[2] R. Roy and T. Kailath, "ESPRIT—Estimation of signal parameters via rotational invariance techniques," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 37, no. 7, pp. 984–995, July 1989.
[3] A. Swindlehurst, B. Ottersten, R. Roy, and T. Kailath, "Multiple invariance ESPRIT," *IEEE Trans. Signal Processing*, vol. 40, no. 4, pp. 867–881, Apr. 1992.
[4] D. V. B. Rao and S. Y. Kung, "A state space approach for the 2-D harmonic retrieval problem," in *Proc. IEEE ICASSP'84*, pp. 4.10.1–4.10.4.
[5] M. Zoltowski and D. Stavrinides, "Sensor array signal processing via a Procrustes rotations based eigenanalysis of the ESPIRT data pencil," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 37, no. 6, pp. 832–861, June 1989.
[6] A. Swindlehurst and T. Kailath, "2-D parameter estimation using arrays with multidimensional invariance structure," in *Proc. 23rd Asilomar Conf. Signals, Syst., Comput.* (Asilomar, CA), Nov. 1989, pp. 950–954.
[7] R. Johnson and G. Miner, "An operational system implementation of the ESPRIT DF algorithm," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 27, no. 1, pp. 159–166, Jan. 1991.
[8] A. Swindlehurst and T. Kailath, "Algorithms for azimuth/elevation direction finding using regular array geometries," in *IEEE Trans. Aerosp. Electron. Syst.*, to be published, Jan. 1993.
[9] P. Ober, E. Deprettere, and A. van der Veen, "Efficient methods to compute azimuth and elevation in high resolution DOA estimation," in *Proc. ICASSP'91*, pp. 3349–3352.
[10] R. A. Horn and C. A. Johnson, *Matrix Analysis*. Cambridge, England: Cambridge University Press, 1985.

Azimuth and Elevation Computation in High Resolution DOA Estimation

A. J. van der Veen, P. B. Ober, and E. F. Deprettere

***Abstract*—In this correspondence, we discuss a number of high-resolution direction finding methods for determining the two-dimensional directions of arrival of a number of plane waves, impinging on a sensor array. The array consists of triplets of sensors that are identical, as an extension of the 1D ESPRIT scenario to two dimensions. New algorithms are devised that yield the correct parameter pairs while avoiding an extensive search over the two separate one-dimensional parameter sets.**

I. Introduction

ESPRIT [1] and ESPRIT-like subspace based high resolution DOA algorithms (e.g., [2], [3], see [4]–[6] for an overview) are usually designed to determine the directions of arrival of narrow-band noncoherent signals in only one parameter dimension, i.e., array and waves are confined to a single plane. The extension to the 2D case, where both azimuth and elevation angles have to be determined, is in general nontrivial. The decomposition of the problem into two independent 1D problems results in two decoupled parameter sets, which have to be combined to correct parameter pairs. The above-mentioned approaches to solve the 1D DOA problem exploit the translation-invariant structure present in the array geometry and typically end up with a matrix pencil, formed on the data after noise reduction by SVD-based algorithms. The generalized eigenvalues of the pencil contain (implicitly) the directions of arrival of the impinging signals. In the 1D based 2D case, two pencils are constructed on three data matrices, and, after having solved these pencils independently, the problem of eigenvalue matching occurs: an algorithm must be found to combine azimuths and elevations correctly. This parameter matching is essentially what makes the 2D problem more difficult to solve. Different approaches have been alluded to in the literature. For example, Zoltowski and Stravrinides [7] describe a correlation technique resulting ultimately in a MUSIC search in parameter space, which is computationally not very attractive. A second approach in [7] is obtained by constructing a third matrix pencil, having generalized eigenvalues equal to the quotient of the correct parameter pairs. These quotients can be used to search for the correct pairs. Another method is described briefly in [8], in which the matching is determined by checking a certain optimization criterion for all possible pairs of angles. This is an $\mathcal{O}(d!)$ process, where d is the number of sources, and without special programming techniques this is feasible only for small d.

In this correspondence, we are interested in algebraic methods that avoid such a combinatorial search. Two new approaches are considered, both based on the observation that the data matrices share the same common set of eigenvectors in case no noise is present. Two matrices sharing the same eigenvectors can be diagonalized by the same similarity transform, and hence the two underlying pencils can be solved for by diagonalizing the first pencil, and applying the similarity transform that was needed to the second. The correct pairings are found directly and without extra effort. When noise is present, this property is lost. We will describe how this method can be adapted such that the correct eigenvalues of each pencil are determined, along with an estimate of the pairing. The first new approach is based on this idea and yields good results, yet is "in style" with the VLSI parallel array architecture described in [5] and is extremely cheap in the number of extra operations. See Section III-A.

The second new approach is based on the idea to approximate the data matrices by adding small perturbation matrices such that the resulting matrices will have equal eigenvectors, or, equivalently, that the two resulting matrices will commute. This approach was briefly reported in [9]. It was brought to our attention that a similar, although not identical, approach was presented independently by Swindlehurst and Kailath [10], [11], who perturb only one of the two matrices such that its eigenvectors will coincide with the eigenvectors of the other (unperturbed) matrix.

II. Preliminaries

A. The Data Model

Consider m sensor triplets, each composed of three identical sensors with unknown gain and phase patterns, which may vary from triplet to triplet. For every triplet, the displacement vectors $\boldsymbol{d}_{xy}$ and $\boldsymbol{d}_{xz}$ between its components are required to be the same. This way three identical, although displaced arrays are obtained, for reference denoted by X, Y, and Z. This is a direct extension of the 1D ESPRIT scenario to two dimensions. Impinging on every array are d narrow-band noncoherent signals $s_k(t)$, having an unknown complex amplitude and a known center frequency ω_0. Assuming additive, stationary, and zero-mean noise, the output signal of the ith

Manuscript received August 23, 1991; revised January 16, 1992. This work was supported in part by the Commission of the EC under ESPRIT BRA Program 3280 (NANA).

The authors are with the Department of Electrical Engineering, Delft University of Technology, 2628 CD Delft, The Netherlands.

IEEE Log Number 9200246.

1053-587X/92$03.00 © 1992 IEEE

sensor of each array will be

$$x_i(t) = \sum_{k=1}^{d} a_{ik}s_k(t) + n_{xi}(t)$$

$$y_i(t) = \sum_{k=1}^{d} a_{ik}\phi_k s_k(t) + n_{yi}(t)$$

$$z_i(t) = \sum_{k=1}^{d} a_{ik}\theta_k s_k(t) + n_{zi}(t) \quad (1)$$

where a_{ik} is the gain of sensor i for signal $s_k(t)$; n_{xi}, n_{yi}, n_{zi} represent noise signals; and ϕ_k and θ_k are complex unit scaling factors representing the phase shift caused by the propagation time difference of the kth incident wave, and hence directly depending upon its direction of arrival. By collecting N snapshots from each sensor, data matrices X, Y, and Z are formed, obeying

$$X = AS + N_x$$

$$Y = A\Phi S + N_y, \qquad \Phi = \text{diag}\,(\phi_1, \phi_2, \cdots, \phi_d)$$

$$Z = A\Theta S + N_z, \qquad \Theta = \text{diag}\,(\theta_1, \theta_2, \cdots, \theta_d) \quad (2)$$

where $X(i, j) = x_i(t_j)$, $A(i, k) = a_{ik}$, $S(k, j) = s_k(t_j)$ and $N_x(i, j) = n_{xi}(t_j)$, and likewise for Y and Z. X, Y, and Z are the data matrices (dimension $m \times N$), A is the array gain matrix ($m \times d$) and S is the signal matrix ($d \times N$). The matrices A and S are unknown, and are not rank deficient by assumption. The matrices Φ and Θ are diagonal and contain the phase shifts for each signal. The DOA problem is to estimate Φ and Θ. From these matrices, the angles of arrival can directly be computed.

B. The Subspace Approach

Matrix polynomials of the form $E - \alpha F$, $\alpha \in \mathbb{C}$, are called matrix pencils. Forming the pencils

$$X - \lambda Y = A(I - \lambda\Phi)S + (N_x - \lambda N_y)$$

$$X - \mu Z = A(I - \mu\Theta)S + (N_x - \mu N_z)$$

it is seen that, in the noise-free case, numbers $\lambda = \lambda_i$ and $\mu = \mu_j$, $i, j = 1, 2, \cdots, d$, that reduce the rank of the pencil by one are equal to ϕ_i^{-1} and θ_j^{-1}, respectively. With square data matrices, these rank reducing numbers are the generalized eigenvalues of the matrix pairs (X, Y) and (X, Z).

With noise present, however, a large number of samples are taken to improve accuracy. As a result, X, Y, and Z will not be square. Noise will also increase the rank of the pencils, and this will introduce new rank reducing numbers. One way to continue is by computing a total least squares projection of the data matrices (see, e.g., [12] on this). Without noise, these matrices have a common (d-dimensional) row space and a common column space, and it is possible to find square $d \times d$ matrices E_x, E_y, and E_z, whose generalized eigenvalues are estimates of the d original rank reducing numbers. The next discussion closely follows the approach outlined in [5], [6]. Compute the SVD's of two matrices, constructed from X, Y, and Z:

$$[X \;\; Y \;\; Z] = U_1\Sigma_1V_1^H \qquad \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = U_2\Sigma_2V_2^H$$

where U_i and V_i are unitary matrices, Σ_i is a diagonal matrix containing the singular values, and ^{H} denotes Hermitian conjugation. Optimal (in the Frobenius norm) rank d approximations of these matrices are obtained by setting the $(m - d)$ smallest singular values in both Σ_1 and Σ_2 equal to zero, yielding $\hat{\Sigma}_1$ and $\hat{\Sigma}_2$, and matrices $\hat{U}_1$ and $\hat{V}_2$ containing the d columns of U_1 and V_2 that correspond to the nonzero singular values in $\hat{\Sigma}_1$ and $\hat{\Sigma}_2$, respectively. The TLS approximations $\hat{X}$, $\hat{Y}$, $\hat{Z}$ of the data matrices share the same column space range ($\hat{U}_1$) and row space range ($\hat{V}_2^H$), and are obtained by projecting X, Y, and Z onto these subspaces:

$$\hat{X} = (\hat{U}_1\hat{U}_1^H)X(\hat{V}_2\hat{V}_2^H) = \hat{U}_1E_x\hat{V}_2^H; \qquad E_x = \hat{U}_1^HX\hat{V}_2$$

$$\hat{Y} = (\hat{U}_1\hat{U}_1^H)Y(\hat{V}_2\hat{V}_2^H) = \hat{U}_1E_y\hat{V}_2^H; \qquad E_y = \hat{U}_1^HY\hat{V}_2$$

$$\hat{Z} = (\hat{U}_1\hat{U}_1^H)Z(\hat{V}_2\hat{V}_2^H) = \hat{U}_1E_z\hat{V}_2^H; \qquad E_z = \hat{U}_1^HZ\hat{V}_2 \quad (3)$$

in which E_x, E_y, and E_z are square $d \times d$ matrices. (For a more detailed discussion of the operations involved, and hints to a VLSI implementation, see [5].) In the noise-free case, the generalized eigenvalues GE (E_y, E_x), or the eigenvalues of $E_x^{-1}E_y$, are equal to the rank reducing numbers of the matrix pencil $\hat{X} - \lambda\hat{Y}$. The same holds true for the generalized eigenvalues of (E_z, E_x) and the pencil $\hat{X} - \lambda\hat{Z}$. Substituting (2) in (3), we have that

$$E_1 = E_x^{-1}E_y = S_e^{-1}\Phi S_e$$

$$E_2 = E_x^{-1}E_z = S_e^{-1}\Theta S_e \quad (4)$$

where $S_e = S\hat{V}_2$. From this set of equations it follows that we are interested in computing the eigenvalues of E_1 and E_2. It is also clear that E_1 and E_2 share, in the noise-free case, the same set of eigenvectors. This means they can be triangularized by the same unitary matrix Q: there exists unitary matrices Q_1, Q_2 such that

$$Q_1^HE_1Q_1 = Q_1^HS_e^{-1}\Phi S_eQ_1 = R_1$$

$$Q_2^HE_2Q_2 = Q_2^HS_e^{-1}\Theta S_eQ_2 = R_2 \quad (5)$$

with $Q_1 = Q_2 = Q$, and the upper triangular matrices R_1 and R_2 have main diagonals equal to Φ and Θ, respectively. Because the same matrix Q will triangularize both E_1 and E_2, the one-to-one correspondence between the ϕ_i's and θ_i's is preserved in the positional correspondence on the diagonals and no pair matching operation needs to be done. Of course, with noise present, Q_1 will differ somewhat from Q_2. This difference is assumed to be only small in the "rotational perturbation" approach in Section III-A.

To assess the difference in eigenvectors of E_1 and E_2, recall that two matrices with the same eigenvectors commute. Thus, in the noise-free case, $E_1E_2 = E_2E_1$. We will devise, in Section III-B, an algorithm that determines small additive perturbations of E_1 and E_2 such that the above relation holds.

III. Pair Matching Algorithms

A. Matching Using Rotational Perturbations

Experience gained with a Jacobi-iteration method for computing the Schur decomposition, as discussed in [5], led to the following new pair matching algorithm, which can be integrated with the eigenvalue computations that are needed anyway. The Jacobi-iteration method consists of a number of sweeps, which in turn consist of a certain number of 2×2 elementary (Givens) rotations that solve 2×2 Schur decompositions. Superimposed on the elementary rotations is a permutation scheme (rotations over $\pi/2$) to ensure ultimate convergence. Two observations from [5] are that, near convergence, the Givens rotation angles are close to 0, so that the rotation matrix is close to the identity matrix, and that the permutation scheme is such that after an even number of sweeps the entries on the main diagonal have their initial ordering.

The above observations are used to solve (5) in such a way that entries at corresponding positions on the main diagonals of R_1 and

R_2 are the estimated eigenvalue pairs. First triangularize E_1 as in (5): determine Q_1 to triangularize E_1, and apply this similarity transform also to E_2, yielding

$$Q_1^H E_1 Q_1 = R_1$$
$$Q_1^H E_2 Q_1 = R_2' \tag{6}$$

in which R_2' will be "almost upper triangular" (in the noise-free case, it would be upper triangular), and its diagonal entries are rough estimates of the true eigenvalues of E_2. According to the above discussion, the assumption is that only "small rotations" in the Jacobi iteration algorithm are needed to make R_2' upper triangular; i.e., there exists a unitary Q_2' that will triangularize R_2' in (6):

$$Q_2^{H\prime} R_2' Q_2' = R_2$$

and which is close to the identity matrix in operator norm. This means that Q_2' is a minimal rotational perturbation of Q_1 such that $Q_2 = Q_1 Q_2'$ triangularizes E_2, and Q_2' does not permute the rough eigenvalue estimates in R_2'. The correct eigenvalue pairs are thus the entries at the same position on the main diagonals of R_1 and R_2. Moreover, these eigenvalues are the eigenvalues of E_1 and E_2 and hence the same as obtained in, e.g., Zoltowski's method using a third matrix pencil [7]: in effect, we have solved two 1D independent eigenvalue problems on (X, Y) and (X, Z), but in a special way that gives us the correct pairing almost for free. (The pairing may be different from Zoltowski's.) In addition, this algorithm is amenable to parallel VLSI implementation in the same way as the 1D algorithm was [5]: the only operations that are needed are elementary rotations (on E_1 and R_2'), and a multilayer structure that allows the rotations that are performed on E_1 to be repeated on E_2 at the same time. Finally, we remark that permutationless rotations for Q_2' are obtained automatically when an *even* number of Jacobi sweeps is performed in the iterative computation of the Schur decomposition of R_2': at the end of every second sweep, the ordering of the diagonal entries is their original ordering (see [5]).

B. Forcing Commutativity of E_1 and E_2

A second pair matching algorithm is based on the following observation. In the presence of noise, E_1 and E_2 do not commute and can only be triangularized with different unitary matrices Q_1, Q_2:

$$Q_1^H E_1 Q_1 = R_1$$
$$Q_2^H E_2 Q_2 = R_2$$

where R_1 and R_2 are upper triangular, having diagonal entries that are approximations to the ϕ_i's and θ_i's of (1). It is not possible to find the correct tuples ϕ_k, θ_k directly, because their ordering along the diagonals of R_1 and R_2 may be different. As the noncommutativity of E_1 and E_2 is caused by additive noise, the idea is to (partially) cancel this noise by adding perturbation matrices to them in such a way that their commutativity is restored. Thus, we are looking for two perturbation matrices P_1 and P_2 such that

$$(E_1 + P_1)(E_2 + P_2) = (E_2 + P_2)(E_1 + P_1)$$

and such that P_1, P_2 obey some minimum norm constraint. Thus with, e.g., a Frobenius norm, the problem is

$$\min_{P_1, P_2} \|P_1\|_F^2 + \|P_2\|_F^2$$

such that

$$[P_1 \quad P_2 \quad I] \begin{bmatrix} -E_2 & 0 & I \\ E_1 & -I & 0 \\ E_2E_1 - E_1E_2 & -E_2 & E_1 \end{bmatrix} \begin{bmatrix} I \\ -P_1 \\ -P_2 \end{bmatrix} = 0. \tag{7}$$

After determining the solution to this nonlinear minimization problem, only the eigenvalue decomposition of $(E_1 + P_1)$ needs to be determined. The same unitary similarity transformation that will make $(E_1 + P_1)$ upper triangular will also triangularize $(E_2 + P_2)$. The assumedly more accurate parameter pairs follow directly from the entries of the diagonals of the resulting matrices.

The exact solution to this minimization problem is in general hard to find, but could be obtained by means of nonlinear programming. Note that (7) is a kind of generalized (constrained) Hamiltonian equation: it would represent a Ricatti equation if $[P_1 \quad P_2]$ would be a square matrix S, and then the problem would be

$$[S \quad I]\boldsymbol{H}\begin{bmatrix} I \\ -S \end{bmatrix} = 0$$

in which case S has a solution given in terms of the eigenvalues $\boldsymbol{H}$. However, this insight does not really help in solving our problem. The nonsquareness makes our problem singular, and implies that there is a collection of perturbation matrices that makes the matrices E_1, E_2 commute.

An approximate solution for P_1 and P_2 that is easier to compute can, however, be obtained by neglecting the term $P_1P_2 - P_2P_1$, under the assumption that the perturbations are small in comparison with the E_i anyway. This results in a condition, linear in the entries of P_1, P_2 that has close resemblance to a Lyapunov equation and can be solved by Kronecker sums. Evaluating this method, we arrive at the following equations:

$$[-E_2^T \oplus E_2 \quad E_1^T \oplus -E_1] \cdot \begin{bmatrix} \text{vec } (P_1) \\ \text{vec } (P_2) \end{bmatrix} = \text{vec } (E_1E_2 - E_2E_1)$$

(where the Kronecker sum $A \oplus B$ equals $A \otimes I + I \otimes B$, $\otimes$ is the Kronecker product, and vec (A) is a vector obtained by stacking the columns of A). From this equation we can find, using the Moore–Penrose pseudoinverse denoted by $^+$, the solution with minimum perturbation norm

$$\begin{bmatrix} \text{vec } (P_1) \\ \text{vec } (P_2) \end{bmatrix} = [-E_2^T \oplus E_2 \quad E_1^T \oplus -E_1]^+ \cdot \text{vec } (E_1E_2 - E_2E_1). \tag{8}$$

While solving the above equation in this way is obviously a computational overkill compared with the original problem, it does find the minimum norm solution to the approximate problem, and can give hints to a (much more) efficient solution. It should be remarked that the usual algorithm to the Lyapunov problem involves an eigenvalue decomposition of B, and cannot be used directly in our problem since the matrices are not square. Also note that, due to the approximation $P_1P_2 - P_2P_1 \approx 0$, the resulting matrices $(E_1 + P_1)$ and $(E_2 + P_2)$ do not commute precisely. Eigenvalue pairs can be obtained by computing an eigenvalue decomposition of $(E_1 + P_1)$, and using the eigenvectors of this matrix to approximately diagonalize $(E_2 + P_2)$, and subsequently ignoring the off-diagonal elements. Alternatively, a Schur decomposition can be used instead of an eigenvalue decomposition, much as in (6).

Another possible simplification of the problem (7) to make it solvable is to add a perturbation term only to E_2, i.e., to take $P_1 = 0$ (see also Swindlehurst and Kailath [10], [11]). The resulting matrices E_1 and $E_2 + P_2$ do commute, and P_2 is obtained via a Kronecker sum:

$$\text{vec } (P_2) = [E_1^T \oplus -E_1]^+ \text{ vec } (E_1E_2 - E_2E_1). \tag{9}$$

Because E_1 is not perturbed, the estimate for Φ is the same as the estimate obtained in Section III-A.: $\Phi = \text{eig } (E_1)$. The estimate for

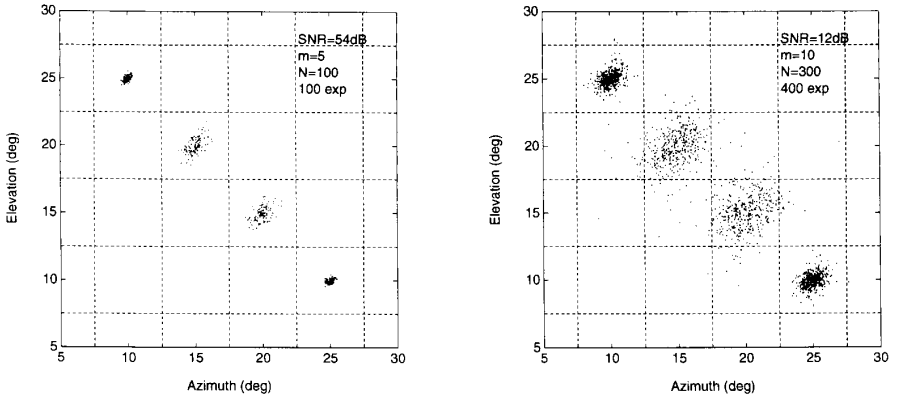

Fig. 1. Method 1: (a) 54 dB; (b) 12 dB.

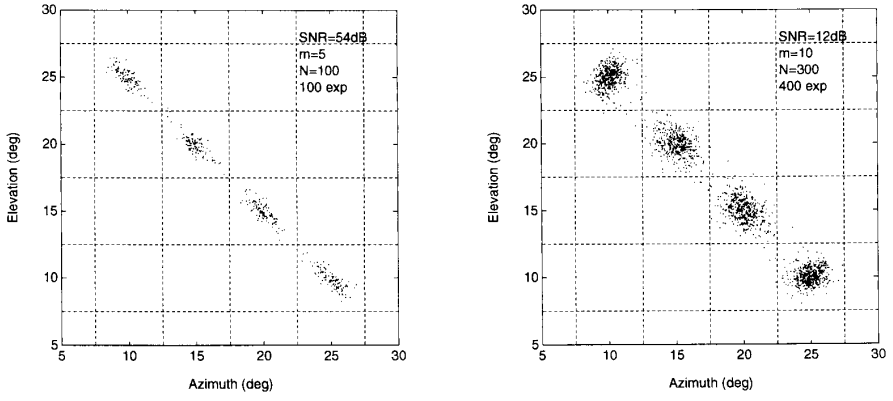

Fig. 2. Method 2: (a) SNR = 54 dB, (b) SNR = 12 dB.

Θ, i.e., the eigenvalues of $E_2 + P_2$, is, of course, in general different from the estimate obtained in the previous sections. A further refinement is obtained by conditionally perturbing either E_1 or E_2, depending on which of the two yields the lowest perturbation norm.

In this respect, note that the method of making matrices commute by adding small perturbation matrices to them is not very well posed: if E_1 and E_2 are similarity transformed to $E_1' = TE_1T^{-1}$ and $E_2' = TE_2T^{-1}$, then their eigenvalues stay the same, yet the minimum norm perturbations that must be added to make them commute are different (not equal to TP_1T^{-1} and TP_2T^{-1}), and hence the resulting eigenvalues of $E_1' + P_1'$ and $E_2' + P_2'$ are also different: the solution is dependent on the initial parametrization of the problem.

IV. Simulation Results

To give an indication of the behavior of the methods discussed in the previous section, we devised the following test scenario. In all simulations, the number of sources is $d = 4$, and their angles of incidence are (10°, 25°), (15°, 20°), (20°, 15°), and (25°, 10°), respectively. The sensors are arranged in a square array of $m \times m$ sensors, and are all equal to each other and omnidirectional. Their interdistance is taken to be $\lambda/4$, where λ is the wavelength of the signals. All possible sensor triplets of distance $\lambda/4$ are taken into account, which results in a total of $(m-1)^2$ sensor pairs per dimension. For each algorithm, two signal-to-noise ratios (SNR's) are considered: a) 54 dB, and b) 12 dB. For the first case (54 dB), we took $m = 5$, the number of samples $N = 100$, resulting in X, Y, and Z matrices of size 16×100. A total of 100 test runs are performed. For the 12-dB case we used $m = 10$, $N = 300$, and 400 runs are done.

In the simulation, method 1 is the rotational perturbation method (Section III-A.), method 2 is the additive perturbation method with P_1 and P_2 according to (8), while in method 3 we take either $P_1 = 0$ and P_2 as in (9), or $P_2 = 0$ and P_1 according to much the same equation (mutatis mutandis), depending on which gives the lowest perturbation norm.

The results are displayed in Figs. 1–3, and some statistics are collected in Table I.

While one must be careful with drawing general conclusions from a single example, the following remarks can be made. From Fig. 1(a) the fact that the rotational perturbation method computes the eigenvalues of E_1 and E_2 independently is reflected in the circular shape of the variance clouds. In contrast, the variance clouds in Figs. 2 and 3 exhibit the line structure of the source configuration,

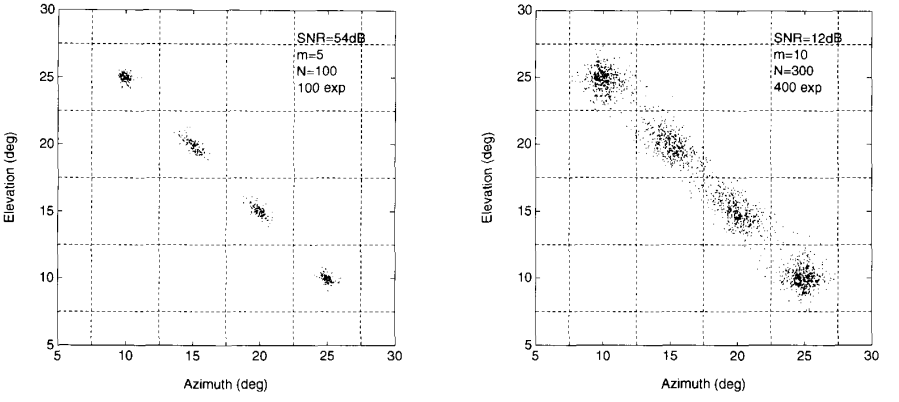

Fig. 3. Method 3: (a) SNR = 54 dB, (b) SNR = 12 dB.

TABLE I
ESTIMATE STATISTICS

	SNR = 54 dB (m = 5, N = 100, 100 runs)				SNR = 12 dB (m = 10, N = 300, 400 runs)			
	Mean (ϕ_i)	Mean (θ_i)	Std (ϕ_i)	Std (θ_i)	Mean (ϕ_i)	Mean (θ_i)	Std (ϕ_i)	Std (θ_i)
Method 1	10.00	24.98	0.2	0.2	10.01	24.96	0.6	1.0
	15.01	20.00	0.5	0.6	14.85	19.95	1.3	1.5
	19.98	15.01	0.5	0.5	20.07	15.08	1.3	1.5
	25.02	9.98	0.2	0.2	25.02	9.99	0.6	0.6
Method 2	9.96	24.99	0.7	0.7	10.11	24.97	0.7	0.8
	14.98	19.97	0.7	0.7	15.08	19.97	0.9	0.9
	19.98	14.99	0.7	0.7	20.03	15.01	0.9	0.9
	25.05	9.92	0.8	0.7	24.90	10.10	0.8	0.7
Method 3	10.02	25.01	0.3	0.3	10.15	24.84	0.8	0.9
	15.03	19.99	0.5	0.5	15.04	19.99	1.2	1.2
	19.94	15.08	0.4	0.4	19.98	15.04	1.1	1.1
	25.01	10.02	0.3	0.3	24.86	10.14	0.8	0.8

which shows that in the additive perturbation methods the azimuth and elevation directions are not treated independently, and that these methods tend to enhance the source configuration in their estimates.

Finally, for a comparison of the proposed new techniques in relation to existing methods we refer to [10], [11]. In these papers a comparison is made between MUSIC, WSF, and suboptimal SSF, of which the pair matching ideas are related to method 3 in this correspondence. The simulations in these papers show that suboptimal SSF performs well in comparison with MUSIC, while the simulations in this correspondence indicate that the two proposed new methods (1 and 2) have a performance comparable to method 3.

REFERENCES

[1] R. Roy, A. Paulraj, and T. Kailath, "ESPRIT—a subspace rotation approach to estimation of parameters of cisoids in noise," *IEEE Trans. Acoust., Speech, Signal Processing*, pp. 1340–1342, 1986.

[2] S. Y. Kung, C. K. Lo, and R. Foka, "A Toeplitz approximation approach to coherent source direction finding," in *Proc. IEEE ICASSP*, 1986.

[3] Y. Hua and T. K. Sarkar, "Matrix pencil method for estimating parameters of exponentially damped/undamped sinusoids in noise," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 35, no. 5, pp. 814–824, 1990.

[4] F. Li and R. J. Vaccaro, "Analytical performance prediction of subspace-based algorithms for DOA estimation," in *Proc. 2nd Int. Workshop SVD and Signal Processing* (Kingston, RI), June 1990, pp. 161–168.

[5] A. J. van der Veen and E. F. Deprettere, "Parallel VLSI matrix pencil algorithm for high resolution direction finding," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 39, no. 2, 1991.

[6] A. J. van der Veen, E. F. Deprettere, and A. L. Swindlehurst, "SVD-based estimation of low rank system parameters," in *Algorithms and Parallel VLSI Architectures*, vol. A, E. F. Deprettere and A. J. van der Veen, Eds. Elsevier, 1991, pp. 203–228.

[7] M. D. Zoltowski and D. Stavrinides, "Sensor array signal processing via a Procrustus rotations based eigenanalysis of the ESPRIT data pencil," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 37, no. 6, 1989.

[8] Y. Hua, "Estimating two-dimensional frequencies by matrix enhancement and matrix pencil," in *Proc. IEEE ICASSP* (Toronto, Canada), 1991, pp. 3073–3076.

[9] P. B. Ober, E. F. Deprettere, and A. J. van der Veen, "Efficient methods to compute azimuth and elevation in high-resolution DOA estimation," in *Proc. IEEE ICASSP* (Toronto, Canada), 1991.

[10] A. L. Swindlehurst and T. Kailath, "2D parameter estimation using arrays with multidimensional invariance structure," in *Proc. 23rd Asilomar Conf. Signals, Syst., Comput.*, 1989, pp. 950–954.

[11] A. L. Swindlehurst and T. Kailath, "Azimuth/elevation direction finding using regular array geometrics," *IEEE Aerosp. Electron. Syst.*, to be published, vol. 29, no. 1, Jan. 1993.

[12] G. Golub and C. F. Van Loan, *Matrix Computations*. Baltimore, MD: Johns Hopkins University Press, 1984.