Analytical Method for Blind Binary Signal Separation

Alle-Jan van der Veen

Abstract—**The blind separation of multiple co-channel binary digital signals using an antenna array involves finding a factorization of a data matrix X into $X = AS$, where all entries of S are $+1$ or -1. It is shown that this problem can be solved exactly and non-iteratively, via a certain generalized eigenvalue decomposition. As indicated by simulations, the algorithm is robust in the presence of noise. An interesting implication is that certain cluster segmentation problems can be solved using eigenvalue techniques.**

I. INTRODUCTION

A core problem in the area of blind signal separation/equalization is the following. Consider d independent sources, transmitting binary symbols $\{+1, -1\}$ at equal rates in a wireless scenario. The signals are received by a central antenna array, consisting of M antennas. Assuming synchronized sources, equal transmission delays, negligible delay spread, and sampling at the bit rate, each antenna receives a linear combination of the transmitted symbol sequences $\mathbf{s}_i = [s_i(T), s_i(2T), \cdots, s_i(NT)]$ $(i = 1, \cdots, d)$, leading to the well-known data model

$$X = AS = \mathbf{a}_1\mathbf{s}_1 + \cdots + \mathbf{a}_d\mathbf{s}_d\,, \qquad S_{ij} \in \{+1, -1\}\,. \qquad (1)$$

Here, $X = [x_{ik}]$ with $x_{ik} = x_i(kT)$ $(i = 1, \cdots, M;\ k = 1, \cdots, N)$ is a complex matrix containing the received data during N symbol periods. $A \in \mathbb{C}^{M\times d}$ is the array response matrix, $S \in \{\pm 1\}^{d\times N}$ contains the transmitted bits. In the blind signal separation scenario, both A and S are unknown, and the objective is, given X, to find the factorization $X = AS$ such that S belongs to the binary alphabet. Alternatively, we try to find a weight matrix W of full row rank d such that $S = W^*X$. Uniqueness of this factorization is important, and was established in [1]: if A is full rank and the columns of S exhaust all 2^{d-1} distinct (up to a sign) possibilities, then this is sufficient for the factorization to be unique up to trivial permutations and scalings by ±1 of the rows of S and columns of A. Hence, once any such factorization of X is found, S contains the binary signals that were originally transmitted, or their negative, but not some ghost signal.

This scenario by itself is perhaps naive, but it is the core problem in more realistic blind (FIR-MIMO) scenarios [2], where long delay multipath is allowed, and sources are not synchronized and are modulated by arbitrary pulse shape functions. This problem is separable into a blind multi-user equalization stage and a separation problem which is precisely of the form (1). Several other binary modulation schemes such as MSK or biphase (Manchester) codes are easily converted to fit the model as well [3].

One of the first papers to consider this problem appeared in full as [1]. In that paper, arbitrary finite alphabets are considered although only BPSK was tested extensively. The problem was cast into an optimization problem, min $\|X - AS\|_F$, subject to $S_{ij} \in \{\pm 1\}$. Two fixed-point iteration algorithms were proposed, one called ILSE which is based on clever enumeration of candidate matrices S, and a second called ILSP which uses alternating projections. The main concern with these algorithms is their initialization and lack of global convergence. Depending on the initialization, the algorithms can converge to a local minimum, and restarts are needed if not all independent signals are found. If successful, ILSE is a conditional maximum likelihood estimator. ILSP is suboptimal but much cheaper to compute, and can be used to initialize ILSE. Later, an unconditional maximum likelihood technique for the estimation of A was derived [4], here called the UML. The algorithm consists of a fixed-point iteration as well, and also requires an accurate initialization. Its performance is similar to ILSE.

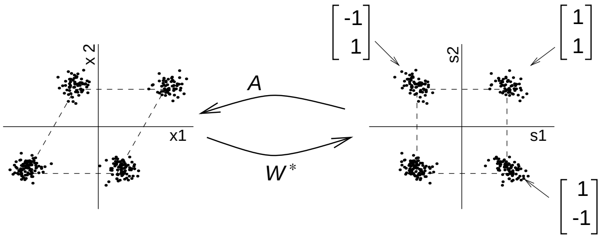

Fig. 1. A maps a source constellation onto a transformed constellation.

Several people noted that the problem (1) is essentially a clustering problem, as illustrated in figure 1 for the case of $d = 2$, $M = 2$. In the absence of noise, X can contain only 2^d distinct vectors. To estimate A, it suffices to determine a suitable assignment of these vectors (or cluster centers) to constellation vectors, i.e., the columns of S, taking symmetry into account. A non-iterative combinatorial algorithm based on such ideas, called SD, was presented in [5]. With noise, however, the segmentation and hence the estimation of the cluster centers is difficult and limits the performance of the algorithm.

The main point of the present paper is the observation that there exists a non-iterative algorithm that finds the factorization (1) exactly and algebraically, by reducing it to a joint diagonalization problem, which is a (generalized) eigenvalue problem. The algorithm is robust in the presence of noise, as demonstrated by simulations. Apart from certain details, it is in fact an almost trivial specialization of a recently developed "analytical constant modulus algorithm" (ACMA) [6], which solves the factorization $X = AS, |S_{ij}| = 1$ for complex matrices A and S, as it is straightforward to restrict S to be real as well. Nonetheless, the problem is sufficiently important to warrant a separate mentioning, especially since it implies the interesting observation that some clustering problems can be solved by eigenvalue techniques.

It should be noted that the blind binary source separation problem fits into the more general class of source separation based on observed linear instantaneous mixtures. Of particular interest here are algorithms that use the statistical independence of the souces, which has led to "independent component analysis" and related high-order statistics techniques, viz. among others [7–9].

Manuscript submitted to IEEE Trans. Signal Processing, Feb. 1996, revised Nov. 1996. A.J. van der Veen is with the Department of Electrical Engineering/DIMES, Delft University of Technology, Delft, The Netherlands.

Rather intriguingly, these methods are also based on joint diagonalizations, in this case of cumulant matrices.

II. REAL ACMA

The blind binary souce separation problem is to find a factorization $X = AS$ where $S_{ij} \in \{\pm 1\}$. Since S is real-valued, it is advantageous to write

$$\begin{array}{rcl} X = AS & \Leftrightarrow & \begin{bmatrix} \text{real}(X) \\ \text{imag}(X) \end{bmatrix} = \begin{bmatrix} \text{real}(A) \\ \text{imag}(A) \end{bmatrix} S \\ & \Leftrightarrow & X_R = A_R S\,, \end{array}$$

with obvious definitions of $X_R \in \mathbb{R}^{2M\times N}$ and $A_R \in \mathbb{R}^{2M\times d}$. This forces S to be real, and at the same time, A_R is usually much better conditioned than A. Equivalently, the problem is to find all independent vectors $\mathbf{w}_R \in \mathbb{R}^{2M}$ such that $\mathbf{w}_R^T X_R = \mathbf{s}$ has entries $(\mathbf{s})_k \in \{\pm 1\}$.

Without noise, X_R is rank-deficient, which leads to ambiguities in $\mathbf{w}_R$. To avoid this, the first step in the algorithm is to reduce the dimension of $\mathbf{w}_R$ from $2M$ to d. Thus let $X_R = \hat{U}\hat{\Sigma}\hat{V}$ be an “economy-size” SVD for X_R, where $\hat{U} : 2M\times d$ has orthogonal columns, $\hat{\Sigma} : d\times d$ is a diagonal matrix containing the non-zero singular values of X_R, and $\hat{V} : d\times N$ has orthogonal rows, which form a basis for the row span of X_R. Thus, the problem is equivalent to finding all independent vectors $\mathbf{w} \in \mathbb{R}^d$ such that

$$\mathbf{w}^T\hat{V} = \mathbf{s}\,, \qquad (\mathbf{s})_k \in \{\pm 1\}\,. \tag{2}$$

The alphabet condition is written as

$$s \in \{\pm 1\} \quad \Leftrightarrow \quad (s-1)(s+1) = 0 \quad \Leftrightarrow \quad s^2 = 1 \tag{3}$$

(with possible extensions to other constellations). Denoting the k-th column of $\hat{V}$ by $\mathbf{v}_k$, substitution of (2) into (3) leads to

$$\mathbf{w}^T\mathbf{v}_k\mathbf{v}_k^T\mathbf{w} = 1\,, \qquad k = 1,\cdots,N\,. \tag{4}$$

Similar equations arose in the solution of the constant modulus problem [6], where we had $|s|^2 = 1$ rather than $s^2 = 1$. As in [6], the conditions can be rewritten in a linear form by using Kronecker products: $[\mathbf{v}_k\otimes\mathbf{v}_k]^T[\mathbf{w}\otimes\mathbf{w}] = 1$, but in the present case the Kronecker product vectors have duplicate entries which can (and have to) be removed. Thus define, for a $d\times d$ real symmetric matrix $Y = [y_{ij}]$, a scaled stacking of the lower triangular part of the columns:

$$\begin{array}{rcl} \text{rvec}(Y) & := & [y_{11} \;\; \frac{y_{21}}{\sqrt{2}} \cdots \frac{y_{d1}}{\sqrt{2}} \;\; y_{22} \;\; \frac{y_{32}}{\sqrt{2}} \cdots \\ & & \cdots \frac{y_{d,d-1}}{\sqrt{2}} \;\; y_{dd}]^T \in \mathbb{R}^{d(d+1)/2}. \end{array}$$

This allows to write (4) as

$$[\text{rvec}(\mathbf{v}_k\mathbf{v}_k^T)]^T\text{rvec}(\mathbf{w}\mathbf{w}^T) = 1\,, \qquad k = 1,\cdots,N. \tag{5}$$

After collecting all rows $\text{rvec}(\mathbf{v}_k\mathbf{v}_k^T)^T$ into a matrix P, the problem reduces to finding all independent vectors $\mathbf{y}$ satisfying

$$P\mathbf{y} = \mathbf{1}\,, \qquad \mathbf{y} = \text{rvec}(\mathbf{w}\mathbf{w}^T)\,,$$

where $\mathbf{1} = [1\cdots 1]^T$. Hence, we have replaced the quadratic equations (4) by a linear system $P\mathbf{y} = \mathbf{1}$, subject to a quadratic constraint which imposes a certain structure on $\mathbf{y}$.

The remaining steps are identical to the procedure in [6], and only summarized here. First transform the linear system to an equivalent system $\hat{P}\mathbf{y} = \mathbf{0}$. Let Q be an orthogonal (Householder) transformation such that $Q\mathbf{1} = [\sqrt{N}\;\; \mathbf{0}_{N-1}]^T$, and let $\hat{P}$ be the last $N-1$ rows of QP (i.e., the first row is removed), then, up to a scaling, solving $P\mathbf{y} = \mathbf{1}$ is equivalent to solving

$$\hat{P}\mathbf{y} = \mathbf{0}\,, \qquad \mathbf{y} \neq \mathbf{0}\,. \tag{6}$$

The general solution of (6) has the form

$$\mathbf{y} = \alpha_1\mathbf{y}_1 + \cdots + \alpha_\delta\mathbf{y}_\delta\,, \qquad (\alpha_i \in \mathbb{R}\,, \textstyle\sum|\alpha_i| \neq 0)\,,$$

where $\{\mathbf{y}_i\}$ is a basis of the null space of $\hat{P}$, and δ is defined to be the dimension of this space. In the presence of noise, both the basis and its dimension are estimated by an SVD of $\hat{P}$. Since we know that there are d linearly independent solutions $\mathbf{w}$, and since linearly independent vectors $\mathbf{w}$ lead to linearly independent vectors $\mathbf{y} = \text{rvec}(\mathbf{w}\mathbf{w}^T)$, there are at least d independent solutions to (6): $\delta \geq d$. On the other hand, if sufficient conditions are imposed by $\hat{P}$, then the dimension of the null space of $\hat{P}$ will not be larger than d. In particular, if $N \gg 2^{d-1}$ then we expect $\delta = d$ with high probability, as is argued later in this section. For this property to hold, it is essential to have used rvec in (5), or else $\hat{P}$ has duplicate columns and the dimension of the kernel will be too large. (This is precisely the reason why BPSK and MSK signals were noted exceptions in the ACMA algorithm [6].)

At this point, we have obtained a basis of solutions $\{\mathbf{y}_i\}$, but since the basis is arbitrary, each $\mathbf{y}_i$ is probably not of the form $\text{rvec}(\mathbf{w}\mathbf{w}^T)$. To force the structural property $\mathbf{y} = \text{rvec}(\mathbf{w}\mathbf{w}^T)$, write $Y_i = \text{rvec}^{-1}(\mathbf{y}_i)$, which gives

$$\mathbf{w}\mathbf{w}^T = \alpha_1 Y_1 + \cdots + \alpha_\delta Y_\delta\,. \tag{7}$$

We have to find all d parameter vectors $[\alpha_1\cdots\alpha_\delta]$ such that the resulting linear combination of the matrices $\{Y_i\}$ is of rank 1 and symmetric, in which case it can be factored as $\mathbf{w}\mathbf{w}^T$. As discussed in [6], this is essentially a generalized eigenvalue problem. For $d = 2$, it is a 2×2 matrix pencil problem with a closed-form solution, for $d > 2$ and with noise, there is in general no exact such decomposition, but we can try to make the linear combination as close to rank 1 as possible. The symmetry property is automatic because for real-valued α-vectors, $Y_1,\cdots,Y_\delta$ are symmetric by construction.

A technique for computing all α-vectors is detailed in [6], for the general complex case, but a specialization to the present real case is immediate. For each α-vector, the corresponding $\mathbf{w}$ follows from (7).

Since d is typically small, the computational effort required by the diagonalization step is negligible in comparison with computing the SVD of $\hat{P}$. This brings the overall computational complexity of the algorithm to around $\mathcal{O}((M^2 + d^4)N)$. There are interesting possibilities for updating the null space estimate of $\hat{P}$ using subspace trackers.

Rank of **P**

From the above, it is clear that the dimension δ of the null space of $\hat{P}$ plays an important role. Using similar arguments as in [6], one can show that this dimension is independent of A

as long as A has full rank. Furthermore, $\delta = d$ iff $\hat{P}$ has rank $\frac{1}{2}d(d+1) - d = \frac{1}{2}d(d-1)$, i.e., iff P has rank $\frac{1}{2}d(d-1)+1$. For this, it is *sufficient* that S contains all 2^{d-1} essentially distinct constellation vectors ("essentially" meaning beyond a factor ±1). A conjecture that any subset of $\frac{1}{2}d(d-1)+1$ constellation vectors out of these 2^{d-1} would already be sufficient turns out not to be true because of linear dependencies. E.g., for $d = 5$ it was found that some subsets of 11 constellation vectors only give $\text{rank}(P) = 10$ rather than 11, and that at least 13 constellation vectors are needed to guarantee $\text{rank}(P) = 11$. An experiment for $d = 10$ shows that 380 vectors out of 512 is still not sufficient in all cases.

For $N \gg 2^{d-1}$ and assuming equal probabilities on the occurrence of any constellation vector, a lower bound on the probability p that S contains all 2^{d-1} constellation vectors is given in [1, §A.2] as $p \geq 1 - 2^{d-1}(1-2^{-(d-1)})^N$. This also gives a lower bound on the probability that $\delta = d$, be it rather pessimistic because there are many subsets that are sufficient as well.

III. SIMULATIONS

To test the algorithm, the following scenario is considered. We have $d = 4$ equipowered sources, with directions-of-arrival $-3°, 0°, 4°, 8°$ with respect to the array broadside. The sensor array is a uniform linear array consisting of $M = 6$ omnidirectional antennas spaced at $\lambda/2$. An arbitrary initial phase of each signal is incorporated in A. The condition number of the complex-valued A is about 300, so that the problem looks quite ill-conditioned. However, since we try to recover *real* signals, the true conditioning of the problem is determined by $A_R = [\text{real}(A)^T \;\; \text{imag}(A)^T]^T$. Unlike the complex case, the conditioning of A_R is very much dependent on the initial (random) phases of the signals: it can be as low as 3, or as high as 200. The median of the distribution was found to be 9.5, with a standard deviation of 8.4, so that the problem is medium-conditioned in the majority of cases.

The signal-to-noise ratio (SNR) is defined with respect to signal 1. We took $N = 100$ snapshots and a total of 2000–8000 Monte Carlo runs. The bit-error rate (BER) is the total BER over all d signals. The percentage of cases where not all d signals are recovered is defined as the recovery failure rate (RFR) and is listed separately. These cases are omitted from the BER statistics.

RACMA is compared to ILSP [1], ILSE [1], UML [4] and SD [5]. Of the latter algorithms, only SD does not require an explicit initial guess for A. ILSP is initialized with $A_0 = I_{M\times d}$. ILSE and UML require a more accurate initialization, and we use the result of the ILSP algorithm for that. It is also possible to use the result of RACMA to initialize ILSP, ILSE and UML, which can improve results because RACMA is not statistically optimal (in fact it is biased). UML requires an estimate of the noise power.

We first test the case where the random initial signal phase is selected once and held fixed during the simulation. Figure $2(a)$ shows the resulting bit error rates as a function of SNR; figure $2(d)$ shows the corresponding recovery failure rates. In this simulation, $\text{cond}(A_R) = 5.3$. In accordance to theory [1], it is seen that the performance of ILSP is limited in comparison to ILSE. For high SNRs, the initialization by $A_0 = I$ is not effective to recover all signals, and very often, the algorithm gets stuck in a local minimum. The ILSE algorithm, when initialized by ILSP ("ILSP+E"), improves on ILSP, but is not able to recover from the local minima at high SNRs either. For reference, the figure also lists the performance of ILSP and ILSE when initialized with the true A ("ILSP(A)" and "ILSE(A)"). Note that ILSE(A) is expected to converge almost surely to the optimal (conditional) ML estimator. It is seen that RACMA has a performance close to ILSP(A). If the estimate of A produced by RACMA is used for initialization of ILSE ("RACMA+ILSE"), we come close to ILSE(A), except for very low SNR. It is natural that the performance of RACMA is limited at low SNR because of the inherent squaring of the data in the construction of P, which increases the effective noise power.

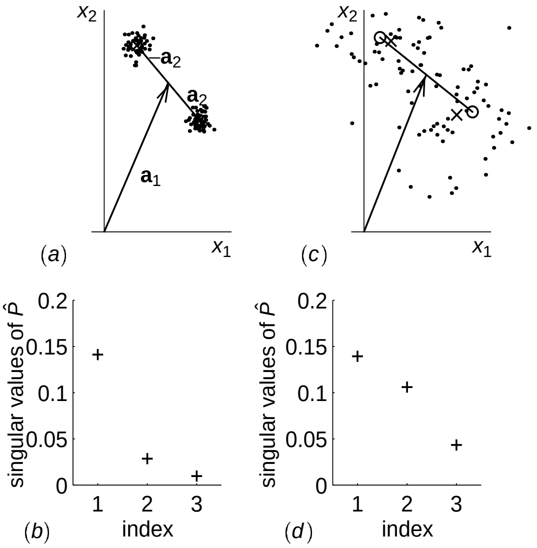

Fig. 3. Example of cluster segmentation using RACMA. (a),(b) low noise, (c),(d) high noise.

Figure $2(b)$,(c) shows the performance for the case where the initial phases of the signals are selected randomly for every Monte-Carlo run. Figure $2(e)$,(f) shows the corresponding RFR. It is seen that SD is effective at high SNRs, but as a noniterative combinatorial method, it is easily confused at low SNRs where it fails to recover all sources in a majority of cases. The performance of UML is virtually the same as that of ILSE, except that its capture performance is slightly better at low SNRs. RACMA has a BER performance similar to ILSP(A), although for low SNR it is less successful in recovering all d signals. By itself, it is suboptimal, but provides a good initial point for ILSE or UML. It does not reach the performance of ILSE(A) any more, because the low-SNR discrepancy observed in figure $2(a)$ is now spread out over all SNRs as A assumes a range of condition numbers.

Finally, $2(c)$ also shows the effect of overestimating d in RACMA: "RACMA(6)" lists the case where $d = 6$ is used in the SVD of X and the construction of P. The BER performance is almost the same as RACMA, but it becomes 3 dB less effective in capturing all signals.

IV. APPLICATION TO CLUSTERING

As remarked in the introduction, for discrete signals the $X = AS$ factorization problem is essentially a cluster segmentation

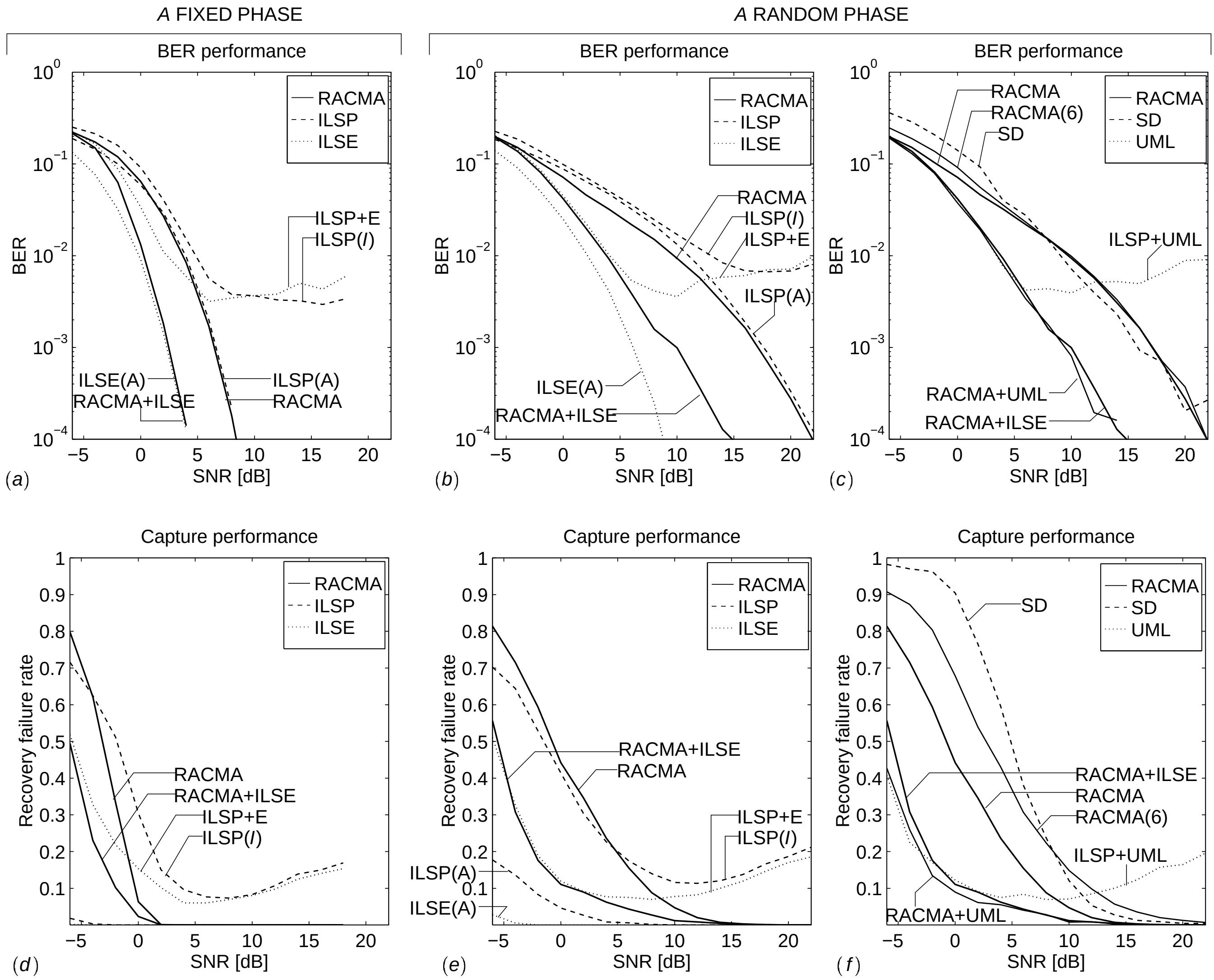

Fig. 2. (a) BER performance for fixed A; (b),(c) BER performance for A with random signal phase; (d)–(f) corresponding failure rate (cases where not all signals are recovered).

problem. This implies that certain cluster segmentation problems can be solved using eigenvalue techniques, which might provide an interesting alternative to the usual iterative algorithms. Although the present algorithm expects the cluster centers to lie on the vertices of a parallelepiped, some generalizations to other configurations are possible.

To illustrate this, consider figure 3, which shows two clusters arbitrarily located in a two-dimensional space. This is a special case of our data model: we have $M = 2$ real-valued sensors, and the received data can be written as

$$X = [\mathbf{a}_1 \quad \mathbf{a}_2]\begin{bmatrix} 1 & 1 & \cdots \\ \pm 1 & \pm 1 & \cdots \end{bmatrix} + E \qquad (8)$$

where E signifies the additive noise. Hence we can set $d = 2$ “sources”, although the first source is in fact constant $(+1)$. Since we receive 2^{d-1} essentially different constellation vectors, this is sufficient for the factorization to be unique. Figure $3(b)$ shows the singular values of $\hat{P}$, which is an $N-1$ by 3 matrix. Clearly, there are $\delta = 2$ small singular values (they would be zero in the noise-free case). With more noise, the small singular values of $\hat{P}$ are increased (figure $3(d)$), and it becomes hard to detect from the singular values that there are really two clusters rather than one. Nonetheless, if it is decided that $\delta = 2$, then the resulting cluster centers are still close to the true centers (indicated by a ‘×’). The singular values could be used for a hypothesis test to distinguish between the presence of one versus two clusters.

A similar example is a case were we have two clusters but only one sensor ($M = 1$), e.g., if we receive a single binary source, distorted by an arbitrary offset c, scaling k, and additive zero mean white noise:

$$x_i = k \cdot s_i + c + e_i\,, \qquad i = 1, \cdots, N.$$

To fit this to an $X = AS$ model where $M \geq d$, we can consider an augmented data matrix,

$$\begin{aligned} X_e &= \begin{bmatrix} 1 & 1 & \cdots & 1 \\ x_1 & x_2 & \cdots & x_N \end{bmatrix} \\ &= \begin{bmatrix} 1 & 0 \\ c & k \end{bmatrix}\begin{bmatrix} 1 & 1 \cdots 1 \\ s_1 & s_2 \cdots s_N \end{bmatrix} + \begin{bmatrix} 0 & 0 \cdots 0 \\ e_1 & e_2 \cdots e_N \end{bmatrix}. \end{aligned}$$

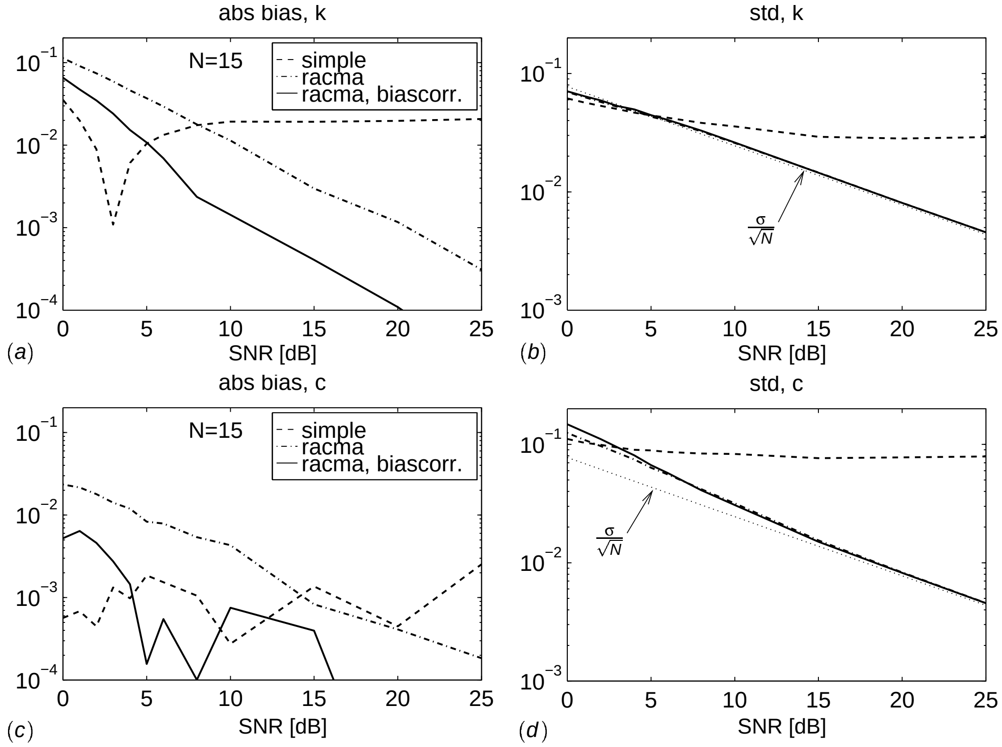

Fig. 4. Parameter estimation of a model $x_i = ks_i + c + e_i$.

It is instructive to partially work out the expressions for the estimation of $\mathbf{w}$ (and hence of k and c, since $\mathbf{w} = [-ck^{-1} \;\; k^{-1}]^T$). In the model $P\mathbf{y} = \mathbf{1}$, we here define for convenience $\mathbf{y} = [w_1^2 \;\; w_1 w_2 \;\; w_2^2]^T$ and the rows of P by $[1 \;\; 2x_k \;\; x_k^2]$. The transformation by Q maps the condition $P\mathbf{y} = \mathbf{1}$ to $\left(\mathbf{1}^T P\mathbf{y} = n \,,\; \hat{P}\mathbf{y} = \mathbf{0}\right)$. Using the simple structure of Q, we then find

$$\begin{aligned} R \;&:=\; \hat{P}^T\hat{P} = P^T(I - \tfrac{1}{N}\mathbf{1}\cdot\mathbf{1}^T)P \\ &\sim \begin{bmatrix} 0 & 0 & 0 \\ 0 & 4(\Sigma_2 - \frac{1}{N}\Sigma_1^2) & 2(\Sigma_3 - \frac{1}{N}\Sigma_1\Sigma_2) \\ 0 & 2(\Sigma_3 - \frac{1}{N}\Sigma_1\Sigma_2) & \Sigma_4 - \frac{1}{N}\Sigma_2^2 \end{bmatrix} + \\ &\quad + \begin{bmatrix} 0 & 0 & 0 \\ 0 & 4E_2 & \frac{4}{N}\Sigma_1 E_2 \\ 0 & \frac{4}{N}\Sigma_1 E_2 & E_4 - \frac{5}{N}E_2^2 + \frac{4}{N}\Sigma_2 E_2 \end{bmatrix} \end{aligned}$$

where $\Sigma_n := \sum_1^N x_i^n$, $E_n := \sum_1^N e_i^n$, and '$\sim$' denotes equality in mean. In the absence of noise, R has two zero eigenvalues, with eigenvectors $\mathbf{y}_1 = [1 \;\; 0 \;\; 0]^T$ and $\mathbf{y}_2 = [0 \;\; a \;\; b]^T$, say. The joint diagonalization step collapses: it directly follows that $\mathbf{w} = \alpha[a \;\; b]^T$, for some scaling α which can be estimated from the condition $\mathbf{1}^T P\mathbf{y} = n$. This example shows that the algorithm is in fact a square-root method based on 4-th order moments of the data.

With noise, it is clear that the nonzero block $\underline{R}$ of R is biased, an effect which so far has not been taken into account. A correction is possible if the moments of the noise are known. E.g., for Gaussian noise with variance σ^2, we have $E_2 \sim N\sigma^2$, $E_4 \sim 3N\sigma^4$, so that the the bias term is asymptotically given by

$$\underline{R}_E = \sigma^2 \begin{bmatrix} 4N & 4\Sigma_1 \\ 4\Sigma_1 & 4\Sigma_2 - 2N\sigma^2 \end{bmatrix}$$

If we neglect the term $-2N\sigma^4$, then the noise variance can be readily estimated as the (smallest) eigenvalue of the pencil $(\underline{R}, \underline{R}_E)$, since this is the value that will make $\underline{R} - \lambda\underline{R}_E$ singular again.

Some simulation results are given in figure 4. In this simulation, $k = 0.3$, $c = 1.2$, $N = 15$, and the results are averaged over 5000 Monte-Carlo runs. It is observed that the bias correction has little influence on the variance of the estimates, which quickly converge to $\frac{1}{N}\sigma^2$: the variance of the ML estimators of k and c for known $\mathbf{s}$ and sufficiently small $\sum_1^N s_i$. The 'simple estimator' is $\hat{c} = \frac{1}{N}\sum_1^N x_i$, $\hat{k} = \frac{1}{N}\sum |x_i - \hat{c}|$, which assumes that there is an equal number of $+1$ and -1 in the data batch. For small N or otherwise asymmetric sources, this estimator is not very good.

References

[1] S. Talwar, M. Viberg, and A. Paulraj, "Blind estimation of synchronous co-channel digital signals using an antenna array. Part I: Algorithms," *IEEE Tr. Signal Proc.*, vol. 44, pp. 1184–1197, May 1996.

[2] 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, pp. 99–102, May 1995.

[3] A.J. van der Veen and A. Paulraj, "Singular value analysis of space-time equalization in the GSM mobile system," in *Proc. IEEE ICASSP*, pp. 1073–1076, May 1996.

[4] B. Halder, B.C. Ng, A. Paulraj, and T. Kailath, "Unconditional maximum likelihood approach for blind estimation of digital signals," in *Proc. IEEE ICASSP*, vol. 2, pp. 1081–1084, May 1996.

[5] K. Anand, G. Mathew, and V.U. Reddy, "Blind separation of multiple co-channel BPSK signals arriving at an antenna array," *IEEE Signal Proc. Letters*, vol. 2, pp. 176–178, Sept. 1995.

[6] A.J. van der Veen and A. Paulraj, "An analytical constant modulus algorithm," *IEEE Tr. Signal Proc.*, vol. 44, pp. 1136–1155, May 1996.

[7] L. Tong, Y. Inouye, and R.-W. Liu, "Waveform-preserving blind estimation of multiple independent sources," *IEEE Tr. Signal Proc.*, vol. 41, pp. 2461–2470, July 1993.

[8] J.F. Cardoso and A. Souloumiac, "Blind beamforming for non-Gaussian signals," *IEE Proceedings F (Radar and Signal Processing)*, vol. 140, pp. 362–370, Dec. 1993.

[9] P. Comon, "Independent component analysis, a new concept?," *Signal Processing*, vol. 36, pp. 287–314, Apr. 1994.