On-Line Subspace Estimation Using a Schur-Type Method

Jürgen Götze and Alle-Jan van der Veen

Abstract—A recently developed Schur-type matrix approximation technique is applied to subspace estimation. The method is applicable if an upper bound of the noise level is approximately known. The main feature of the algorithm is that updating and downdating is straightforward and efficient and that the subspace dimension is elegantly tracked as well.

I. INTRODUCTION

Subspace estimation plays an important role in many modern high-resolution parameter estimation algorithms. Examples are the direction-of-arrival estimation problem, harmonic retrieval, system identification, and the (total) least squares solution of systems of overdetermined linear equations. In a typical application, we have a $m \times n$ measured data matrix $X = \hat{X} + N$, where $\hat{X}$ is a rank-deficient noise-free matrix based on the signals of interest, and N represents additive noise. The objective is to estimate the column span $\mathcal{R}(\hat{X})$ of $\hat{X}$, assuming that the noise is sufficiently "small," say $\|N\| \leq \gamma$. For a given threshold γ, an estimate of $\hat{X}$ is obtained by solving

$$\min_{\hat{X}} \operatorname{rank}(\hat{X}) \quad \text{s.t.} \quad \|X - \hat{X}\| \leq \gamma. \tag{1}$$

Usually, the Frobenius norm is chosen. In this norm, the solution $\hat{X}$, which also minimizes $\|X - \hat{X}\|$, is given by the truncated singular value decomposition (SVD) of X; if

$$
\begin{aligned}
X &= U \Sigma V^H \\
&= [U_1 \quad U_2] \begin{bmatrix} \Sigma_1 & \\ & \Sigma_2 \end{bmatrix} \begin{bmatrix} V_1^H \\ V_2^H \end{bmatrix}; \\
&(\Sigma_1)_{ii} > \gamma, \ (\Sigma_2)_{ii} \leq \gamma
\end{aligned}
\tag{2}
$$

is the SVD of X, then a solution of (1) is $\hat{X} = U_1 \Sigma_1 V_1^H$. The "signal subspace," or principal subspace, is given by $\mathcal{R}(U_1)$. Its dimension is equal to the number of singular values of X that are larger than, say, γ, d. A number of alternative decompositions have been developed to estimate $\mathcal{R}(U_1)$ in a computationally more efficient way. Examples are the URV decomposition [1] and the rank revealing QR (RRQR) decomposition [2], [3]. The complexity of these algorithms can be reduced further by considering updating techniques [4], [5], although at this point, norm bounds on the estimation error hold only asymptotically. Furthermore, a condition estimation is usually required for determining d, which might not be very accurate or computationally elegant, and requires a sufficiently large gap between the signal and noise singular values.

In this correspondence, we take a different route and propose to compute subspace estimates based on the *matrix 2-norm* (i.e., the largest singular value of the matrix). The 2-norm is strong enough to give accurate subspace estimates, and the subspace dimension is still

equal to d. However, approximants in this norm are considerably easier to compute, essentially by a Schur-type method that boils down to a hyperbolic QR (HQR) factorization at a complexity of order $1/2m^2 n$ rotations. Updating and downdating the HQR is very straightforward. The norm bound (1) is exact and still holds when the data is not stationary, for example, when the number of sources is changing.

The mathematical details of this Schur method are in a separate paper [6]. The purpose of this correspondence is to demonstrate its relevance to signal processing applications. To this end, we consider the problem of direction finding. It is shown by computer simulations that even under critical circumstances (low SNR's, spatially close signals), a particular choice of the subspace estimates obtained by the Schur method is close to the true signal subspace as defined by the principal singular vectors.

II. SCHUR-TYPE METHOD FOR SUBSPACE ESTIMATION

A. Background

The Schur-type subspace estimation (SSE) method is based on an implicit factorization of

$$X X^H - \gamma^2 I =: B B^H - A A^H \tag{3}$$

for matrices A, B of minimal dimensions. In (3), the spectrum of $X X^H$ is shifted such that its small eigenvalues (corresponding to the small singular values of X) become negative and can be separated. A and B are not unique, but their dimensions are fixed; if d is the number of singular values of X that are larger than γ, and none of them are equal to γ, then B has d columns, and A has $m - d$ columns. In addition, it is shown in [6] that for every such factorization, there exists a matrix $E : d \times n$ such that $\hat{X} = BE$ satisfies (1). Thus, $\mathcal{R}(B)$ is a valid 2-norm principal subspace estimate. It is also shown in [6] that *all* such estimates are given by $\mathcal{R}(B - A S_L)$, where S_L is any $(m - d \times d)$-matrix satisfying $\|S_L\| < 1$. Some choices for S_L may lead to a smaller approximation error $\|X - \hat{X}\|$. We propose one particular choice later.

To describe the hyperbolic factorization, we need the following definitions. A *signature matrix* $\tilde{J}$ is a diagonal matrix with diagonal entries equal to $+1$ or -1. A matrix $\tilde{\Theta}$ is *J-unitary* if it satisfies

$$
\begin{aligned}
(\tilde{\Theta})^H \tilde{J}_1 \tilde{\Theta} &= \tilde{J}_2, \\
\tilde{\Theta} \tilde{J}_2 (\tilde{\Theta})^H &= \tilde{J}_1
\end{aligned}
\tag{4}
$$

for signature matrices $\tilde{J}_1$, $\tilde{J}_2$. Using permutations, the entries of a signature matrix can be sorted into a positive and a negative block, and we partition Θ accordingly as

$$
\begin{aligned}
\Theta &= \begin{bmatrix} \Theta_{11} & \Theta_{12} \\ \Theta_{21} & \Theta_{22} \end{bmatrix}, \\
J_1 &= \begin{bmatrix} I & \\ & -I \end{bmatrix} \\
&= J_2
\end{aligned}
\tag{5}
$$

(for identity matrices of appropriate sizes). We will denote an unsorted signature matrix by a tilde.

B. The HQR Algorithm

The main step in the SSE method is the implicit computation of the factorization (3) via a HQR factorization. Suppose that none of the

TABLE I
SIX DIFFERENT CASES OF ROTATION COMPUTATION

1. $\tilde{J}_1 = \begin{bmatrix} +1 & \\ & -1 \end{bmatrix}$, $|l| > |x|$ $\Rightarrow$ $\tilde{J}_2 = \begin{bmatrix} +1 & \\ & -1 \end{bmatrix}$, $\tilde{\theta} = \begin{bmatrix} 1 & -s \\ -s^* & 1 \end{bmatrix} \dfrac{1}{c}$

2. $\tilde{J}_1 = \begin{bmatrix} +1 & \\ & -1 \end{bmatrix}$, $|l| < |x|$ $\Rightarrow$ $\tilde{J}_2 = \begin{bmatrix} -1 & \\ & +1 \end{bmatrix}$, $\tilde{\theta} = \begin{bmatrix} -s^* & 1 \\ 1 & -s \end{bmatrix} \dfrac{1}{c}$

3. $\tilde{J}_1 = \begin{bmatrix} -1 & \\ & +1 \end{bmatrix}$, $|l| < |x|$ $\Rightarrow$ $\tilde{J}_2 = \begin{bmatrix} +1 & \\ & -1 \end{bmatrix}$, $\tilde{\theta} = \begin{bmatrix} -s^* & 1 \\ 1 & -s \end{bmatrix} \dfrac{1}{c}$

4. $\tilde{J}_1 = \begin{bmatrix} -1 & \\ & +1 \end{bmatrix}$, $|l| > |x|$ $\Rightarrow$ $\tilde{J}_2 = \begin{bmatrix} -1 & \\ & +1 \end{bmatrix}$, $\tilde{\theta} = \begin{bmatrix} 1 & -s \\ -s^* & 1 \end{bmatrix} \dfrac{1}{c}$

5. $\tilde{J}_1 = \begin{bmatrix} +1 & \\ & +1 \end{bmatrix}$ $\Rightarrow$ $\tilde{J}_2 = \begin{bmatrix} +1 & \\ & +1 \end{bmatrix}$, $\tilde{\theta} = \begin{bmatrix} c^* & -s \\ +s^* & c \end{bmatrix}$

6. $\tilde{J}_1 = \begin{bmatrix} -1 & \\ & -1 \end{bmatrix}$ $\Rightarrow$ $\tilde{J}_2 = \begin{bmatrix} -1 & \\ & -1 \end{bmatrix}$, $\tilde{\theta} = \begin{bmatrix} c^* & -s \\ +s^* & c \end{bmatrix}$

The rotation parameters s and c of the different cases follow from the equation $[l' \quad x']\tilde{\theta} = [* \quad 0]$ as

case 1, 4: $s = x'/l'$, $c = (1 - |s|^2)^{1/2}$

case 2, 3: $s = l'/x'$, $c = (1 - |s|^2)^{1/2}$

case 5, 6: $s = x'(|l'|^2 + |x'|^2)^{-1/2}$, $c = (1 - |s|^2)^{1/2}$.

singular values of X are equal to γ. Then, there exists a factorization

$$\begin{bmatrix} \overset{+}{\gamma I}_{m \times m} & \overset{-}{X} \end{bmatrix} \tilde{\Theta} = \begin{bmatrix} \overset{+/-}{L} & \overset{+/-}{0_{m \times n}} \end{bmatrix} \qquad (6)$$

where L is a *lower* triangular $m \times m$ matrix, and $\tilde{\Theta}$ is $(\tilde{J}_1, \tilde{J}_2)$-unitary. Here, J_1 is given on the outset as

$$J_1 = \begin{bmatrix} I_{m \times m} & \\ & -I_{n \times n} \end{bmatrix}$$

which associates a positive signature with the columns of γI and a negative signature with the columns of X. $\tilde{J}_2$ is an unsorted output signature, which is to be determined along with L. After sorting the columns of $[L \quad 0]$ according to their signatures, one obtains

$$[\overset{+}{\gamma I} \quad \overset{-}{X}]\Theta = [\overset{+}{A'} \quad \overset{-}{B'}];$$
$$A' = [\underbrace{A}_{m-d} \quad \underbrace{0}_{d}],$$
$$B' = [\underbrace{B}_{d} \quad \underbrace{0}_{n-d}] \qquad (7)$$

where Θ is a matrix that leaves the "J-norm" invariant so that (3) follows from (7). Thus, A is equal to the columns of L that have a positive signature, and B consists of the columns with a negative signature. Θ is obtained by applying the same permutations to $\tilde{\Theta}$.

The algorithm to obtain (6) is similar in structure to an ordinary LQ factorization algorithm using elementary (Givens) rotations, except that we have to keep track of the signature of every column, and the type of elementary rotation (circular or hyperbolic) is dependent on these signatures. Thus, $\tilde{\Theta}$ is computed by subsequently zeroing every entry x_{ik} of X in a column-wise top-down fashion: $\tilde{\Theta} = \prod_{k=1,\cdots,n}^{\rightarrow} \prod_{i=1,\cdots,m}^{\rightarrow} \tilde{\Theta}^{(i,\,k)}$. At stage (i, k), suppose we are acting on $[L' \quad X']$ (with L' lower triangular):

$$[L' \quad X']\tilde{\Theta}^{(i,\,k)} = [L'' \quad X''].$$

$\tilde{\Theta}^{(i,\,k)}$ is an embedding of an elementary hyperbolic rotation $\tilde{\theta}^{(ik)}$ of size 2×2 into a plane rotation, where $\tilde{\theta}^{(ik)}$ is such that entry x'_{ik} of X' is zeroed against entry l'_{ii} of L':

$$[\overset{+/-}{l'_{ii}} \quad \overset{+/-}{x'_{ik}}]\tilde{\theta}^{(ik)} = [\overset{+/-}{l''_{ii}} \quad \overset{+/-}{0}].$$

This will only change column i of L' and column k of X', as well as the signatures associated with these columns. L'' remains lower triangular because entries 1 to $i-1$ of this column of X' are already zero.

At the elementary level, we have to compute $\tilde{\theta}$ such that $[l' \quad x']\tilde{\theta} = [l'' \quad 0]$, where the input signature $\tilde{j}_1$ is given along with l' and x'. Using the fact that $\tilde{\theta}$ is $(\tilde{j}_1, \tilde{j}_2)$-unitary, l'' and its signature $(\tilde{j}_2)_{11}$ follows from $[l' \quad x']\tilde{j}_1[l' \quad x']^H = |l''|^2(\tilde{j}_2)_{11}$. Altogether, this gives six possible cases to consider. These cases are summarized in Table I. The case where $|l'| = |x'|$ is degenerate and has to be ruled out. This case occurs if one of the singular values of X, or of its principal minors, is precisely equal to γ.

Since the HQR algorithm is similar to the QR factorization, it can be implemented on an upper triangular processor array. The mere difference is that the processor cells must be able to perform circular as well as hyperbolic rotations and that each column has a signature associated with it. Fig. 1 shows the corresponding signal flow graph.

C. Subspace Estimators

As mentioned in Section II-A, all 2-norm signal subspace estimates are given by the range of $U_{SSE} = B - AS_L$, where S_L is any $m - d \times d$-matrix with $\|S_L\| \leq 1$. A particularly simple subspace estimate is obtained by taking $S_L = 0$, which gives
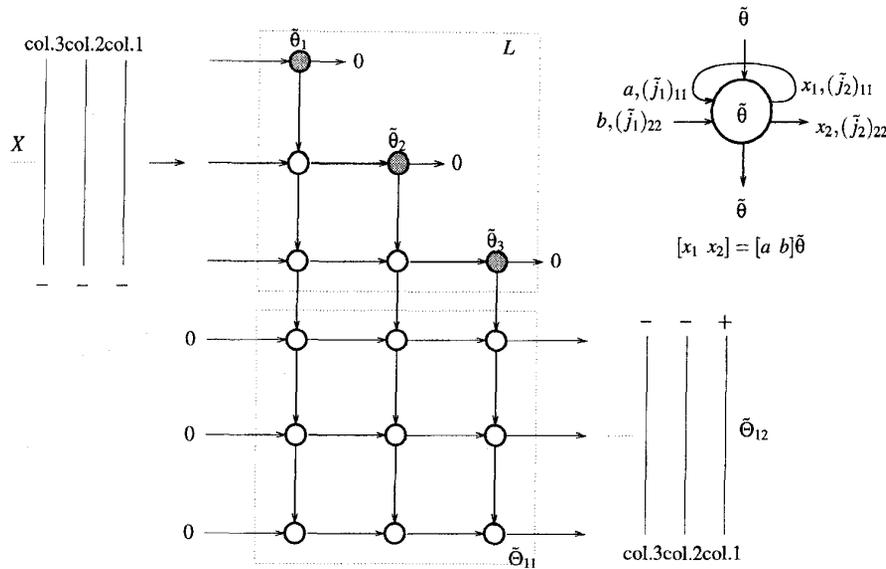
$$U_{SSE1} = B \qquad (8)$$

Fig. 1. Signal flow graph of the Schur algorithm. Associated with every matrix entry is also its signature $(+ \text{ or } -)$. L contains a permutation of $[A \ B]$ and is initialized by γI. $\tilde{\Theta}_{11}$ is initialized by I. The shaded processors compute rotation parameters per Table I.

and will be called the Schur subspace estimate-1 (SSE-1) for reference. Despite its simplicity, this subspace estimate gives quite good results when the conditions are noncritical (high enough SNR, well-separated sources) so that there is a large gap between σ_d and σ_{d+1}. In more critical cases, the estimate U_{SSE1} exhibits biased results since the estimation error is not minimized but only kept smaller than γ. In particular, if $N = 0$ (all nonzero singular values of X are larger than γ) so that $\hat{X} = X$ would be an approximant with zero error, the choice $S_L = 0$ in general gives an approximant $\hat{X} \neq X$. It is possible, however, to compute the S_L that leads to $\hat{X} = X$ in this case [6]. The corresponding subspace estimate is called SSE-2 and is given by

$$U_{SSE2} = B - A \cdot [I_{m-d} \quad 0] \Theta_{11}^{-1} \Theta_{12} \begin{bmatrix} I_d \\ 0 \end{bmatrix}. \tag{9}$$

Simulation examples indicate that this subspace estimate is usually close to the principal subspace. It is unbiased in the sense that $\mathcal{R}(U_{SSE2}) \subset \mathcal{R}(X)$ [16], which is something that is not true for U_{SSE1}. Although this is not clear at first sight, U_{SSE2} can be computed and updated efficiently at about four times the complexity of U_{SSE1} [7].

D. Updating and Downdating the Subspace Estimate

Because the HQR factorization (6) is computed column-by-column, it can be updated with new columns x of X in a straightforward manner. It suffices to continue with x, giving it a negative signature. Downdating (removing a column x of X) is simple as well. It consists of an updating step by the column of X that has to be removed but now with a positive signature. This is because the HQR factorization provides an implicit minimal factorization of $\gamma^2 I - XX^H = L(\tilde{J}_2)_{11} L^H$, and after downdating, we should have an implicit factorization of $\gamma^2 I - XX^H + xx^H$. This updating/downdating scheme gives a sliding window adaptive subspace tracking algorithm. Downdating with an exponential forgetting factor (scaling X by $\lambda \leq 1$) is also possible but, in its exact form, is less efficient. This is because simply scaling the result L is not sufficient as this will also scale γ to $\gamma\lambda$. To restore γ, the scaling of L by λ has to be followed by an update by $(1 - \lambda^2)^{1/2} \gamma I_m$ (with a positive signature).

This involves the processing of m columns, rather than 1, but we can instead do a *stochastic* update with a single column n with covariance $E(nn^H) = I_m \cdot (1 - \lambda^2)\gamma$, generated by a white noise process. Note that we could use the same technique to adjust γ to any other value. It is possible to change the noise threshold "on-the-fly." One application would be to devise an iterative or adaptive scheme to find a tight lower bound for the noise threshold, e.g., based on the residual power after projection of the data vectors on the estimated signal subspace.

E. Choice of γ

Suppose that the noise $N = [n_i]$ added to $\hat{X}$ is white and has a known variance σ: $E(nn^H) = \sigma^2 I_m$. It is standard and straightforward to show that $\sigma_{d+1} \to \sigma\sqrt{n} (n \to \infty)$ so that a tight value for the threshold γ would be $\gamma_0 = \sigma\sqrt{n}$. However, for finite data lengths n, σ_{d+1} is a stochastic variable that is typically larger than $\sigma\sqrt{n}$; therefore, this threshold is too low. To achieve $\gamma > \sigma_{d+1}$ with some level of confidence, we model $\sigma_{d+1} = (1 + \alpha) \cdot \gamma_0$, where $\alpha > 0$ is a finite-sample correction term to the limiting value of σ_{d+1}. In a first-order approximation (small d), α is only dependent on the noise matrix N and not on the signal matrix $\hat{X}$; therefore, it can be modeled as a function of n and the number of sensors m only. Empirically, we obtain that α has a mean value and standard deviation given by, respectively

$$\overline{\alpha} \simeq 0.5 \cdot \sqrt{m} \cdot n^{-2/5},$$

$$s_\alpha \simeq \tfrac{1}{3} n^{-0.46}.$$

Therefore, a suitable value for γ such that $\gamma > \sigma_{d+1}$ with probability larger than 0.998 is

$$\gamma = (1 + \overline{\alpha} + 3s_\alpha)\gamma_0. \tag{10}$$

III. SIMULATION RESULTS

To assess the performance of the SSE technique, we consider a standard direction finding problem. Suppose that we have a linear array of m $\lambda/2$-spaced omnidirectional sensors, receiving d sinusoidal signals with directions of arrival (DOA's) ϕ_k, $k = 1, , \cdots, d$. A total number of n snapshots is taken, producing an $m \times n$ data matrix X that satisfies the model $X = \mathcal{A}S + N$. Here, $\mathcal{A} = \mathcal{A}(\phi_1, \cdots, \phi_d)$:
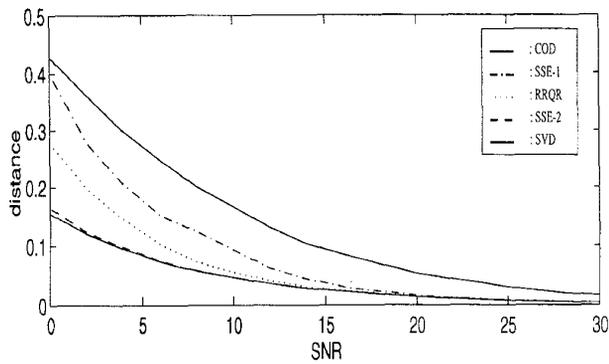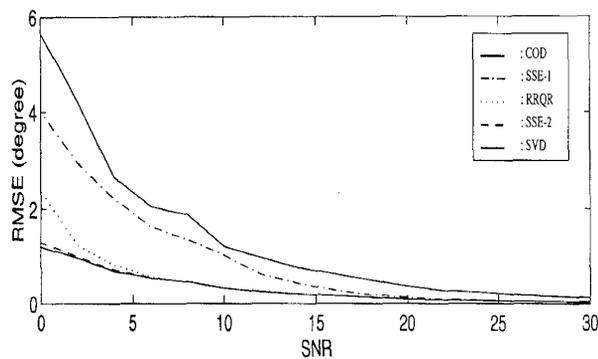
Fig. 2. Experiment 1: (a) Averaged subspace angle between the estimated and the "true" subspace and (b) RMS error of the estimated DOA's as a function of the SNR. Number of correct estimates of d: Schur = 100%, COD = 97%, RRQR = 64%.

Fig. 3. Experiment 2: Same as Fig. 2 as function of $\phi_2 - \phi_1$. Number of correct estimates of d: Schur = 100%, COD = 95%, RRQR = 69%.

$m \times d$ is the array response matrix that is a function of the DOA's, $S : d \times n$ contains the n samples of the d source signals, and N contains samples of white additive i.i.d. noise sources with variance $\sigma^2 I$, independent of the signals. Given X, the ϕ_k are to be estimated.
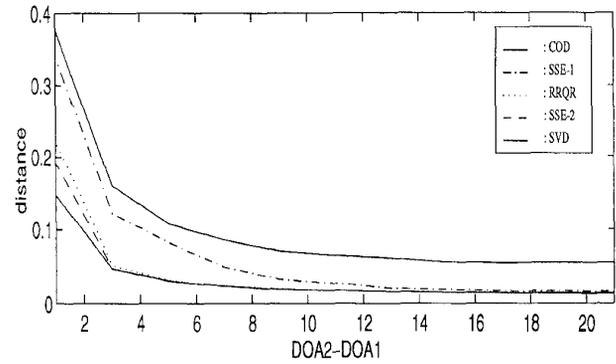
The ESPRIT algorithm for estimating the DOA's [8] works in two steps. The first step is to estimate a basis for the signal subspace $\mathcal{R}(\mathcal{A})$, which is usually taken to be the d principal left singular vectors of X, i.e., $U_S = U_1$. We will compare this with the Schur-based subspace estimates ($U_S = U_{SSE1}$ and $U_S = U_{SSE2}$) and estimates based on the complete orthogonal decomposition (COD) [9] and RRQR [2], [3]. Once the signal subspaces are estimated, the DOA's are obtained from a certain eigenvalue decomposition based on these subspaces.

The quality of subspace estimates can be measured by computing the distance between the estimated and the true subspaces, as defined in [9]. We conducted three sets of experiments: varying SNR's, varying source gaps, and varying source correlations. Figs. 2–4 show the averaged distance between the estimated and true subspace (a) and the RMSE of the resulting estimated DOA's (b). These statistics are computed from 100 repetitions of each particular experiment.
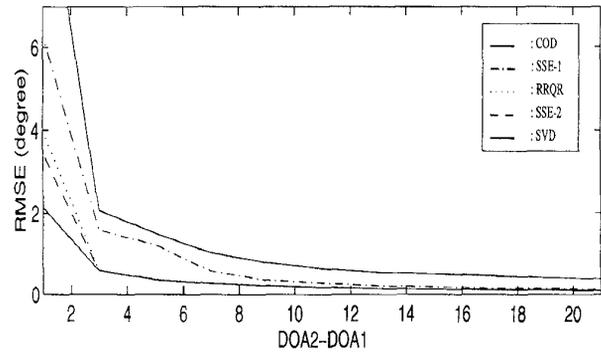
Experiment 1: $n = 30$; $m = 6$; $d = 2$ uncorrelated signals; DOA's are $\phi_1 = 20°$, and $\phi_2 = 40°$. The SNR is varied from 0–30 dB in steps of 2 dB (see Fig. 2).

Experiment 2: $n = 30$; $m = 6$; $d = 2$ uncorrelated signals; SNR = 20 dB; DOA's are $\phi_1 = 20°$, and ϕ_2 is varied from 21–41° in steps of 2° (see Fig. 3).

Experiment 3: $n = 30$; $m = 6$; $d = 2$ correlated signals; SNR = 20 dB; DOA's are $\phi_1 = 20°$, and $\phi_2 = 40°$. The correlation coefficient

between the two signals is varied from 0.0–0.9 in steps of 0.1 (see Fig. 4).

For the Schur-type methods, the threshold γ is selected as in (10); this produced the correct estimate of the number of sources in almost all cases. The experiments show that the SSE-2 performs almost as well as the truncated SVD method. The simpler SSE-1 is applicable for noncritical scenarios.

For the COD and the RRQR, the value of γ in (10) is often too tight to produce the correct principle subspace dimension, especially when the gap between σ_d and σ_{d+1} is narrow (closely spaced signals, low SNR, or highly correlated signals). In these cases, we had to increase the value of γ during the experiments; otherwise, the results of COD and RRQR are worse than shown in the figures. The percentage of correct estimates of the number of sources is listed in the figure captions. In the case of incorrect dimension estimations, usually, the number of sources is overestimated, and we included the results of these cases in the figures, assuming that the "true" DOA's can be detected. Of course, the number of wrong dimension estimates can be significantly reduced for COD and RRQR by using more advanced condition estimators. However, this is an additional computational effort that is not required for the Schur method.

IV. CONCLUSIONS

In the SSE method, *a priori* knowledge of the noise level is assumed in the selection of the threshold γ. Such information is often available in practice at little cost. For example, for bursty signals, one can use periods in which the signals are absent to measure the noise level. Another possibility is to measure the average power over a wide enough frequency band, which, before sampling, requires not
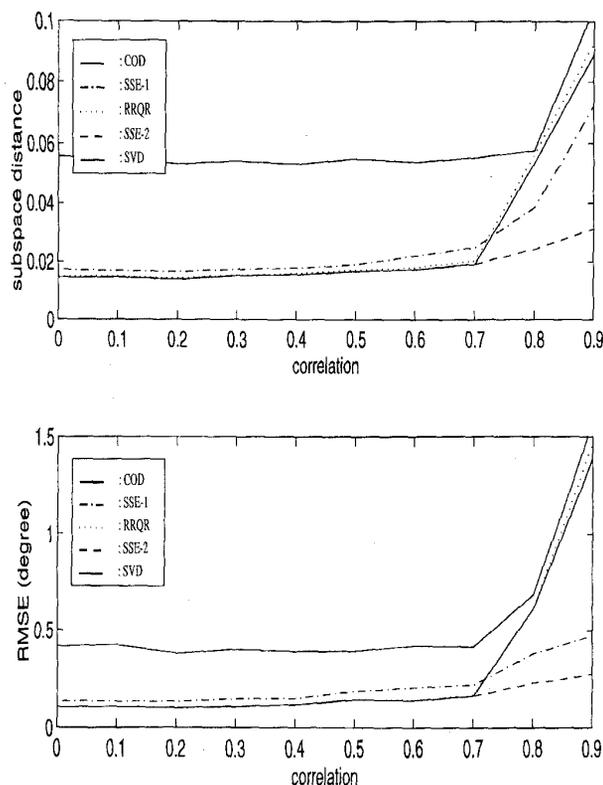
Fig. 4. Experiment 3: Same as Fig. 2 as function of the correlation coefficient between the two signals. Number of correct estimates of d: Schur $= 99.6\%$, COD $= 89\%$, RRQR $= 62\%$.

much more than an analog bandpass filter and an RMS meter. The precise value of γ is not critical as long as it is above the noise level and below the smallest singular value due to the signals. As we mentioned at the end of Section II-D, the noise power can be measured as well from the residual power after a first estimate of the signal subspace has been obtained, and γ can be made more accurate iteratively or adaptively [7]. For direction-of-arrival applications, additional symmetry assumptions on the antenna array structure can be exploited to lead to more accurate DOA estimates as well, viz. the "unitary Schur ESPRIT method" [10]. The Schur method provides a *generalized* subspace estimate, which is similar to a generalized SVD (or quotient SVD); if the noise is not spatially white but has a known covariance matrix NN^H, then this information can be incorporated without any additional effort by computing and updating a HQR factorization of $[\gamma L_N \quad X]$ rather than $[\gamma I \quad X]$, where L_N is the Cholesky factor of NN^H.

In this correspondence, we have limited our comparisons to methods based on a given upper bound on the noise level, all of which require $O(m^2)$ per update. A host of other methods are based on knowledge of the subspace dimension d, and some of these exploit additional statistical assumptions on the noise as well. The resulting deflated subspace methods (e.g., Karasalo's method [11] and the PAST method [12]) only track the signal subspace and require $O(md^2)$ per update, which is interesting if $d^2 < m$. The accuracy of these methods is quite good; however, the existence of parallel implementations is unclear at this point. It might be possible to derive a deflated Schur-type method as well.

REFERENCES

[1] G. W. Stewart, "An updating algorithm for subspace tracking," *IEEE Trans. Signal Processing*, vol. 40, pp. 1535–1541, June 1992.

[2] L. V. Foster, "Rank and null space calculations using matrix decomposition without column interchanges," *Lin. Alg. Appl.*, vol. 74, pp. 47–71, 1986.

[3] T. F. Chan, "Rank revealing QR factorizations," *Lin. Alg. Appl.*, vol. 88, 89, pp. 67–82, 1987.

[4] C. H. Bischof and G. M. Schroff, "On updating signal subspaces," *IEEE Trans. Signal Processing*, vol. 40, pp. 96–105, Jan. 1992.

[5] M. Moonen, P. Van Dooren, and J. Vandewalle, "An SVD updating algorithm for subspace tracking," *SIAM J. Matrix Anal. Appl.*, vol. 13, no. 4, pp. 1015–1038, 1992.

[6] A. J. van der Veen, "A Schur method for low-rank matrix approximation," *SIAM J. Matrix Anal. Appl.*, Jan. 1996.

[7] ——, "Updating the Schur subspace estimator," submitted to *IEEE Trans. Signal Processing*, Sept. 1995.

[8] 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*, vol. ASSP-34, no. 5, pp. 1340–1342, 1986.

[9] G. H. Golub and C. F. van Loan, *Matrix Computations*, 2nd ed. Baltimore, MD: The John Hopkins University Press, 1989.

[10] J. Götze, M. Haardt, and J. A. Nossek, "Subspace estimation using unitary Schur-type methods," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing*, Detroit, 1995.

[11] I. Karasalo, "Estimating the covariance matrix by signal subspace averaging," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-34, pp. 8–12, 1986.

[12] B. Yang, "Projection approximation subspace tracking," *IEEE Trans. Signal Processing*, vol. 43, pp. 95–107, Jan. 1995.

[13] C. H. Bischof and P. C. Hansen, "Structure preserving and rank-revealing QR-factorizations," *SISSC*, vol. 12, pp. 1332–1350, 1991.

Robust Adaptive Beamforming via Target Tracking

Saeed Gazor, Sofiène Affes, and Yves Grenier

Abstract— **The proposed robust beamformer adaptively self-corrects and tracks desired-source location errors or variations regardless of the eigenstructure of the input correlation matrix, of the array shape and nature of the noise. It only requires an order of computations of the sensors number and performs nearly as well as MUSIC in localization.**

I. INTRODUCTION

Adaptive beamforming such as the GSC [1] is very sensitive to source location errors. In the case of location uncertainty (e.g., mobile