ASYMPTOTIC BEHAVIOR OF ACMA

Alle-Jan van der Veen

Delft University of Technology, Dept. Electrical Engineering/DIMES, 2628 CD Delft, The Netherlands

The algebraic constant modulus algorithm (ACMA) is a non-iterative blind source separation algorithm. It computes jointly beamforming vectors for all constant modulus sources as the solution of a joint diagonalization problem. In this paper we analyze its asymptotic properties and show that (unlike the iterative CMA) it converges to the Wiener solution in samples or SNR. We also sketch its connection to the related JADE algorithm.

1. INTRODUCTION

Constant modulus algorithms (CMAs) enjoy widespread popularity as methods for blind source separation and equalization of communication signals. Iterative CMAs are straightforward to implement and computationally of modest complexity. For the purpose of blind source separation, a complication is that only a single weight vector is found at a time. To recover the other signals successively, we have to remove the previous solutions from the data, which relies on independence and requires many samples [1]. Moreover, converge can be slow, with unpredictable convergence speed depending upon initialization.

The ACMA was introduced in [2] as an algebraic method for computing the complete collection of beamformers in one shot, as the solution of a generalized eigenvalue problem. Only a small batch of samples is needed, and the number of CM signals can be detected as well. Convergence is not an issue.

In this paper, we analyze the asymptotic properties of ACMA and show that, with Gaussian noise, it converges to the Wiener solution in samples or SNR. We also discuss its connection to the JADE algorithm.

2. DATA MODEL

Consider d independent sources, transmitting signals $s_i(t)$ with constant modulus waveforms ($|s_i(t)| = 1$) in a wireless scenario. The signals are received by an array of m antennas. We stack the antenna outputs $x_i(t)$ into vectors $\mathbf{x}(t)$ and collect n samples in a matrix $X : m\times n$. Assuming that the sources are sufficiently narrow-band in comparison to the delay spread of the multipath channel, this leads to the well-known data model

$$\mathbf{x}_k = A\mathbf{s}_k \quad \Rightarrow \quad X = AS. \tag{1}$$

$A = [\mathbf{a}_1 \ \cdots \mathbf{a}_d] \in \mathbb{C}^{m\times d}$ is the array response matrix. The rows of $S \in \mathbb{C}^{d\times n}$ contain the samples of the source signals. Both A and S are unknown, and the objective is, given X, to find the factorization $X = AS$ such that $|S_{ij}| = 1$. Alternatively, we try to find a beamforming matrix $W = [\mathbf{w}_1, \cdots, \mathbf{w}_d] \in \mathbb{C}^{m\times d}$ of full row rank d such that $S = W^*X$. For this we need A to have full rank d, and $m \geq d$.

In the presence of additive noise, we write $\tilde{\mathbf{x}}_k = A\mathbf{s}_k + \mathbf{n}_k$, or

$$\tilde{X} = AS + N. \tag{2}$$

The noise is assumed to be additive white, zero mean, circularly symmetric, with finite covariance $\mathrm{E}(\mathbf{n}\mathbf{n}^*)$ and fourth-order moments, and independent from the sources.

Notation Overbar ($\overline{\ \ }$) denotes complex conjugation, ^{T} is the matrix transpose, * the matrix complex conjugate transpose, † the matrix pseudo-inverse (Moore-Penrose inverse). $\mathbf{0}$ and $\mathbf{1}$ are vectors for which all entries are equal to 0 and 1, respectively. We use the tilde ($\tilde{\ }$) to denote variables derived from noisy data.

$\mathrm{vec}(A)$ is a stacking of the columns of a matrix A into a vector. $\otimes$ is the Kronecker product, $\circ$ is the Khatri-Rao product, which is a column-wise Kronecker product:

$$A \circ B = [\mathbf{a}_1 \otimes \mathbf{b}_1 \quad \mathbf{a}_2 \otimes \mathbf{b}_2 \quad \cdots].$$

3. DERIVATION OF THE ACMA

We summarize the derivation of the basic ACMA algorithm for the noiseless case. The objective is to find all independent beamforming vectors $\mathbf{w}$ that reconstruct a signal with a constant modulus, i.e.,

$$\mathbf{w}^*X = \mathbf{s}, \qquad \text{such that} \quad |s_k|^2 = 1 \quad (k = 1, \cdots, n).$$

Let $\mathbf{x}_k$ be the k-th column of X. By substitution, we find

$$\mathbf{w}^*(\mathbf{x}_k\mathbf{x}_k^*)\mathbf{w} = 1, \qquad k = 1, \cdots, n. \tag{3}$$

Note that $\mathbf{w}^*(\mathbf{x}_k\mathbf{x}_k^*)\mathbf{w} = (\bar{\mathbf{x}}_k \otimes \mathbf{x}_k)^*(\bar{\mathbf{w}} \otimes \mathbf{w})$. Thus let $P := [\bar{X} \circ X]^*$ (size $n\times d^2$). Then (3) is equivalent to finding all $\mathbf{w}$ that satisfy

$$P\mathbf{y} = \mathbf{1}, \qquad \mathbf{y} = \bar{\mathbf{w}} \otimes \mathbf{w}.$$

This is a linear system of equations, subject to a quadratic constraint. The linear system is overdetermined once $n \geq d^2$, and we will assume that this is the case.

In general outline, the ACMA technique solves this problem by the following steps:

1. *First solve the linear system* $P\mathbf{y} = \mathbf{1}$. Note that there are at least d independent solutions to the linear system, namely $\bar{\mathbf{w}}_i \otimes \mathbf{w}_i$ ($i = 1, \cdots, d$). But also a linear combination of these solutions

   $$\mathbf{y} = \lambda_1(\bar{\mathbf{w}}_1 \otimes \mathbf{w}_1) + \cdots + \lambda_d(\bar{\mathbf{w}}_d \otimes \mathbf{w}_d)$$

   (scaled such that $\sum \lambda_i = 1$) will solve $P\mathbf{y} = \mathbf{1}$.

   To find a basis $\{\mathbf{y}_i\}$ of solutions, let Q be any unitary matrix such that $Q\mathbf{1} = \sqrt{n}\begin{bmatrix}1\\ \mathbf{0}\end{bmatrix}$. Apply Q to $[\mathbf{1} \ \ P]$:

   $$Q[\mathbf{1} \quad P] =: \sqrt{n}\begin{bmatrix} 1 & \mathbf{p}^* \\ \mathbf{0} & G \end{bmatrix}. \tag{4}$$

   Then

   $$P\mathbf{y} = \mathbf{1} \;\Leftrightarrow\; Q[\mathbf{1} \quad P]\begin{bmatrix}-1\\ \mathbf{y}\end{bmatrix} = \mathbf{0} \;\Leftrightarrow\; \begin{cases} \mathbf{p}^*\mathbf{y} = 1 \\ G\mathbf{y} = \mathbf{0} \end{cases} \tag{5}$$

   Thus, $\{\mathbf{y}_i\}$ is a basis for the null space of the matrix G, and can be conveniently found from an SVD of G. Generically (after prefiltering, see below), there are precisely d solutions.

2. *Decouple:* find a basis $\{\bar{\mathbf{w}}_1 \otimes \mathbf{w}_1, \cdots, \bar{\mathbf{w}}_d \otimes \mathbf{w}_d\}$ that spans the same linear subspace as $\{\mathbf{y}_1, \cdots, \mathbf{y}_d\}$. Since
$$\bar{\mathbf{w}}_i \otimes \mathbf{w}_i = \mathrm{vec}(\mathbf{w}_i \mathbf{w}_i^*)$$
we can associate to each structured basis vector a rank-1 hermitian matrix. In the same way we can associate to each $\mathbf{y}_i$ a $d \times d$ matrix Y_i such that $\mathrm{vec}(Y_i) = \mathbf{y}_i$. Since each $\mathbf{y}_i$ is in the span of $\{\bar{\mathbf{w}}_i \otimes \mathbf{w}_i\}$, each Y_i is an unknown linear combination of the rank-1 matrices:
$$\begin{cases} Y_1 = \lambda_{11}\mathbf{w}_1\mathbf{w}_1^* + \cdots + \lambda_{1d}\mathbf{w}_d\mathbf{w}_d^* \\ \quad\vdots \\ Y_d = \lambda_{d1}\mathbf{w}_1\mathbf{w}_1^* + \cdots + \lambda_{dd}\mathbf{w}_d\mathbf{w}_d^* \end{cases} \Leftrightarrow \begin{cases} Y_1 = W\Lambda_1 W^* \\ \quad\vdots \\ Y_d = W\Lambda_d W^* \end{cases}$$
where $\Lambda_i = \mathrm{diag}[\lambda_{i1}, \cdots, \lambda_{id}]$ and $W = [\mathbf{w}_1, \cdots, \mathbf{w}_d]$.
This is a joint diagonalization problem, a generalization of the standard eigenvalue decomposition, and can be solved.
3. *Scale each solution* $\mathbf{w}_i$ such that the average output power is equal to 1.

In the noise-free case and with $n \geq d^2$, this algorithm produces the exact separating beamformer $W = A^{\dagger *}$.

By squaring (4), we obtain explicit expressions for $\mathbf{p}$ and $C := G^*G$ that will be useful later:
$$\begin{aligned} \mathbf{p} &= \tfrac{1}{n}P^*\mathbf{1} = \tfrac{1}{n}\sum \bar{\mathbf{x}}_k \otimes \mathbf{x}_k \\ C &:= G^*G = \tfrac{1}{n}P^*P - \mathbf{p}\mathbf{p}^* \qquad (6)\\ &= \tfrac{1}{n}\sum(\bar{\mathbf{x}}_k \otimes \mathbf{x}_k)(\bar{\mathbf{x}}_k \otimes \mathbf{x}_k)^* - \left[\tfrac{1}{n}\sum \bar{\mathbf{x}}_k \otimes \mathbf{x}_k\right]\left[\tfrac{1}{n}\sum \bar{\mathbf{x}}_k \otimes \mathbf{x}_k\right]^* . \end{aligned}$$
The former expression shows that (for $\mathbf{y} = \bar{\mathbf{w}} \otimes \mathbf{w}$)
$$\mathbf{p}^*\mathbf{y} = \left[\tfrac{1}{n}\sum \bar{\mathbf{x}}_k \otimes \mathbf{x}_k\right]^* \mathbf{y} = \mathbf{w}^*\left[\tfrac{1}{n}\sum \mathbf{x}_k\mathbf{x}_k^*\right]\mathbf{w} = \mathbf{w}^* R_x \mathbf{w} \qquad (7)$$
Thus, the condition $\mathbf{p}^*\mathbf{y} = 1$ in (5) is implemented by the last step of the algorithm outline, where the average output power of each beamformer is fixed to 1.

Whitening and rank reduction

To avoid problems with nonunique solutions that would occur if A is a tall matrix, we have to perform a dimension-reducing prefiltering. The underscore (_) is used to denote prefiltered variables. Thus, let $\underline{\tilde{X}} := F^*\tilde{X}$ where $F : m \times d$. Then
$$\underline{\tilde{X}} = \underline{A}S + \underline{N}, \qquad \text{where} \quad \underline{A} := F^*A, \; \underline{N} := F^*N .$$
$\underline{\tilde{X}}$ has only d channels. The blind beamforming problem is now replaced by finding a separating beamforming matrix $T : d \times d$ with columns $\mathbf{t}_i$, acting on $\underline{\tilde{X}}$. After T has been found, the beamforming matrix on the original data will be $W = FT$.

Assuming white i.i.d. noise with covariance matrix $\sigma^2 I$, we will choose F such that the data is whitened. Let $\tilde{R}_x = \frac{1}{n}\tilde{X}\tilde{X}^*$ be the noisy sample data covariance matrix, with eigenvalue decomposition
$$\tilde{R}_x = U\Sigma^2 U^* . \qquad (8)$$
Here, U is $m \times m$ unitary, and Σ is $m \times m$ diagonal. Collect the d largest eigenvalues into a diagonal matrix $\hat{\Sigma}^2$ and the corresponding d eigenvectors into $\hat{U}$, and define F as
$$F = \hat{U}\hat{\Sigma}^{-1} . \qquad (9)$$
This prewhitening is such that $\underline{\tilde{R}}_x := \frac{1}{n}\underline{\tilde{X}}\underline{\tilde{X}}^*$ becomes unity: $\underline{\tilde{R}}_x = I$. After prewhitening, we can continue with the algorithm as outlined before. See [2] for details.

4. ACMA IN NOISE

Let us now assume that our observations are noise perturbed: $\tilde{\mathbf{x}}_k = \mathbf{x}_k + \mathbf{n}_k$ $(k = 1, \cdots, n)$. Our objective in this section is to rederive the ACMA procedure by starting from the CMA(2,2) cost function, here determinstically defined as
$$\mathbf{w} = \underset{\mathbf{w}}{\mathrm{argmin}}\, \frac{1}{n}\sum(|\hat{s}_k|^2 - 1)^2 , \qquad \hat{s}_k = \mathbf{w}^*\tilde{\mathbf{x}}_k .$$
Following the outline of section 3, and using the same factorization as in (4), we can make a similar derivation:
$$\begin{aligned} \tfrac{1}{n}\sum(|\hat{s}_k|^2 - 1)^2 &= \tfrac{1}{n}\sum\left[(\bar{\tilde{\mathbf{x}}}_k \otimes \tilde{\mathbf{x}}_k)^*(\bar{\mathbf{w}} \otimes \mathbf{w}) - 1\right]^2 \\ &= \tfrac{1}{n}\|\tilde{P}\mathbf{y} - \mathbf{1}\|^2 \qquad (\mathbf{y} = \bar{\mathbf{w}} \otimes \mathbf{w}) \\ &= |\tilde{\mathbf{p}}^*\mathbf{y} - 1|^2 + \|\tilde{G}\mathbf{y}\|^2 . \end{aligned}$$
Let $\hat{\mathbf{y}}$ be the (structured) minimizer of this expression, and define $\hat{\beta} = \tilde{\mathbf{p}}^*\hat{\mathbf{y}}$. We can add a condition that $\tilde{\mathbf{p}}^*\mathbf{y} = \hat{\beta}$ to the optimization problem without changing the outcome:
$$\begin{aligned} \hat{\mathbf{y}} &= \underset{\substack{\mathbf{y} = \bar{\mathbf{w}} \otimes \mathbf{w} \\ \tilde{\mathbf{p}}^*\mathbf{y} = \hat{\beta}}}{\mathrm{argmin}} |\tilde{\mathbf{p}}^*\mathbf{y} - 1|^2 + \|\tilde{G}\mathbf{y}\|^2 = \underset{\substack{\mathbf{y} = \bar{\mathbf{w}} \otimes \mathbf{w} \\ \tilde{\mathbf{p}}^*\mathbf{y} = \hat{\beta}}}{\mathrm{argmin}} |\hat{\beta} - \beta|^2 + \|\tilde{G}\mathbf{y}\|^2 \\ &= \underset{\substack{\mathbf{y} = \bar{\mathbf{w}} \otimes \mathbf{w} \\ \tilde{\mathbf{p}}^*\mathbf{y} = \hat{\beta}}}{\mathrm{argmin}} \|\tilde{G}\mathbf{y}\|^2 . \end{aligned}$$
Scaling $\hat{\beta}$ to 1 will scale the solution $\hat{\mathbf{y}}$ accordingly, and does not affect the fact that it has a Kronecker structure. The scaled condition $\tilde{\mathbf{p}}^*\mathbf{y} = 1$ in turn motivates in a natural way the choice of a prewhitening filter F as given in (9), viz.
$$\underline{\tilde{\mathbf{x}}} = F^*\tilde{\mathbf{x}}, \qquad \mathbf{w} = F\mathbf{t}, \qquad \text{where} \quad F = \hat{U}\hat{\Sigma}^{-1} .$$
Indeed, we derived in (7) that $\tilde{\mathbf{p}}^*\mathbf{y} = \mathbf{w}^*\tilde{R}_x\mathbf{w}$. If we change variables by prewhitening with dimension reduction, $\underline{\tilde{\mathbf{x}}} = \hat{\Sigma}^{-1}\hat{U}^*\tilde{\mathbf{x}}$ and $\mathbf{w} = \hat{U}\hat{\Sigma}^{-1}\mathbf{t}$, then $\underline{\tilde{R}}_x = I$ and $\mathbf{w}^*\tilde{R}_x\mathbf{w} = \mathbf{t}^*\mathbf{t}$. Moreover, $\|\underline{\mathbf{y}}\|^2 = \underline{\mathbf{y}}^*\underline{\mathbf{y}} = (\bar{\mathbf{t}} \otimes \mathbf{t})^*(\bar{\mathbf{t}} \otimes \mathbf{t}) = \bar{\mathbf{t}}^*\bar{\mathbf{t}} \otimes \mathbf{t}^*\mathbf{t} = \|\bar{\mathbf{t}}\|^2 \otimes \|\mathbf{t}\|^2 = \|\mathbf{t}\|^4$. It thus follows that the linear constraint on $\mathbf{y}$ can be replaced by a more pleasant unit-norm constraint on $\underline{\mathbf{y}}$ in the whitened domain. In summary, we have obtained
$$\mathbf{w} = \underset{\substack{\mathbf{y} = \bar{\mathbf{w}} \otimes \mathbf{w} \\ \mathbf{w}^*\tilde{R}_x\mathbf{w} = 1}}{\mathrm{argmin}} \|\tilde{G}\mathbf{y}\|^2 \quad \text{or} \quad \mathbf{t} = \underset{\substack{\mathbf{y} = \bar{\mathbf{t}} \otimes \mathbf{t} \\ \|\underline{\mathbf{y}}\| = 1}}{\mathrm{argmin}} \|\underline{\tilde{G}}\underline{\mathbf{y}}\|^2 . \qquad (10)$$
The first minimization problem is equivalent to the CMA(2,2) problem up to a scaling which is not important. The second minimization problem is almost equal to the first, except that the whitening also involves a dimension reduction: this will force $\mathbf{w} = \hat{U}\hat{\Sigma}^{-1}\mathbf{t}$ to lie in the dominant column span of $\tilde{X}$.

At this point, ACMA and CMA(2,2) will diverge in two distinct but closely related directions. CMA(2,2) has to numerically optimize the first minimization problem in (10), and find d independent solutions. With noise, the solutions will not exactly be in the approximate nullspace of $\underline{\tilde{G}}$.

ACMA is making a twist on this problem: instead of solving for the true minimum, it first finds a basis for the the d-dimensional approximate nullspace of $\underline{\tilde{G}}$, then looks for unit-norm vectors in this subspace that best fit the required structure. As we show in the next section, this modification ensures that asymptotically ACMA converges to the Wiener solution, whereas CMA(2,2) is known to be close but unequal to the Wiener solution [3].

5. ASYMPTOTIC BEHAVIOR

In the noiseless case, we have derived in (6) an expression for $C = G^* G$. In the presence of noise, $\tilde{\mathbf{x}}_k = A\mathbf{s}_k + \mathbf{n}_k$, assume that we compute in the same way

$$\tilde{C} = \frac{1}{n}\sum(\bar{\tilde{\mathbf{x}}}_k \otimes \tilde{\mathbf{x}}_k)(\bar{\tilde{\mathbf{x}}}_k \otimes \tilde{\mathbf{x}}_k)^* - \frac{1}{n}\left[\sum \bar{\tilde{\mathbf{x}}}_k \otimes \tilde{\mathbf{x}}_k\right] \frac{1}{n}\left[\sum \bar{\tilde{\mathbf{x}}}_k \otimes \tilde{\mathbf{x}}_k\right]^* . \tag{11}$$

We analyze the contribution of the noise in this expression, assuming that it is zero mean, circularly symmetric, independent of the sources, and with finite covariance $R_n = \mathrm{E}(\mathbf{n}\mathbf{n}^*)$ and fourth-order moments.

Cumulants

The asymptotic analysis is best done via the introduction of fourth-order cumulants. For a zero-mean circularly symmetric vector-signal $\mathbf{x}(k)$ with components $x_i(k)$, define the tensor with entries

$$\begin{aligned}\kappa_{i,k}^{j,l} &:= \mathit{cum}(x_i, \bar{x}_j, x_k, \bar{x}_l) \\ &:= \mathrm{E}(x_i\bar{x}_j x_k \bar{x}_l) - \mathrm{E}(x_i\bar{x}_j)\mathrm{E}(x_k\bar{x}_l) - \mathrm{E}(x_i\bar{x}_l)\mathrm{E}(x_k\bar{x}_j)\end{aligned}$$

where $i,j,k,l = 1,\cdots,m$ and m is the dimension of $\mathbf{x}$. If we collect the entries $\kappa_{i,k}^{j,l}$ into a matrix K_x with entries $K_{i+jm,l+km} = \kappa_{i,k}^{j,l}$, then

$$K_x = \mathrm{E}[(\bar{\mathbf{x}}\otimes\mathbf{x})(\bar{\mathbf{x}}\otimes\mathbf{x})^*] - \mathrm{E}[\bar{\mathbf{x}}\otimes\mathbf{x}]\mathrm{E}[\bar{\mathbf{x}}\otimes\mathbf{x})]^* - \mathrm{E}[\mathbf{x}\mathbf{x}^*]^T \otimes \mathrm{E}[\mathbf{x}\mathbf{x}^*]$$

Note that $\mathrm{E}[\mathbf{x}\mathbf{x}^*] = R_x$, $\mathrm{E}[\bar{\mathbf{x}}\otimes\mathbf{x}] = \mathrm{vec}(R_x)$. Comparing to (11), it is seen that, asymptotically,

$$C = K_x + R_x^T \otimes R_x\,, \qquad \tilde{C} = \tilde{K}_x + \tilde{R}_x^T \otimes \tilde{R}_x\,.$$

Cumulants are used because they have several well-known nice properties, such as multilinearity, additivity and the fact that Gaussian signals have zero cumulants. For our model $\tilde{\mathbf{x}}_k = A\mathbf{s}_k + \mathbf{n}_k$, assuming independent circularly symmetric CM signals (autocumulants $\kappa = -1$) and independent Gaussian noise, this results in

$$\begin{aligned}\tilde{K}_x &= [\bar{A}\otimes A]\,K_s\,[\bar{A}\otimes A]^* + K_n \\ &= [\bar{A}\circ A]\,(-I)\,[\bar{A}\circ A]^*\end{aligned} \tag{12}$$

Using these properties we can derive that, *without noise*, the CMA(2,2) or ACMA criterion matrix becomes (using $R_x = AA^*$)

$$\begin{aligned}C &= K_x + R_x^T \otimes R_x \\ &= [\bar{A}\otimes A]K_s[\bar{A}\otimes A]^* + \bar{A}\bar{A}^* \otimes AA^* \\ &= [\bar{A}\otimes A]\,(K_s+I)\,[\bar{A}\otimes A]^* \quad = \quad [\bar{A}\otimes A]C_s[\bar{A}\otimes A]^*\,.\end{aligned} \tag{13}$$

Note that $C_s = K_s + I$ is diagonal, with zero entries at the location of the source autocumulants, and '1' entries elsewhere on the diagonal. The null space of C_s is given by $\{\mathbf{e}_i\otimes\mathbf{e}_i\}$, and hence the null space of C by $\{\bar{\mathbf{w}}_i\otimes\mathbf{w}_i\}$, plus the null space of $[\bar{A}\otimes A]^*$ (this is removed by prefiltering).

With noise, the CMA(2,2) or ACMA criterion matrix becomes ($\tilde{R}_x = R_x + R_n$)

$$\tilde{C} = \tilde{K}_x + \tilde{R}_x^T \otimes \tilde{R}_x \;=\; C \;+\; E \;+\; C_n \tag{14}$$

where C is given in (13) and

$$E := R_x^T \otimes R_n + R_n^T \otimes R_x\,, \quad C_n := K_n + R_n^T \otimes R_n\,.$$

Thus, the noise contributes a second-order and a fourth-order term to the ACMA criterion matrix, even if it would be Gaussian. In a previous paper, we tried to remove this bias and derived the W-ACMA algorithm [4], whose asymptotic performance is close to a zero-forcing (ZF) beamformer. However, as we show next, this bias is precisely such that ACMA converges to the Wiener solution.

Asymptotic analysis of ACMA

In the analysis of ACMA, we also have to take the effect of the initial prewhitening step into account. Recall that this step is $\underline{\tilde{X}} = F^*\tilde{X} = \underline{A}S + \underline{N}$, where $F = \hat{U}\hat{\Sigma}^{-1}$ so that $\underline{\tilde{R}}_x = F^*\tilde{R}_x F = I$. In the whitened domain, we search for d-dimensional beamformers $\mathbf{t}$, the overall beamformers are then given by $\mathbf{w} = F\mathbf{t}$.

Introducing this into the first expression for $\tilde{C}$ in (14), assuming Gaussian noise ($K_n = 0$), and using (12), we obtain

$$\begin{aligned}\underline{\tilde{C}} &= (F\otimes F)^*\tilde{C}(F\otimes F) \\ &= (\bar{F}\otimes F)^*\tilde{K}_x(\bar{F}\otimes F) \;+\; I\otimes I \\ &= [\bar{\underline{A}}\circ\underline{A}](-I)[\bar{\underline{A}}\circ\underline{A}]^* \;+\; I\,.\end{aligned} \tag{15}$$

Inserting this in the CMA(2,2) cost function, equation (10), it follows that both ACMA and CMA(2,2) look at the optimization problem

$$\begin{aligned}\underset{\mathbf{y}=\bar{\mathbf{t}}\otimes\mathbf{t},\|\mathbf{y}\|=1}{\operatorname{argmin}} \;\mathbf{y}^*\underline{\tilde{C}}\mathbf{y} &= \underset{\mathbf{y}=\bar{\mathbf{t}}\otimes\mathbf{t},\|\mathbf{y}\|=1}{\operatorname{argmin}} \;\mathbf{y}^*\left\{[\bar{\underline{A}}\circ\underline{A}](-I)[\bar{\underline{A}}\circ\underline{A}]^* + I\right\}\mathbf{y} \\ &= \underset{\mathbf{y}=\bar{\mathbf{t}}\otimes\mathbf{t},\|\mathbf{y}\|=1}{\operatorname{argmax}} \;\mathbf{y}^*\left\{[\bar{\underline{A}}\circ\underline{A}][\bar{\underline{A}}\circ\underline{A}]^*\right\}\mathbf{y}\,.\end{aligned} \tag{16}$$

CMA(2,2) continues to optimize this problem. The result is in general *not* the desired vectors of the form $\bar{\underline{\mathbf{a}}}_i\otimes\underline{\mathbf{a}}_i$, unless $\underline{A}$ is unitary. ACMA is taking a slightly different approach at this point: it does not optimize (16), but solves the unstructured problem first. Indeed, it looks for an unconstrained d-dimensional basis $\{\mathbf{y}_i\}$ of the approximate null space of $\underline{\tilde{G}}$, equivalently the null space of $\underline{\tilde{C}}$, or d independent unit-norm vectors $\mathbf{y}$ that minimize $\mathbf{y}^*\underline{\tilde{C}}\mathbf{y}$. With the factorization in (16), we see that these are the d dominant eigenvectors of $[\bar{\underline{A}}\circ\underline{A}][\bar{\underline{A}}\circ\underline{A}]^*$. Since this is a rank-$d$ matrix, we have that the d dominant eigenvectors together span the same subspace as the column span of $[\bar{\underline{A}}\circ\underline{A}]$, hence

$$\mathrm{span}\{\mathbf{y}_1,\cdots,\mathbf{y}_d\} = \mathrm{span}[\bar{\underline{A}}\circ\underline{A}] = \mathrm{span}\{\bar{\underline{\mathbf{a}}}_1\otimes\underline{\mathbf{a}}_1\,,\cdots,\bar{\underline{\mathbf{a}}}_d\otimes\underline{\mathbf{a}}_d\}$$

As a second step, it will use the joint diagonalization to replace the unstructured basis by one that has the required Kronecker product structure, i.e., look for d independent vectors of the form $\bar{\mathbf{t}}\otimes\mathbf{t}$ within this column span. From the above equation, we see that the unique solution is $\bar{\mathbf{t}}_i\otimes\mathbf{t}_i = \bar{\underline{\mathbf{a}}}_i\otimes\underline{\mathbf{a}}_i$, so that

$$\mathbf{t}_i = \underline{\mathbf{a}}_i\,, \qquad i = 1,\cdots,d\,.$$

The beamformer on the whitened problem is equal to the whitened direction vector (a matched spatial filter). If we go back to the resulting beamformer on the original (unwhitened) data matrix X, we find (for $i = 1,\cdots,d$)

$$\mathbf{t}_i = \underline{\mathbf{a}}_i = F^*\mathbf{a}_i \qquad \Rightarrow \qquad \mathbf{w}_i = F\mathbf{t}_i = FF^*\mathbf{a}_i = \tilde{R}_x^{-1}\mathbf{a}_i\,, \tag{17}$$

since $F = \hat{U}\hat{\Sigma}^{-1}$, $\tilde{R}_x = U\Sigma^2U^* = \hat{U}\hat{\Sigma}^2\hat{U} + \sigma^2\hat{U}^\perp\hat{U}^{\perp *}$, and $\hat{U}^{\perp *}\mathbf{a}_i = 0$. We have just shown that, with Gaussian noise, ACMA is asymptotically equal to the Wiener solution (alias MMSE beamformer). In general, this is a very attractive property.

Connection to JADE

JADE [5] is a widely used algorithm for the blind separation of independent non-Gaussian sources in Gaussian noise. It is based on the construction of the fourth-order cumulant matrix $\tilde{K}_x$ in equation (12), but uses a different prefiltering strategy, $F = \hat{U}(\hat{\Sigma}^2 - \sigma^2 I)^{-1/2}$ where $\hat{U}$ and $\hat{\Sigma}$ are estimated from the eigenvalue decomposition of $\tilde{R}_x$. The prefiltering leads to $\underline{\underline{\tilde{X}}} = F^*\tilde{X} = \underline{\underline{A}}S + \underline{\underline{N}}$, where $\underline{\underline{A}} = F^*A$.

This choice is motivated by the fact that, asymptotically, F converges to $F = U_A\Sigma_A^{-1}$ (based on the SVD $A = U_A\Sigma_A V_A$), and thus

$$\underline{\underline{A}} \;=\; \Sigma_A^{-1}U_A^{*}A \;=\; V_A$$

is a unitary matrix. Asymptotically, the fourth-order cumulant matrix is given by

$$\underline{\underline{\tilde{K}}}_x = [\underline{\bar{A}} \circ \underline{A}]\,(-I)\,[\underline{\bar{A}} \circ \underline{A}]^{*}$$

JADE computes a basis of the dominant column span of this matrix, which in this asymptotic situation spans the same subspace as

$$\{\underline{\underline{\bar{\mathbf{a}}}}_i \otimes \underline{\underline{\mathbf{a}}}_i\,;\; i = 1,\cdots,d\}$$

Like ACMA, it then performs a joint diagonalization to identify the vectors $\underline{\underline{\mathbf{a}}}_i$. After correcting for the prefiltering, we find

$$T = \underline{\underline{A}} = V_A \quad \Rightarrow \quad W = FT = U_A\Sigma_A^{-1}V_A = A^{\dagger *}.$$

Hence, this strategy leads asymptotically to the zero-forcing beamformer, as well as the true A-matrix.

Apart from different prefiltering, the asymptotic equations of JADE and ACMA look similar. However, the finite-sample properties are quite different. In the absence of noise, the null space information of $\tilde{C}$ in ACMA is exact by construction, and hence the algorithm produces the exact separating beamformers. The dominant column span of $\tilde{K}_x$ used in JADE is not exact since the signal sources do not decorrelate exactly in finite samples: K_s is a full matrix. Thus, keeping the number of samples fixed, the SNR-asymptotic performance of JADE saturates.

6. SIMULATIONS

Some performance results are shown in figures 1 and 2. In this simulation, we took a uniform linear array consisting of $m = 4$ antennas, and $d = 3$ equal-power constant-modulus sources with directions $-10°, 0°, 20°$ respectively. We compare the performance of ACMA, W-ACMA [4] and JADE [5], both for the residual signal to interference ratio (SIR), which indicates how well the computed beamforming matrix W is an inverse of A, and for the Signal to Interference and Noise ratio (SINR), indicating the output error.

Figure 1 shows that the SIR performance of JADE saturates for finite n because it relies on the convergence of fourth-order statistics, whereas the SIR performance of ACMA saturates for finite SNR, because it converges to the Wiener solution and hence it is biased. It is seen that the whitening in W-ACMA removes this saturation so that it can converge to a few dB below the ZF solution. Figure 2 shows that ACMA converges asymptotically (in n) to the Wiener solution.

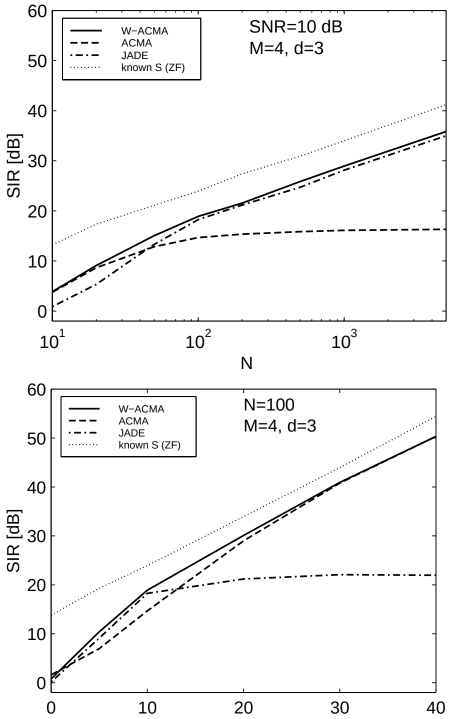

Figure 1. SIR performance of W-ACMA, ACMA and JADE

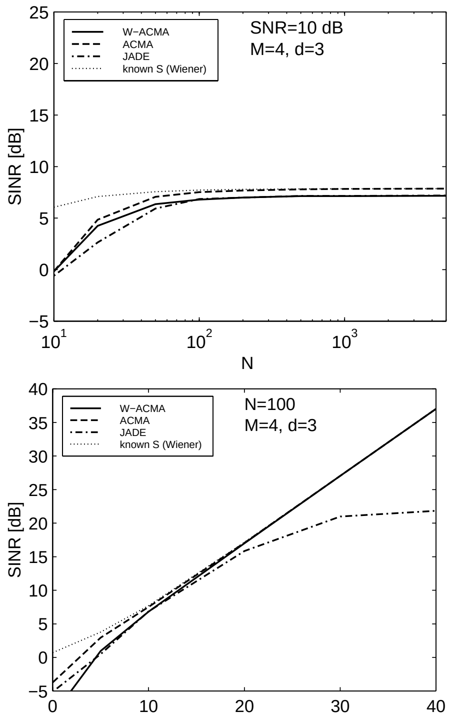

Figure 2. SINR performance of W-ACMA, ACMA and JADE

References

[1] J.J. Shynk and R.P. Gooch, "The constant modulus array for cochannel signal copy and direction finding," *IEEE Trans. Signal Proc.*, vol. 44, pp. 652–660, Mar. 1996.

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

[3] H.H. Zeng, Lang Tong, and C.R. Johnson, "Relationships between the constant modulus and Wiener receivers," *IEEE Trans. Inform. Theory*, vol. 44, pp. 1523–1538, July 1998.

[4] A.J. van der Veen, "Weighted ACMA," in *Proc. IEEE ICASSP*, (Phoenix (AZ)), IEEE, Mar. 1999.

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