A FAST BLIND SOURCE SEPARATION FOR DIGITAL WIRELESS APPLICATIONS

M. Torlak, L. K. Hansen, and G. Xu

Depart. of Elect. & Comp. Eng., The University of Texas at Austin, Austin, TX 78712-1084, USA
E-mail:torlak@globe.ece.utexas.edu,lkhansen@globe.ece.utexas.edu,xu@globe.ece.utexas.edu

ABSTRACT

The problem of *blindly* estimating multiple digital co-channel communication signals using an antenna array is studied in the presence of multipath fading. We develop a fast sequential-estimation algorithm for separating multi-user signals based on the geometric observation made by Hansen and Xu in [2]. When the signals are constrained to a finite alphabet, it is possible to visualize the geometric properties of the problem, which can be exploited to sequentially extract the digital co-channel communication signals. We will present simulation results comparing speed and BER performance with different methods.

1. INTRODUCTION

A source separation problem is considered in the context of blind joint space-time equalization of multiple digital signals. Such a problem is motivated by digital wireless communications signals transmitted over multipath channels. To solve this problem, we have two properties that we can exploit. The first is that the signals share a known M-ASK or QAM digital signaling alphabet. The second is that the signals have different, but unknown, spatio-temporal characteristics as measured through an antenna array and/or oversampling.

In a deterministic discrete-time setting of a multiple-inputs, multiple-outputs (MIMO) system, the block-Toeplitz structure has been exploited by subspace-based algorithms. Recently, this property has been exploited to deconvolve the effects of channels (called multi-channel deconvolution (MCD)) by the different researchers [9, 8]. However, the MCD only converts the FIR-MIMO problem into the problem of the separation of instantaneous multiple signals. If the signals possess the property of the finite alphabet, some simplifications occur. For separation, these simplifications have led to the development of the finite-alphabet algorithms. Among them, the ILSP and ILSE described in [6, 7] are the most famous ones. The MCD and finite-alphabet algorithms have been readily combined by the different researchers in order to solve the FIR-MIMO problem. Since MCD is quite costly, it cannot be readily used in practical systems where the speed is paramount. This was our main motivation for this paper.

In this paper, we show that we can eliminate computationally intensive multi-channel deconvolution. Although we use the original geometric observations made by Hansen and Xu, we extend the present paper beyond [2] by providing a significantly more efficient implementation for FIR-MIMO systems. Otherwise, the original approach would require us to work on the outputs of MCD of FIR-MIMO sytems.

2. PROBLEM STATEMENT

We consider the general digital co-channel communication system with d ($d > 1$) users and with an array of M antennas. The vector of array outputs at a time k through multipath channels $\{\mathbf{h}_i\}$

$$\mathbf{x}(k) = \sum_{i=1}^{d}\sum_{j=0}^{L_i} \begin{bmatrix} h_{1,i}(j) \\ \vdots \\ h_{M,i}(j) \end{bmatrix} s_i(k-j) \tag{1}$$

where L_i is the maximum order of the ith user's M channels ($M \geq 2$ and $L \geq 1$). For simplicity of the derivation, we initially assume that all the channels are equal with the length of L. We do not address the general case due to space limitations.

For a finite number of samples, we describe the algebraic relation between the input and output

$$\mathbf{X} = \mathbf{HS} = \begin{bmatrix} \mathbf{H}_1 & \mathbf{H}_2 & \cdots & \mathbf{H}_d \end{bmatrix} \begin{bmatrix} \mathbf{S}_1(r) \\ \mathbf{S}_2(r) \\ \vdots \\ \mathbf{S}_d(r) \end{bmatrix}. \tag{2}$$

where $\mathbf{H}_i\mathbf{S}_i(r)$ is given by

$$\begin{bmatrix} \mathbf{h}_i(L-1) & \cdots & \mathbf{h}_i(0) & \cdots & \mathbf{0} \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ \mathbf{0} & \cdots & \mathbf{h}_i(L-1) & \cdots & \mathbf{h}_i(0) \end{bmatrix} \times \begin{bmatrix} s_i(1) & s_i(2) & \cdots & s_i(N-r+1) \\ s_i(2) & s_i(3) & \cdots & s_i(N-r+2) \\ \vdots & \vdots & \cdots & \vdots \\ s_i(r) & s_i(r+1) & \cdots & s_i(N) \end{bmatrix}. \tag{3}$$

In-phase and quadrature sampling causes the case of $\mathbf{H}$ complex which requires a slight modification to the above procedure. We reduce this case to the real case by redefining

$$\mathbf{X} \equiv \begin{bmatrix} \tilde{\mathbf{X}}_R \\ \tilde{\mathbf{X}}_I \end{bmatrix} = \begin{bmatrix} \tilde{\mathbf{H}}_R \\ \tilde{\mathbf{H}}_I \end{bmatrix} \mathbf{S} + \begin{bmatrix} \mathbf{V}_R \\ \mathbf{V}_I \end{bmatrix} \equiv \mathbf{HS} + \mathbf{V}. \tag{4}$$

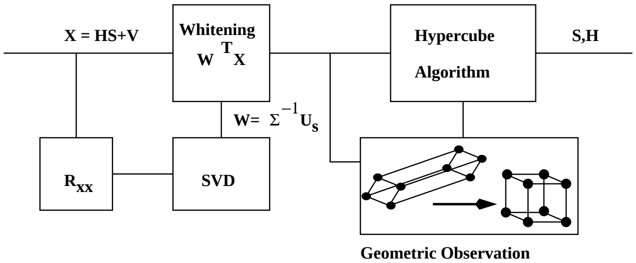

Figure 1. Source Separation with Geometric Ideas

This effectively doubles the number of antennas and the equations.

Because our algorithm operates on a whitened and dimension-reduced space, we first obtain the source correlation matrix from the deterministic correlation of the received signal dataset. Let

$$\mathbf{R_{XX}} = [\mathbf{U}_s \ \mathbf{U}_o] \begin{bmatrix} \mathbf{\Sigma}_s^2 & \mathbf{0} \\ \mathbf{0} & \mathbf{\Sigma}_o^2 \end{bmatrix} \begin{bmatrix} \mathbf{U}_s^T \\ \mathbf{U}_o^T \end{bmatrix} \tag{5}$$

where the whitened and dimension-reduced problem is defined by $\mathbf{Y} = \mathbf{WX} = \mathbf{W\tilde{H}S} + \mathbf{W\tilde{V}} = \mathbf{HS} + \mathbf{V}$ where $\mathbf{W}$ is the whitening matrix. And $\mathbf{W}$ is obtained by $\mathbf{W} \equiv \mathbf{\Sigma}_s^{-1}\mathbf{U}_s^T$.

3. A GEOMETRIC-BASED APPROACH

Our blind source separation scheme is based on the geometric properties of the problem when M antennas receive multiple synchronous users with a common digital communication method via memoryless channels. Let us briefly describe the geometric ideas which were originally developed for multiple synchronous users in [2]. We first consider the BPSK case. We assume that $\mathbf{S}$ consists of 2^{Ld} distinct columns. This special signal matrix $\mathbf{S}$ will be denoted as $\mathbf{S}_{Ld}$. We can think of the 2^{Ld} columns of $\mathbf{S}_{Ld}$ as points in $\mathcal{R}^{Ld}$ defining the vertices of a Ld-cube or a hypercube. Note that the linear transformation of $\mathbf{S}_{Ld}$ by $\mathbf{H}$ maps the cube into a parallelogram. If $\mathbf{H}$ is unitary, we get a rotated cube. In the next section, we will explain how to get a approximate unitary $\mathbf{H}$ from the noisy dataset, *i.e.*,$\mathbf{X} = \mathbf{HS} + \mathbf{V}$. We can express $\mathbf{S}_{Ld}$ mathematically as

$$\mathbf{S}_{Ld} = \bigcap_{i=1}^{Ld} \bigcup_{\rho \in \mathcal{S}} H(\mathbf{e}_i, \rho). \tag{6}$$

where $H(\mathbf{e}_i,\rho)$ defines a hyperplane with normal vector $\mathbf{e}_i$ (the ith unit vector) and offset ρ. In this paper, we will consider only the real M-ASK alphabet

$$\mathcal{S}_{\text{M-ASK}} = \{\pm 1, \pm 3, \cdots, \pm(\mathcal{M}-1)\}, \tag{7}$$

The real data space $\mathbf{X} \equiv \mathbf{HS}_{Ld}$ then can be expressed as

$$\mathbf{X}_{Ld} = \bigcap_{i=1}^{Ld} \bigcup_{\rho \in \mathcal{S}} H(\mathbf{H}^{-T}\mathbf{e}_i, \rho). \tag{8}$$

Note that the transpose of the normal vectors in this equation, *i.e.*, $\mathbf{e}_i^T\mathbf{H}^{-1}$, are the rows of $\mathbf{H}^{-1}$. Thus finding one of the normal vectors is equivalent to finding a row of $\mathbf{H}^{-1}$ up to a scale factor.

3.1. Extended Hyperplane Algorithm

To identify a single row of $\mathbf{S}$ from the data $\mathbf{X}$, we would estimate a single row of $\mathbf{H}^{-1}$ and perform multiplication on $\mathbf{X}$ from the left. As discussed in the previous section, in the real case we can think of our estimated vector as defining a hyperplane that partitions the space. Recalling equation 8, what we would like is a hyperplane passing through the origin that is parallel to a pair of "sides" of the convex hull of $\mathbf{X}$. The resulting normal vector would take the form $\mathbf{H}^{-T}\mathbf{e}_i$. This result was described with a theorem in [1]. Although the the assumptions of the theorem appear quite restrictive, with $\mathbf{H}$ unitary, noise $\mathbf{V} = \mathbf{0}$, and special $\mathbf{S} = \mathbf{S}_{Ld}$, the more general problem $\mathbf{X} = \mathbf{HS} + \mathbf{WV}$ inherits enough of the behavior of the objective function to yield useful estimates. Note that in Theorem in [1], the objective function is invariant with respect to a re-ordering of the columns of $\mathbf{X}_{Ld}$ (or equivalently $\mathbf{S}_{Ld}$). If, in addition to the assumptions of Theorem in [1], we also assume that $\mathbf{S}_{Ld}$ has a block-Toeplitz structure, then we can apply the following theorem.

Theorem 1 *Assume $\mathbf{X}_{Ld} = \mathbf{HS}_{Ld}$, with $\mathbf{S}_{Ld}$ having structure as above. Perform the following maximization:*

$$\max_{\|\boldsymbol{\alpha}_l\|_2=1} \sum_{l=1}^{L} \sum_{\mathbf{x}\in\mathbf{X}} |\boldsymbol{\alpha}_l^T\mathbf{X}| \tag{9}$$

subject to

$$\|\boldsymbol{\alpha}_l\|_2 = 1$$
$$\mathit{signum}(\boldsymbol{\alpha}_{l+1}^T\mathbf{X}_{Ld}) = \mathit{shift}\left[\mathit{signum}(\boldsymbol{\alpha}_l^T\mathbf{X}_{Ld})\right] \tag{10}$$

where $l = 1, \ldots, L-1$; shift $[\cdot]$ operator is defined as

$$[s(k+1) \ \cdots \ s(N)] = \mathit{shift}\,[s(k) \ \cdots \ s(N-1)]\,.$$

Assume $\mathbf{H}$ is ordered

$$\mathbf{H} = \left[\ \mathbf{H}_1 \quad \cdots \quad \mathbf{H}_i \quad \cdots \quad \mathbf{H}_L\ \right]. \tag{11}$$

Then

$$[\boldsymbol{\alpha}_1 \ \cdots \ \boldsymbol{\alpha}_L] = \pm\mathbf{H}^{-T} \ \left[\mathbf{e}_{d(i-1)+1} \ \cdots \ \mathbf{e}_{d(i-1)+L}\right] \tag{12}$$

is a globally maximizing solution to our problem.

The proof can be found in [10].

We can use a very simple gradient search to maximize the objective function in Theorem 1. More importantly, this simple gradient search generalizes to the case of noisy data $\mathbf{X} = \mathbf{HS} + \mathbf{WV}$. Consider

$$\sum_{\mathbf{x}\in\mathbf{X}} \left|\boldsymbol{\alpha}_l^T\mathbf{x}\right| = \boldsymbol{\alpha}_l^T \sum_{\mathbf{x}\in\mathbf{X}} \text{signum}(\boldsymbol{\alpha}_l^T\mathbf{x})(\mathbf{x}) \equiv \boldsymbol{\alpha}_l^T\boldsymbol{\beta}_l, \tag{13}$$

where $\{\boldsymbol{\beta}_l\}$ are the local gradients.

Because of the deterministic data setting, we can force the block Toeplitz structure for each user. In this case, we extract the Toeplitz row block from the signal matrix $\mathbf{S}$. We estimate L rows of $\mathbf{H}$ simultaneously, in order to enforce the Toeplitz structure on the partition vectors $\{\mathbf{p}_{n,0}, \mathbf{p}_{n,2}, \cdots, \mathbf{p}_{n,L-1}\}$. We use a majority rule in constructing a block Toeplitz matrix required for the next iteration. We use this gradient search as the basis for our Hyperplane subroutine which follows:

1. Given whitened real dataset $\mathbf{X} = \mathbf{HS} + \mathbf{WV}$, initial estimate of the matrix $\mathcal{A}_0 = [\boldsymbol{\alpha}_{0,0} \ \cdots \ \boldsymbol{\alpha}_{0,L-1}]$, and $n = 0$.
2. (a) $n = n + 1$

   (b) $\begin{bmatrix} \mathbf{p}_{n,0} \\ \vdots \\ \mathbf{p}_{n,L-1} \end{bmatrix} = \text{signum}\left(\begin{bmatrix} \boldsymbol{\alpha}_{n-1,0}^T \\ \vdots \\ \boldsymbol{\alpha}_{n-1,L-1}^T \end{bmatrix} \mathbf{X} \right)$

   (c) Use a majority rule in forcing a block-Toeplitz structure on $\begin{bmatrix} \mathbf{p}_{n,0}^T & \cdots & \mathbf{p}_{n,L-1}^T \end{bmatrix}$ and form the block-Toeplitz partition matrix

   $$\mathbf{P}_n = \begin{bmatrix} \hat{\mathbf{p}}_n^T(N) & \cdots & \hat{\mathbf{p}}_n^T(N-L+1) \end{bmatrix}$$

   (d) $\begin{bmatrix} \boldsymbol{\beta}_{n,0} & \cdots & \boldsymbol{\beta}_{n,L-1} \end{bmatrix} = \begin{bmatrix} \frac{\mathbf{XP}_{n,0}}{\|\mathbf{XP}_{n,0}\|} & \cdots & \frac{\mathbf{XP_{n,L-1}}}{\|\mathbf{XP}_{n,L-1}\|} \end{bmatrix}$

   (e) $\boldsymbol{\alpha}_{n,l} = \boldsymbol{\alpha}_{n-1,l} + \left[\mathbf{I} - \boldsymbol{\alpha}_{n-1,l}\boldsymbol{\alpha}_{n-1,l}^T\right] \boldsymbol{\beta}_{n,l}$

   (f) $\boldsymbol{\alpha}_{n,l} = \frac{\boldsymbol{\alpha}_{n,l}}{\|\boldsymbol{\alpha}_{n,l}\|} \qquad l = 0, \ldots, L-1$
3. Continue until $\|\mathcal{A}_n - \mathcal{A}_{n-1}\|$ converges
4. $\mathcal{A} = [\boldsymbol{\alpha}_{n,0} \ \cdots \ \boldsymbol{\alpha}_{n,L-1}]$,
5. $\mathbf{S} = {}_{\mathcal{S}}[\mathcal{A}^T \mathbf{X}]$
6. Return $\mathcal{A}$ and $\mathbf{S}^T$

The function ${}_{\mathcal{S}}[.]$ denotes a projection to the closest element in the alphabet $\mathcal{S}$. The projection matrix in step 2e projects $\boldsymbol{\beta}_{n,l}$ onto the constraint space $\boldsymbol{\alpha}_{n-1,l}^T \boldsymbol{\alpha}_{n-1,l} = 1$.

The Extended Hyperplane algorithm provides the estimates of $\mathcal{A}^H$ and $\hat{\mathbf{S}}_i^H$. $\hat{\mathbf{S}}_i^H$ is one of the block-Toeplitz submatrices of $\mathbf{S}$. We need to remove the parts corresponding to the formerly estimated the block-Toeplitz sub-matrices, *i.e.*, $\mathbf{S}_k = [\mathbf{S}_1^T \ \cdots \ \mathbf{S}_k^T]^T$, by *deflating* the dataset. In order to deflate the problem to dimension $(d-k)L$, we will remove the contributions of the currently estimated $\mathbf{S}_k$ from $\mathbf{X}$ by forming an oblique projection operator $\hat{\mathcal{N}}_k^T \mathbf{E}$ [10, 2]. However, we omit the derivation of $\hat{\mathcal{N}}_k^T \mathbf{E}$ due to space limitations.

3.2. Testing for the Global Maxima

The extended Hyperplane subroutine will converge to a global maximum if it is sufficiently close to that maximum. Because the subroutine may find local maxima, it is imperative to test for global convergence. Under the hypothesis that the subroutine has successfully estimated $\hat{\mathbf{S}}_{k+1}^T = \mathbf{S}_{k+1}^T$ and $\mathcal{A}_1^T = \begin{bmatrix} \boldsymbol{\alpha}_1 & \cdots & \boldsymbol{\alpha}_L \end{bmatrix}^T = (\hat{\mathcal{N}}_k^T \mathbf{H}_{d-k})^{-1} [\mathbf{e}_1 \ \ldots \ \mathbf{e}_L]$, then

$$\hat{\mathbf{S}}_{k+1}^T = \begin{bmatrix} \boldsymbol{\alpha}_1 & \cdots & \boldsymbol{\alpha}_L \end{bmatrix}^T \hat{\mathcal{N}}_k^T \mathbf{EX} \tag{14}$$

Using the derivations in [10], the residual term can be rewritten as

$$\mathbf{r}^T = \hat{\mathbf{S}}_{k+1} - \mathbf{S}_{k+1} = -\mathbf{S}_{k+1} \mathbf{P}_{\mathbf{S}_k} + (\mathcal{A}^T \hat{\mathcal{N}}_k^T \mathbf{EW}) \mathbf{V}. \tag{15}$$

Since our test is a statistical test, we need to find the total variance of the squared error of the residual.

$$\begin{aligned} \rho^2 \quad &= \quad \text{tr}\left[\mathcal{E}\{\mathbf{r}^T \mathbf{r}\}\right] = \text{tr}\left[\mathbf{S}_{k+1}^T \mathbf{P}_{\mathbf{S}_k} \mathbf{S}_{k+1}\right] \\ &\quad + \sum_{l=1}^{L} (\boldsymbol{\alpha}_l^T \hat{\mathcal{N}}_k^T \mathbf{EW}) \mathcal{E}\{\mathbf{V}\mathbf{V}^T\} (\boldsymbol{\alpha}_l^T \hat{\mathcal{N}}_k^T \mathbf{EW})^T. \end{aligned}$$

Under our hypothesis, the quantities $\mathbf{S}_{k+1}^T \mathbf{P}_{\mathbf{S}_k} \mathbf{S}_{k+1}$ and $(\mathcal{A}^T \hat{\mathcal{N}}_k^T \mathbf{EW})$ are known. Note that the quantity $(\mathcal{A}^T \hat{\mathcal{N}}_k^T \mathbf{EW}) \mathbf{V}$ has zero mean with a total variance $\sigma^2 \text{tr}\left[(\mathcal{A}^T \hat{\mathcal{N}}_k^T \mathbf{EW})(\mathcal{A}^T \hat{\mathcal{N}}_k^T \mathbf{EW})^T\right]$. We can thus form the test statistic

$$\chi_t = \frac{\rho^2 - \text{tr}\left[\mathbf{S}_{k+1}^H \mathbf{P}_{\mathbf{S}_k} \mathbf{S}_{k+1}\right]}{\sigma^2 \text{tr}\left[(\mathcal{A}^T \hat{\mathcal{N}}_k^T \mathbf{EW})(\mathcal{A}^T \hat{\mathcal{N}}_k^T \mathbf{EW})^T\right]}, \tag{16}$$

which under our hypothesis will be distributed as $\chi^2(N)$. Note that $\chi^2(N)$ will be certainly true when $\mathbf{r}^T$ has a single row. Multiple rows will be correlated, so the test statistic is only approximately $\chi^2(N)$. As we will see in the simulations, this approximation is valid in most of cases. Thus, even though we do not know the correlation between $\mathbf{S}_k$ and $\mathbf{S}_{d-k}$, we can form a simple test statistic to determine whether the Hyperplane algorithm has converged to a correct estimate of $\mathbf{S}_{k+1}$.

4. EXTENDED HYPERCUBE ALGORITHM

Having discussed all the steps, we can now outline our approach to improve the presentation.

1. Given $\tilde{\mathbf{X}} = \tilde{\mathbf{H}}\mathbf{S} + \mathbf{V}$ and initial (or random) $\tilde{\mathbf{H}}_0$.
2. Calculate the whitening matrix $\mathbf{W}$ and form whitened dataset $\mathbf{X} = \mathbf{W}\tilde{\mathbf{X}}$.
3. $\mathbf{X}_1 = \mathbf{X}$, $\hat{\mathbf{S}}_0 = [\,]$ $\mathbf{W}_1 = \mathbf{W}$, $\mathbf{B} = \left(\mathbf{W}\tilde{\mathbf{H}}_0\right)^{-T}$, $\mathbf{B}_1 = \mathbf{B}$.
4. For $k = 1$ to d

   (a) $\boldsymbol{\alpha}_{0,l} = \frac{\mathbf{B}_k \mathbf{e}_k(l)}{\|\mathbf{B}_k \mathbf{e}_k(l)\|}$ and $\mathcal{A}_0 = [\boldsymbol{\alpha}_{0,0} \ \ldots \ \boldsymbol{\alpha}_{0,L-1}]$.

   (b) Call the extended Hyperplane algorithm with input $\mathbf{X}_k$ and $\mathcal{A}_0$. Return $\hat{\mathbf{S}}_k^T$ and $\mathcal{A}$.

   (c) Form the residual vectors $\mathbf{r}^T = \mathcal{A}^T \mathbf{X}_k - \hat{\mathbf{S}}_k^T$.

   (d) Test the convergence by forming the $\chi^2(N)$ test statistic in equation (16) and test against a threshold. If smaller than threshold, continue, else repeat steps 4b and 4c.

   (e) Calculate $\hat{\mathcal{N}}_k^T \mathbf{E}$ as in [10], then $\mathbf{X}_{k+1} = \hat{\mathcal{N}}_k^T \mathbf{EX}$.
5. If desired, we can estimate $\tilde{\mathbf{H}}$: $\hat{\tilde{\mathbf{H}}} = \tilde{\mathbf{X}} \hat{\mathbf{S}}^T (\hat{\mathbf{S}} \hat{\mathbf{S}}^T)^{-1}$.
6. Return $\hat{\mathbf{S}}$ and $\hat{\mathbf{H}}$ (and/or $\hat{\tilde{\mathbf{H}}}$.)

The $\mathbf{B}_k$ matrices are used to exploit *a priori* knowledge of the $\tilde{\mathbf{H}}$ matrix, if any; otherwise a random initial choice can be used. When $\mathcal{A} = [\boldsymbol{\alpha}_0 \ \cdots \ \boldsymbol{\alpha}_{L-1}]$ have misconverged the residual is dramatically larger than for global optima. The complexity of each individual iteration is $\mathcal{O}(kLN)$. Since this is an iterative step, we will define a constant K_{HP} representing the average number of iterations used in the Hyperplane subroutine. Thus the complexity of the extended Hyperplane call in step 4b of the Extended Hypercube algorithm is $\mathcal{O}(K_{\text{HP}} LdN)$. The complexity of the step 4 is $\mathcal{O}((Ld)^2 N)$. Since step 4 loops d times, its total order is $\mathcal{O}(L^2 d^3 N + K_{\text{HP}} L d^2 N)$. The complexity of the Extended Hypercube algorithm is then $\mathcal{O}(M^2 N + L^2 d^3 N + K_{\text{HP}} L d^2 N)$.

5. SIMULATION RESULTS AND CONCLUSIONS

We simulated the proposed method in MATLAB with a varied noise power. In the experiment, we had $d = 2$ BPSK sources transmitting over the randomly generated channels. In the simulation for each user, the multipath delay, and the number of multipath components were randomly chosen to be uniformly distributed within $[0 \quad 3T]$, and $[1 \quad 10]$, respectively. We also used $M = 8$ antennas and $N = 100$ data samples. The symbols in $\mathbf{S}$ were chosen with equal probability. We scaled the received power of both signals to be equal. The real and imaginary components of $\mathbf{V}$ where each drawn from a zero-mean Gaussian distribution with variance $\sigma^2/2$ for a total noise power of σ^2. The signal-to-noise ratio defined as the average SNR per signal per antenna SNR $= \frac{\|\mathbf{HS}\|_2}{dMN\sigma^2}$. After each run, cumulative bit errors for all d signals were calculated.

In order to have a comparison for our BER curves, we also processed the data with known pseudo-inverse processing (ZF), or zero-forcing defined in [5]. One of the well-known disadvantages of zero-forcing equalizer is that it enhance the noise. Because this causes very high bit error rates as we will see in the experiment, we also employed a MSE (minimum squared error) equalizer with the known $\mathbf{H}$ and noise variance, σ^2 [5].

Figure 2 displays the BER curves for the both the extended hypercube algorithm and MCD-ILSP algorithm as a function of input SNR for $d = 2$ and $M = 8$ with a BPSK alphabet. The curve for ZF equalizer and MSE equalizer with known $\tilde{\mathbf{H}}$ is also plotted. Except the first user, MSE did not generate any errors for this experiment. Note that our approach outperforms the MCD-ILSP. Figure 3 displays the average Kilo-flops per block of $M \times N$ data. The Modified Hypercube algorithm requires markedly fewer floating point operations than the MCD-ILSP algorithm because of the elimination of costly MCD processing.

REFERENCES

[1] L. K. Hansen and G. Xu. "A hyperplane-based algorithm for the digital co-channel communications problem," *IEEE Trans. Info. Theory*, 43(5):1536–1548. September 1997.

[2] L. K. Hansen and G. Xu. "Geometric properties of the blind digital co-channel communications problem," In *Proc. ICASSP'96*, pages 1085–1088, Atlanta, GA, May 1996.

[3] L. K. Hansen and G. Xu. "A fast algorithm for the blind separation of digital co-channel signals," *Proc. 30th Asilomar Conference on Signals, Systems, and Comp.*, Pacific Grove, CA, November 1996.

[4] R. T. Behrens and L. L. Scharf. "Signal processing applications of oblique projection operators," *IEEE Transactions on Signal Processing*, 42(6):1413–1424, June 1994.

[5] M. Torlak and G. Xu. "Maximum likelihood detection of co-channel communication signals exploiting the spatio-temporal diversity," In *Proc. of the 1996 30th Asilomar Conference on Signals, Systems & Computers*, pp. 728-732, Nov. 1996.

[6] S. Talwar, M. Viberg, and A. Paulraj. "Blind estimation of multiple co-channel digital signals using an antenna array," *IEEE Signal Processing Letters*, 1(2):29–31, Feb. 1994.

[7] S. Talwar, M. Viberg, and A. Paulraj. "Blind separation of synchronous co-channel digital signals using an antenna array—part i: Algorithms," *IEEE Transactions on Signal Processing*, 44(5):1184–1197, May 1996.

[8] A. J. van der Veen, S. Talwar, and A. Paulraj, "A subspace approach to blind space-time signal processing for wireless communication systems," *IEEE Trans. Signal Processing*, vol. 45, pp. 173-189, Jan. 1997.

[9] H. Liu and G. Xu, "Closed-form blind symbol estimation in digital communications," *IEEE Trans. Signal Processing*, vol. 43, pp. 2714-2723, Nov. 1995.

[10] M. Torlak, L. K. Hansen, and G. Xu. "A geometric approach to blind source seperation for digital wireless applications," submitted to Signal Processing, 1997.

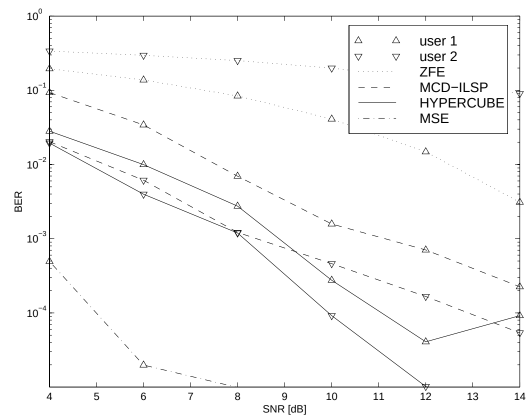

Figure 2. Bit error rates for $M = 8$ antennas, $d = 2$ BPSK signals, and $N = 100$ data samples as a function of input SNR.

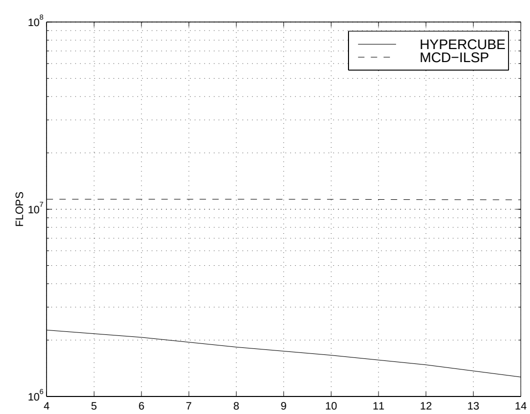

Figure 3. Average number of Kilo-flops per data block for $M = 8$ antennas, $d = 2$ BPSK signals, and $N = 100$ data samples as a function of input SNR.