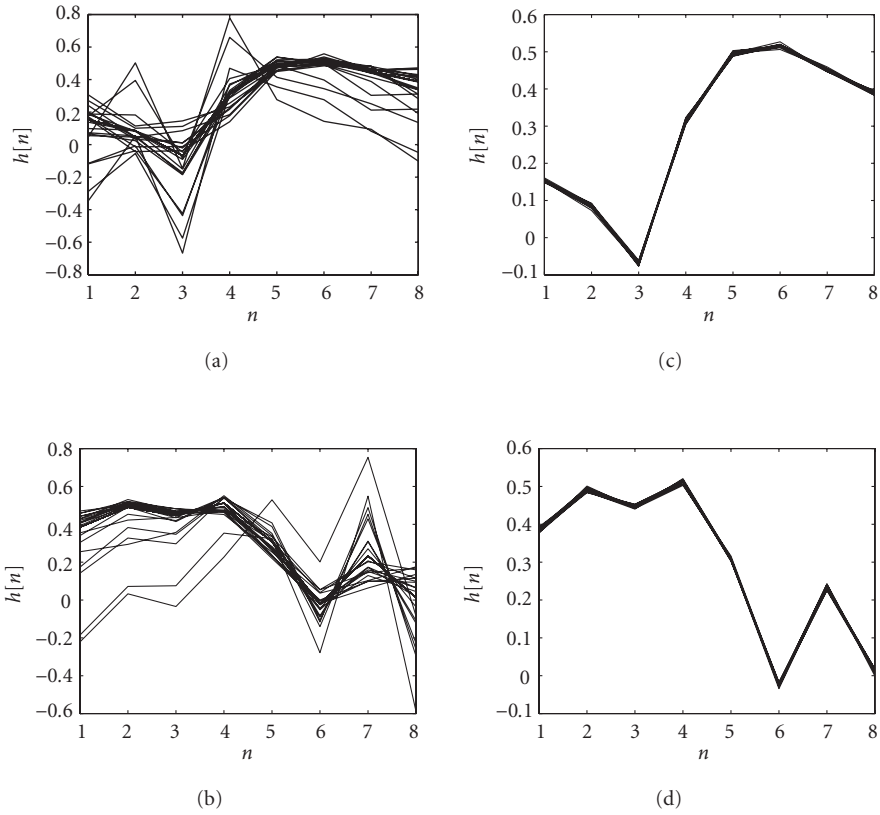


Figure 7: (a), (b) Algorithm in [14], 100 symbols, SNR = 25 dB. (c), (d) MIMO subspace algorithm without error correction, 50 symbols, SNR = 25 dB.

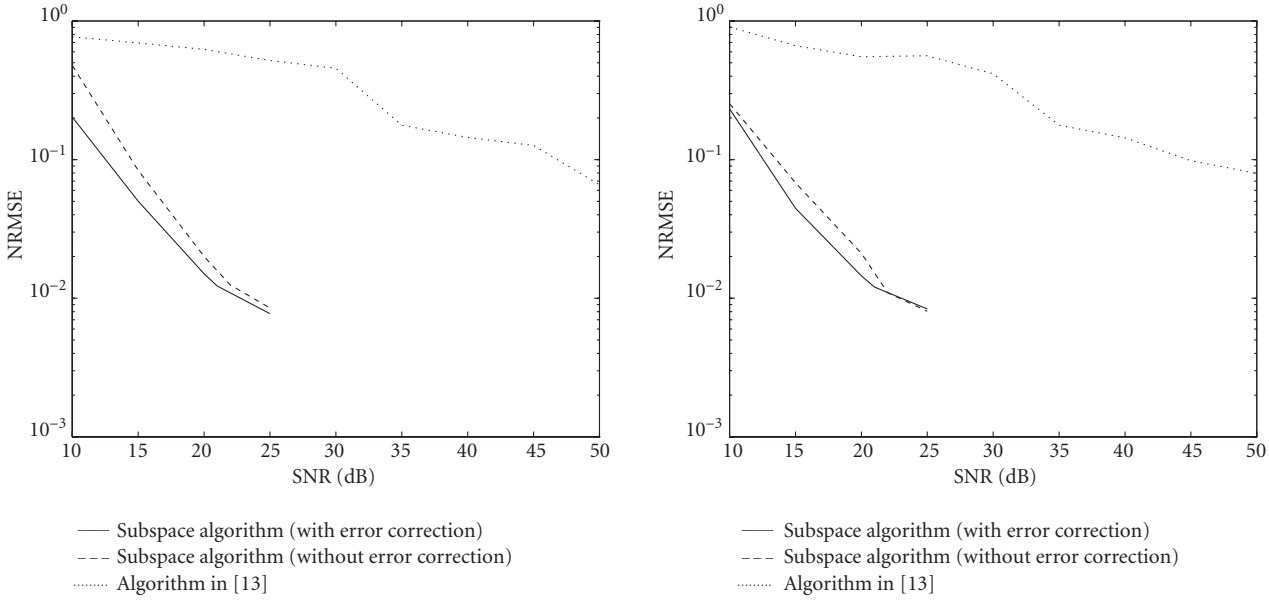


Figure 8: NRMSE versus SNR for the two input sequences. 100 symbols used for each estimate.

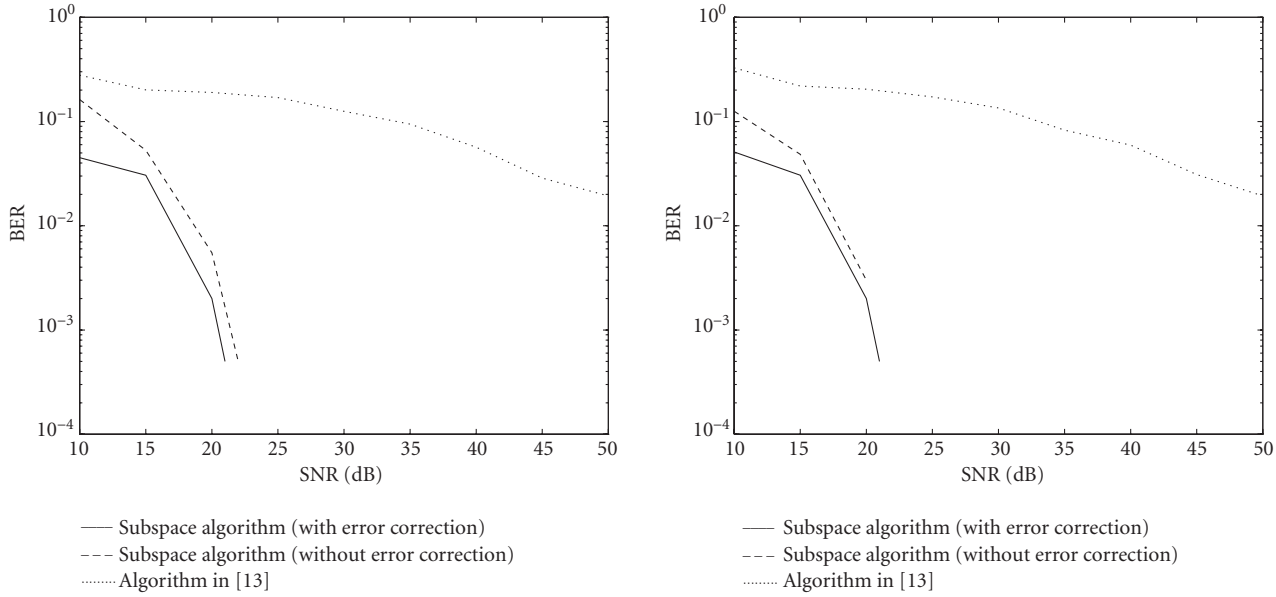


Figure 9: BER versus SNR for the two input sequences. 100 symbols used for each estimate.

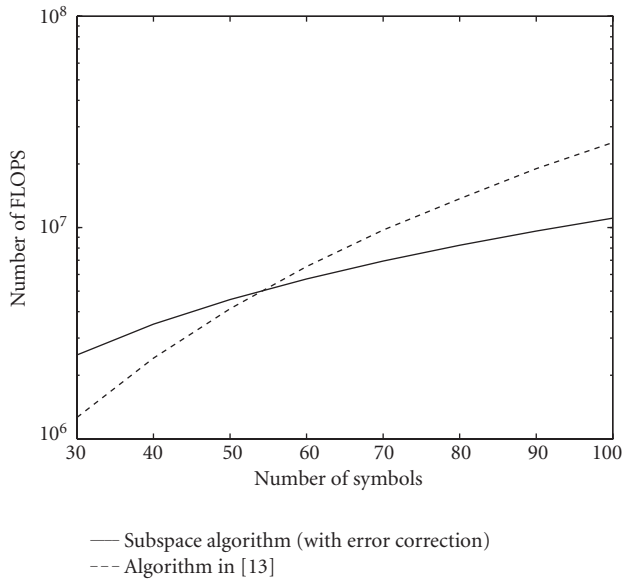


Figure 10: Computational complexity versus observation interval. SNR = 25 dB.

sharply for the proposed subspace algorithm (for both—with and without error correction) with an increase in the SNR. Also we can clearly see that both the values of the NRMSE and BER for the subspace algorithm are much lower than their values for the algorithm in [14]. Figure 10 shows the computational complexity of this algorithm (using error correction) and the algorithm in [14]. As the number of symbols increases, the complexity of the algorithm in [14] rapidly increases and becomes much more computationally intensive than the MIMO subspace algorithm presented here.

5. CONCLUSIONS

In this paper, we generalized the basic subspace algorithm presented in [11, 12]. This generalization allows us to use it even in cases when the sampling rate cannot be chosen to be equal to the FIR channel length. We compared this result to other algorithms and found that this algorithm is more general than the one in [11, 12] and performs much better than other algorithms in the literature [8, 9, 10]. The simulation study also showed that for ill-conditioned channels as the sampling rate is reduced, the performance becomes better.

In this paper, we have also proposed a new algorithm for the blind equalization and identification of MIMO systems. This algorithm uses a subspace based approach in combination with a searching procedure based on the finite alphabet property of the input sequence. As the simulation results show, this algorithm needs fewer symbols than the algorithm in [14] for obtaining a good estimate of the MIMO channel response as well as the transmitted sequences. The accuracy of the channel estimate is also better than [14]. Finally, this algorithm is quite robust to noise and computationally also more efficient than the algorithm in [14].

APPENDIX

Proof of Lemma 1. Let the vector β be a vector in the null space of $\mathbf{\Gamma}$ where

$$\beta = \Big[\beta_{1,1}, \beta_{2,1}, \ldots, \beta_{J,1}, \beta_{1,2}, \beta_{2,2}, \ldots, \beta_{J,2}, \ldots, \boldsymbol{\beta}_{1,J}, \beta_{2,J}, \ldots, \beta_{J,J}\Big]^T. \tag{A.1}$$

Then

$$\mathbf{\Gamma}\beta = 0. \tag{A.2}$$

From this equation, and, using the fact that since the vectors $\mathcal{B}_0, \mathcal{B}_1, \ldots, \mathcal{B}_J$ are linearly independent, a linear combination of them cannot be zero unless all the corresponding coefficients are zero, we get the following equations:

$$\begin{aligned}
\beta_{1,i} &= 0 \quad \forall i = 2, 3, \ldots, J-1, J, \\
\beta_{1,i} - \beta_{2,i+1} &= 0 \quad \forall i = 1, 2, 3, \ldots, J-1, \\
\beta_{2,i} - \beta_{3,i+1} &= 0 \quad \forall i = 1, 2, 3, \ldots, J-1, \\
&\vdots \\
\beta_{J-1,i} - \beta_{J,i+1} &= 0 \quad \forall i = 1, 2, 3, \ldots, J-1, \\
\beta_{J,i} &= 0 \quad \forall i = 1, 2, 3, \ldots, J-1.
\end{aligned} \tag{A.3}$$

From these equations it can be concluded that $\beta_{i,j}$ for $i \neq j$ are all zero and $\beta_{1,1} = \beta_{2,2} = \cdots = \beta_{J,J}$. This means that the vector β is fixed up to a multiplication factor and since β is an arbitrary vector in the null space of $\mathbf{\Gamma}$, we infer that $\mathbf{\Gamma}$ has a one-dimensional null space generated by the vector $[\mathbf{e}_1 \ \mathbf{e}_2 \ \mathbf{e}_3 \ \cdots \ \mathbf{e}_J]^T$. □

REFERENCES

[1] G. Giannakis, Y. Inouye, and J. Mendel, "Cumulant-based identification of multichannel moving average models," *IEEE Trans. on Automatic Control*, vol. 34, no. 7, pp. 783–787, 1989.

[2] G. Giannakis and J. Mendel, "Identification of nonminimum phase systems using higher-order statistics," *IEEE Trans. Acoustics, Speech, and Signal Processing*, vol. 37, no. 3, pp. 360–377, 1989.

[3] J. Mendel, "Tutorial on higher order statistics (spectra) in signal processing and system theory: theoretical results and some applications," *Proceedings of the IEEE*, vol. 79, no. 3, pp. 278–305, 1991.

[4] C. L. Nikias, "Blind deconvolution using higher-order statistics," in *Proc. 2nd Int. Conf. Higher-Order Statistics*, pp. 49 –56, Elsevier, 1992.

[5] L. Tong, G. Xu, and T. Kailath, "Blind identification and equalization based on second-order statistics: a time domain approach," *IEEE Transactions on Information Theory*, vol. 40, no. 2, pp. 340–349, 1994.

[6] Y. Li and Z. Ding, "Blind channel identification based on second order cyclostationary statistics," in *Proc. IEEE Int. Conf. Acoustics, Speech, Signal Processing*, vol. 4, pp. 81–84, Minneapolis, Minn, USA, April 1993.

[7] D. T. M. Slock, "Blind fractionally-spaced equalization, perfectreconstruction filter banks and multichannel linear prediction," in *Proc. IEEE Int. Conf. Acoustics, Speech, Signal Processing*, vol. 4, pp. 585–588, May 1994.

[8] E. Moulines, P. Duhamel, J. F. Cardoso, and S. Mayrargue, "Subspace methods for the blind identification of multichannel FIR filters," *IEEE Trans. Signal Processing*, vol. 43, no. 2, pp. 516–525, 1995.

[9] G. Xu, H. Liu, L. Tong, and T. Kailath, "A least-squares approach to blind channel identification," *IEEE Trans. Signal Processing*, vol. 43, no. 12, pp. 2982–2993, 1995.

[10] H. Liu and G. Xu, "Closed-form blind symbol estimation in digital communications," *IEEE Trans. Signal Processing*, vol. 43, no. 11, pp. 2714–2723, 1995.

[11] B. Sampath, Y. Li, and K. J. R. Liu, "A subspace based blind identification and equalization algorithm," in *Proc. 1996 IEEE International Conference on Communications*, vol. 2, pp. 1010–1014, Dallas, Tex, USA, June 1996.

[12] B. Sampath, Y. Li, and K. J. R. Liu, "Error correcting least squares subspace algorithm for blind identification and equalization," *Signal Processing*, vol. 81, no. 10, pp. 2069–2087, 2001.

[13] Y. Li and K. J. R. Liu, "Adaptive blind source separation and equalization for multiple-input/multiple-output systems," *IEEE Transactions on Information Theory*, vol. 44, no. 7, pp. 2864–2876, 1998.

[14] A. J. van der Veen, S. Talwar, and A. Paulraj, "Blind estimation of multiple digital signals transmitted over FIR channels," *IEEE Signal Processing Letters*, vol. 2, no. 5, pp. 99–102, 1995.

[15] S. Talwar, M. Viberg, and A. Paulraj, "Blind estimation of multiple co-channel digital signals using an antenna array," *IEEE Signal Processing Letters*, vol. 1, no. 2, pp. 29–31, 1994.

Balaji Sampath received his B.S. degree from the Indian Institute of Technology, Madras, in 1994 and Ph.D. degree from University of Maryland, College Park, in 1997. Dr. Sampath ranked All Indian Fourth in the national examination to Indian Institute of Technology, and received the Institute Merit Prize during his study there. He stood All Indian First in the Physics Talent Test in 1990 and represented his state in the Indian National Math Olympiad. Dr. Sampath received Graduate School Fellowship when attending University of Maryland. Dr. Sampath is a founder of the Association for *India's Development (AID)* which is a charitable organization offering volunteering services in poor and remote villages of India. Since his graduation, Dr. Sampath has been a volunteer of AID helping unfortunate people in remote villages of India.

K. J. Ray Liu received his B.S. degree from the National Taiwan University, and the Ph.D. degree from UCLA, both in electrical engineering. He is Professor of Electrical and Computer Engineering Department of University of Maryland, College Park. His research interests span broad aspects of signal processing architectures; multimedia signal processing; wireless communications and networking; information security; and bioinformatics in which he has published over 230 refereed papers, of which over 70 are in archival journals. Dr. Liu is the recipient of numerous awards including the 1994 National Science Foundation Young Investigator, the IEEE Signal Processing Society's 1993 Senior Award, IEEE 50th Vehicular Technology Conference Best Paper Award, Amsterdam, 1999. He also received the George Corcoran Award in 1994 for outstanding contributions to electrical engineering education and the Outstanding Systems Engineering Faculty Award in 1996 in recognition of outstanding contributions in interdisciplinary research, both from the University of Maryland. Dr. Liu is Editor-in-Chief of the EURASIP Journal on Applied Signal Processing, and has been an Associate Editor of IEEE Transactions on Signal Processing, a Guest Editor of special issues on Multimedia Signal Processing of Proceedings of the IEEE, a Guest Editor of special issue on Signal Processing for Wireless Communications of IEEE Journal of Selected Areas in Communications, a Guest Editor of special issue on Multimedia Communications over Networks of IEEE Signal Processing Magazine, a Guest Editor of special issue on Multimedia over IP of IEEE Trans. on Multimedia, and an editor of Journal of VLSI Signal Processing Systems.

Ye (Geoffrey) Li was born in Jiangsu, China. He received his B.S.E. and M.S.E. degrees in 1983 and 1986, respectively, from the Department of Wireless Engineering, Nanjing Institute of Technology, Nanjing, China, and his Ph.D. degree in 1994 from the Department of Electrical Engineering, Auburn University, Alabama. From 1986 to 1991, he was a Teaching Assistant and then a Lecturer with Southeast University, Nanjing, China. From 1991 to 1994, he was a Research and Teaching Assistant with Auburn University, Alabama. From 1994 to 1996, he was a Post-Doctoral Research Associate with the University of Maryland at College Park, Maryland. From 1996 to 2000, he was with AT&T Labs-Research at Red Bank, New Jersey. Since August 2000, he has been with Georgia Tech as an Associate Professor. His general research interests include statistical signal processing and wireless mobile systems with emphasis on signal processing in communications. He once served as a Guest Editor for special issues on Signal Processing for Wireless Communications for the IEEE Journal on Selected Areas in Communications and is currently serving as an editor for Wireless Communication Theory for the IEEE Transactions on Communications and an editorial board member of EURASIP Journal on Applied Signal Processing.

:: Open Access ::

EURASIP Journal on

Audio, Speech, and Music Processing

http://www.hindawi.com/journals/asmp/

Special Issue on

Intelligent Audio, Speech, and Music Processing Applications

Call for Papers

Future audio, speech, and music processing applications need innovative intelligent algorithms that allow for human/environmental-based interactive interfaces with surrounding devices/systems in real-world settings. The need exists to control, process, render, and playback/project sound signals for different platforms under a diverse range of listening environments. The presence of these intelligent audio, speech, and music processing applications will create an environment that is sensitive, adaptive, and responsive to the presence of users.

Generally, three areas of research are considered, namely, analysis, communication, and interaction. Analysis covers both preprocessing of sound signals and extraction of information from the environment. Communication covers the transmission path/network, coding techniques, and conversion between spatial audio formats. The final area involves intelligent interaction with the audio/speech/music environment based on the users' location, signal information, and acoustical environment.

This special issue includes a variety of intelligent devices and applications working together to create an integral audio experience/environment for the users. A vision of intelligent signal processing environments and applications can also be proposed and describe how new audio/speech/music applications can enhance everyday entertainment as well as human communication experiences.

Topics of interest include (but are not limited to):

- 3D and spatial audio
- Automotive audio
- Audio/speech installation
- Audio network
- Audio/speech delivery
- Audio/speech for mobile and handheld/wearable devices
- Environmental noise controlactive & passive
- Embedded audio/speech intelligence
- Intelligent speech communication
- Coding techniques
- Music processing
- Personal soundscape/personal audio space
- Speech enhancement
- New applications/domains

Authors should follow the EURASIP Journal on Audio, Speech, and Music Processing manuscript format described at the journal site http://www.hindawi.com/journals/asmp/. Prospective authors should submit an electronic copy of their complete manuscript through the journal Manuscript Tracking System at http://mts.hindawi.com/, according to the following timetable:

Manuscript Due	November 1, 2007
First Round of Reviews	February 1, 2008
Publication Date	May 1, 2008

Guest Editors

Woon-Seng Gan, Digital Signal Processing Laboratory, School of Electrical and Electronic Engineering, Nanyang Technological University, 50 Nanyang Avenue, Singapore 639798; ewsgan@ntu.edu.sg

Sen M. Kuo, Department of Electrical Engineering, Northern Illinois University, Dekalb, IL 60115, USA; kuo@ceet.niu.edu

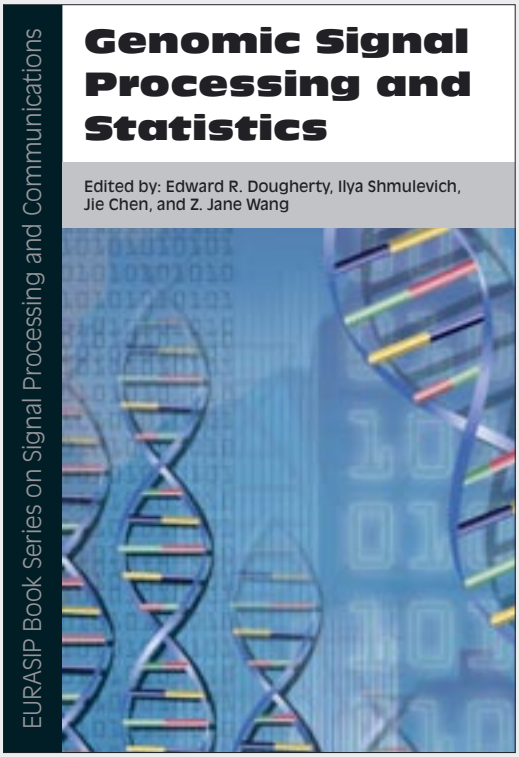
John H. L. Hansen, Department of Electrical Engineering, University of Texas at Dallas, Richardson, TX 75083, USA; john.hansen@utdallas.edu

Hindawi Publishing Corporation
410 Park Avenue, 15th Floor, #287 pmb, New York, NY 10022, USA

HINDAWI

GENOMIC SIGNAL PROCESSING AND STATISTICS

Edited by: Edward R. Dougherty, Ilya Shmulevich, Jie Chen, and Z. Jane Wang



Recent advances in genomic studies have stimulated synergetic research and development in many cross-disciplinary areas. Genomic data, especially the recent large-scale microarray gene expression data, represents enormous challenges for signal processing and statistics in processing these vast data to reveal the complex biological functionality. This perspective naturally leads to a new field, genomic signal processing (GSP), which studies the processing of genomic signals by integrating the theory of signal processing and statistics. Written by an international, interdisciplinary team of authors, this invaluable edited volume is accessible to students just entering this emergent field, and to researchers, both in academia and industry, in the fields of molecular biology, engineering, statistics, and signal processing. The book provides tutorial-level overviews and addresses the specific needs of genomic signal processing students and researchers as a reference book.

The book aims to address current genomic challenges by exploiting potential synergies between genomics, signal processing, and statistics, with special emphasis on signal processing and statistical tools for structural and functional understanding of genomic data. The book is partitioned into three parts. In part I, a brief history of genomic research and a background introduction from both biological and signal processing/statistical perspectives are provided so that readers can easily follow the material presented in the rest of the book. In part II, overviews of state-of-the-art techniques are provided. We start with a chapter on sequence analysis, and follow with chapters on feature selection, clustering, and classification of microarray data. The next three chapters discuss the modeling, analysis, and simulation of biological regulatory networks, especially gene regulatory networks based on Boolean and Bayesian approaches. The next two chapters treat visualization and compression of gene data, and supercomputer implementation of genomic signal processing systems. Part II concludes with two chapters on systems biology and medical implications of genomic research. Finally, part III discusses the future trends in genomic signal processing and statistics research.

EURASIP Book Series on SP&C, Volume 2, ISBN 977-5945-07-0.
Please visit http://www.hindawi.com/spc.2.html for more information about the book. To place an order while taking advantage of our current promotional offer, please contact books.orders@hindawi.com