A NEW ADAPTIVE EIGENDECOMPOSITION ALGORITHM BASED ON A FIRST-ORDER PERTURBATION CRITERION

Benoit Champagne

INRS-Télécommunications, Université du Québec,
3 Place du Commerce, Verdun, Québec, Canada H3E 1H6.

ABSTRACT

In this paper, we present a new adaptive eigendecomposition algorithm that can be used for on-line high-resolution spectral/spatial analysis. The formulation of the algorithm is based on the interpretation of the correction term in the recursive update of the data covariance matrix estimate as a perturbation term, with the forgetting factor playing the role of a perturbation parameter. Following this interpretation, a first-order perturbation analysis is made to obtain a new recursion expressing the eigenstructure estimate of the true data covariance matrix at time k, in terms of the eigenstructure estimate at time $k-1$. The resulting algorithm can be realized by means of M linear combiners with non-linear weight-vector adaptation equations, where M is the signal-subspace dimensionality. Moreover, it does not require an explicit Gram-Schmidt orthogonalization step. Comparative simulation results for narrow-band array data indicate very good performance of the proposed algorithm.

I. INTRODUCTION

Signal-subspace algorithms, which are based on the eigendecomposition of the data covariance matrix, have been applied successfully to both temporal and spatial-domain high-resolution spectral analysis [1]. It is common practice to implement these algorithms in a batch mode, using a sample covariance matrix obtained by collecting observation vectors over a sufficiently long time interval. This approach, which relies on the assumption of stationarity of the observed data, can not be used in situations where signal characteristics change with time. In this case, the application of signal-subspace algorithms requires repeated eigendecomposition of an updated sample covariance matrix, a task which is generally prohibitive.

Various adaptive eigendecomposition algorithms have been suggested in the past to overcome this difficulty. Most of these can be classified into two distinct families. In the first family (e.g. [2]), the determination of the signal subspace is formulated as a constrained optimization problem which is solved via a stochastic gradient search over time, using a recursive estimate of the array covariance matrix to evaluate the gradient vector. In the second family (e.g. [3]), variations and extensions of Bunch's rank one eigenstructure updating algorithm are used to update the eigenstructure of the sample covariance matrix of the data at each iteration.

In this paper, we present a new adaptive eigendecomposition algorithm that is based on an alternate formulation of the problem. In this formulation, the correction term in the recursive update of the data covariance matrix estimate is interpreted as a perturbation term, with the (small) forgetting factor playing the role of a perturbation parameter. Following this interpretation, a first-order perturbation analysis is made to obtain a recursion expressing the eigenstructure estimate at time k, in terms of the eigenstructure estimate at time $k-1$. The resulting algorithm can be realized by means of M linear combiners with non-linear weight-vector adaptation equations, where M is the signal-subspace dimensionality. Moreover, it does not require an explicit Gram-Schmidt orthogonalization step. Simulation results for narrow-band array data indicate improved performance of the proposed algorithm when compared to the family of adaptive algorithms proposed in [2].

The paper is organized as follows. In Section II, we describe the data model and we formulate the adaptive eigendecomposition of the data covariance matrix as a perturbation problem. The first-order perturbation analysis along with the resulting algorithm are discussed in Section III. Finally, comparative simulation results are presented in Section IV.

II. A PERTURBATION CRITERION

Let $x(k)$ denote the complex L-dimensional data vector observed at time k. In time series analysis, $x(k)$ would consist of the signal samples over a given frame of length L, while in array processing, $x(k)$ could be the narrow-band output of an L-sensor array. We assume that $x(k)$ consists of the linear superposition of M source signals over a background noise. More precisely, let

$$x(k) = A(k)s(k) + n(k) \tag{1}$$

where s(k) is a complex M-dimensional signal process, $A(k)$ is a $L \times M$ transmission matrix, and $n(k)$ is a complex L-dimensional background noise process. We assume that $s(k)$ and $n(k)$ are zero-mean, uncorrelated random processes with covariance matrices at time k given by

$$R_s(k) = E[s(k)s^H(k)] \tag{2}$$

$$R_n(k) = \sigma_n^2(k)\, I_L \tag{3}$$

where the superscript H denotes complex conjugate transposition, $\sigma_n^2(k)$ is the noise variance at time k and I_L is the $L \times L$ identity matrix. From these assumptions, the data covariance matrix satisfies

$$R_x(k) = A(k)\, R_s(k)\, A^H(k) + \sigma_n^2(k)\, I_L. \tag{4}$$

Let $\lambda_i(k)$ and $q_i(k)$, $i = 1, \ldots, L$, denote the eigenvalues and corresponding orthonormalized eigenvectors of the covariance matrix $R_x(k)$, with the eigenvalues arranged in descending order. It is well known that if the rank of $A(k)$ is equal to

0-7803-0532-9/92 $3.00

M, then

$$\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_M > \lambda_{M+1} = \cdots = \lambda_L = \sigma_n^2 \qquad (5)$$

$$A^H Q_n(k) = 0 \qquad (6)$$

where 0 denotes the $M \times (L-M)$ zero matrix and

$$Q_n(k) = [q_{M+1}, q_{M+2}, \ldots, q_L]. \qquad (7)$$

Hence, the number of source signals is given indirectly by the multiplicity of the smallest eigenvalue, while the column span of A is identical to that of the matrix

$$Q_s(k) = [q_1, q_2, \ldots, q_M]. \qquad (8)$$

The span of $Q_s(k)$ and $Q_n(k)$ are appropriately referred to as the signal-subspace and the noise-subspace, respectively.

Signal-subspace algorithms such as MUSIC [1] use the information embedded in the eigendecomposition of the data covariance matrix to achieve high-resolution spectral analysis of the observed data. Practical implementation of these algorithms are usually based on batch estimation of the data covariance matrix, as in

$$\hat{R}_x = \frac{1}{K} \sum_{k=1}^{K} x(k) x^H(k), \qquad (9)$$

followed by eigendecomposition of $\hat{R}_x$. In a stationary environment, the performance of the signal-subspace algorithms will usually improve as K is increased.

In many applications, however, the observed data can only be considered stationary over a limited time interval due to changes in the characteristics of the signal generation mechanisms. In array processing, for instance, one might be observing plane wave signals whose directions of propagation are changing with time. In these situations, a better estimate of the data covariance matrix is given by

$$\hat{R}_x(k) = (1-\alpha)\, \hat{R}_x(k-1) + \alpha\, x(k)\, x^H(k) \qquad (10)$$

where the parameter α $(0 < \alpha < 1)$ is used to control the memory of the estimate. In this case, a major drawback of the above batch approach is that it requires repeated eigendecomposition of the time-varying sample covariance matrix $\hat{R}_x(k)$, a task which is computationally very expensive.

As indicated earlier, various adaptive eigendecomposition algorithms have been suggested in the past to overcome this difficulty. Rather than performing a complete eigendecomposition of $\hat{R}_x(k)$ at each time iteration, these algorithms attempt to reduce the computational load by recursively updating the current eigendecomposition. Naturally, some approximations are required to make the recursion simple and computationally efficient. These algorithms can be grouped into two distinct families, depending upon the approach used to derive the recursion. In the following paragraphs, we present an alternate formulation of the adaptive eigendecomposition problem that will lead to a new class of algorithms.

Let us rewrite the updating equation (10) for $\hat{R}_x(k)$ as

$$\hat{R}_x(k) = \hat{R}_x(k-1) + \varepsilon\, [x(k)x^H(k) - \hat{R}_x(k-1)] \qquad (11)$$

where ε now replaces α. For ε small (compared to 1), the correction term $\varepsilon[x(k)x^H(k) - \hat{R}_x(k-1)]$ in (11) can be viewed as a perturbation of $\hat{R}_x(k-1)$. Since $\hat{R}_x(k)$ (11) is a convergent power series in ε around $\hat{R}_x(k-1)$, the theory of perturbation of Hermitian matrices [4] asserts that the eigenvalues and eigenvectors of $\hat{R}_x(k)$ (if properly selected) can be expanded in power series in ε, convergent for ε small, around the eigenvalues and eigenvectors of $\hat{R}_x(k-1)$. This theory also provides methods for determining these power series. Hence, if we knew the exact eigendecomposition of $\hat{R}_x(k-1)$, it would be possible to express the eigenvalues and eigenvectors of $\hat{R}_x(k)$ as power series in ε around the corresponding eigenvalues and eigenvectors of $\hat{R}_x(k-1)$, subject to appropriate convergence restrictions. By truncating these power series at a prescribed power of ε, we would obtain a closed form recursion for updating the eigendecomposition.

While the above argument is at the basis of the mathematical derivations presented in this paper, some modifications are needed before it lends itself to the development of a new adaptive eigendecomposition algorithm. Indeed, we generally do not know the exact eigendecomposition of $\hat{R}_x(k-1)$ (and even if we knew it, application of the proposed approach would lead to an approximate eigendecomposition at time k). This technical difficulty can be overcome as follows. Let $\gamma_i(k)$ and $u_i(k)$ denote the desired estimates of the true eigenvalues $\lambda_i(k)$ and corresponding eigenvectors $q_i(k)$ of $R_x(k)$. Rather than seeking estimates which are exact eigenvalues and eigenvectors of $\hat{R}_x(k)$ (11), we require that for every values of k,

$$\hat{R}_x(k)u_i(k) = \gamma_i(k)u_i(k) + O(\varepsilon^{n+1}) \qquad (12)$$

$$u_i^H(k)u_j(k) = \delta_{ij} + O(\varepsilon^{n+1}) \qquad (13)$$

where n is the highest power of ε to be included in the perturbative series, $O(\varepsilon^{n+1})$ represents an error term of order ε^{n+1}, and δ_{ij} is the Kronecker delta. For example, when $n = 1$, (12)-(13) state that $\gamma_i(k)$ and $u_i(k)$ are eigenvalues and orthonormalized eigenvectors of $\hat{R}_x(k)$, respectively, up to error terms of order ε^2.

At this point, a sensible application of the theory of perturbation will lead to perturbative series expressing the estimates at time k, i.e. $\gamma_i(k)$ and $u_i(k)$ $(i = 1, \ldots, L)$, in terms of the estimates at time $k-1$, i.e. $\gamma_j(k-1)$ and $u_j(k-1)$, $j = 1, \ldots, L$. The recursive equations so obtained will enable us to *adapt* our estimates of the eigenstructure of $R_x(k)$ as the time index k is incremented, without having to completely recompute these estimates. The perturbation analysis is discussed in the next section for the case $n = 1$.

III. FIRST-ORDER PERTURBATION ANALYSIS

In the case $n = 1$, the desired series expansions for the eigendecomposition estimates of $R_x(k)$ take the simple form

$$\gamma_i(k) = \gamma_{0i} + \gamma_{1i}\varepsilon, \quad \gamma_{0i} = \gamma_i(k-1) \qquad (14)$$

$$u_i(k) = u_{0i} + u_{1i}\varepsilon, \quad u_{0i} = u_i(k-1) \qquad (15)$$

where the coefficients of ε, i.e. γ_{1i} and u_{1i}, remain to be determined. Because the series (14)-(15) are linear in ε, we refer to the determination of these coefficients as a first-order perturbation analysis. Although $\gamma_i(k-1)$ and $u_i(k-1)$ are not exact eigenvalues and eigenvectors of the unperturbed matrix

$\hat{R}_x(k-1)$, the conditions (12)-(13) ensure that the theory of perturbation can be applied in a meaningful way.

In this respect, the most serious difficulty comes from the degenerate nature of the noise subspace. If all the eigenvalue estimates $\gamma_i(k-1)$ where distinct, the application of the theory would be relatively straightforward. However, because of the particular structure of the data covariance matrix $R_x(k)$ (4) whose $L-M$ smallest eigenvalues are degenerate, any reasonable estimation procedure will eventually result in $\gamma_{M+1}(k),\ldots,\gamma_L(k)$ clustering together as k increases. In turn, this will make the recursion obtained under the assumption of distinct $\gamma_i(k-1)$ unstable. It is therefore necessary to impose the following condition on the $L-M$ smallest eigenvalue estimates:

$$\gamma_{M+1}(k) = \cdots = \gamma_L(k) \tag{16}$$

for all k. That is, we construct the recursive eigenvalue estimates so that the smallest eigenvalue has the proper multiplicity. Of course, this requires that the signal subspace dimensionality M be known in advance.

A similar problem occurs if some of the eigenvalues $\lambda_i(k)$ of $R_x(k)$ are degenerate for $1 \le i \le M$. In principle, this problem can be handled in the same way as above. However, to simplify the presentation, we shall assume that

$$\gamma_1(k) > \cdots > \gamma_M(k). \tag{17}$$

We refer to (17) as the *non-degenerate signal-subspace* assumption.

Mathematical details of the first-order perturbation analysis under the constraints (16)-(17) are given in [5]. One important characteristic of the resulting coefficients γ_{1i} and u_{1i} is that they depend only on the new data vector $x(k)$ and the previous estimates $\gamma_i(k-1)$ and $u_i(k-1)$, and not on the previous estimate $\hat{R}_x(k-1)$ of the array covariance matrix. Once these coefficients are available, the formulation of a complete adaptive eigendecomposition algorithm is straightforward. The resulting algorithm (P1) is given below:

Data needed at time k:

$$x = x(k)$$

$$u_{0i} = u_i(k-1), \quad i = 1,\ldots,M$$

$$\gamma_{0i} = \gamma_i(k-1), \quad i = 1,\ldots,M+1$$

Computation of first-order coefficients:

$$y_i = u_{0i}^H x, \quad i = 1,\ldots,M \tag{18}$$

$$\gamma_{1i} = |y_i|^2 - \gamma_{0i}, \quad i = 1,\ldots,M \tag{19}$$

$$\gamma_{1,M+1} = \frac{1}{(L-M)}\{\, x^H x - \sum_{i=1}^{M} |y_i|^2 \,\} - \gamma_{0,M+1} \tag{20}$$

$$v_i = y_i u_{0i}, \quad i = 1,\ldots,M \tag{21}$$

$$b_{ji} = \begin{cases} 1, & j = i \\ \dfrac{\gamma_{0j} - \gamma_{0,M+1}}{\gamma_{0j} - \gamma_{0i}}, & j \ne i \end{cases} \tag{22}$$

$$u_{1i} = \frac{y_i^*}{\gamma_{0i} - \gamma_{0,M+1}}\{x - \sum_{j=1}^{M} b_{ji} v_j\} \tag{23}$$

Updating eigenstructure:

$$\gamma_i(k) = \gamma_{0i} + \varepsilon\gamma_{1i}, \quad i = 1,\ldots,M+1 \tag{24}$$

$$u_i(k) = u_{0i} + \varepsilon u_{1i}, \quad i = 1,\ldots,M. \tag{25}$$

The dimensionality of the signal subspace, M, and the initial values $\gamma_i(0)$, $i=1,\ldots,M+1$ and $u_i(0)$, $i=1,\ldots,M$ are needed to start the recursion. They can be obtained, for example, by performing a single eigendecomposition on an initial estimate $\hat{R}_x(0)$ of the array covariance matrix.

Equation (18) can be realized by means of M linear combiners with complex weight vectors given by $u_i(k-1)$. The output y_i of the ith combiner is used to evaluate γ_{1i} (19). To evaluate $\gamma_{1,M+1}$ (20), we must further evaluate the energy of $x(k)$. The quantity v_i (21) is the projection of $x(k)$ along $u_i(k-1)$. The coefficient vector u_{1i} is obtained by subtracting the proper linear combination of the v_j from $x(k)$ (using the b_{ji} as weights), and then scaling the result. In (24)-(25), the coefficients γ_{1i} and u_{1i} are used to update the estimates at time k. Together, (23) and (25) provide a non-linear weight-vector adaptation equation for the ith linear combiner. This non-linearity appears to be a substitute for the Gram-Schmidt orthonormalization step found in other algorithms [2].

In the case of widely spread eigenvalues, considerable simplifications of the above algorithm are possible. Indeed, suppose that

$$\gamma_{01} \gg \cdots \gg \gamma_{0,M+1}. \tag{26}$$

Then,

$$b_{ji} = \begin{cases} 1, & j \le i \\ 0, & j > i \end{cases} \tag{27}$$

and (23) can be simplified as follows:

$$u_{1i} = \frac{y_i^*}{\gamma_{0i}}\{x - \sum_{j=1}^{i} v_j\}. \tag{28}$$

We refer to the resulting algorithm as the approximate perturbation algorithm (P2).

IV. SIMULATIONS

Computer simulated narrow-band array data was used to compare the convergence behavior of the proposed algorithm to that of various gradient-based adaptive eigendecomposition algorithms [2]. The scenario considered for the simulation is the following. A uniform linear array of $L=8$ sensors is used to monitor $M=2$ uncorrelated narrow-band plane waves of common frequency with directions of arrival initially given by $\theta_1 = 9^o$ and $\theta_2 = 12^o$. The intersensor spacing equals half the wavelength. The plane waves are monitored in the presence of background noise as described in Section II. The signal-to-noise ratios are set to $\mathrm{SNR}_1 = \mathrm{SNR}_2 = 10dB$.

To initialize the algorithms, a sequence of 10 independent array output vectors $x(k)$ is first used to estimate $\hat{R}_x$ as in (9). The 2 largest eigenvalues of $\hat{R}_x$ are used as the initial

values $\gamma_1(0)$ and $\gamma_2(0)$, and the corresponding eigenvectors are used as $u_1(0)$ and $u_2(0)$. The initial value $\gamma_3(0)$ is obtained by averaging the remaining eigenvalues.

The performance of the various algorithms is evaluated in terms of the following measure:

$$J(k) = \frac{1}{\sqrt{M}} \| U_s(k) U_s^H(k) - Q_s(k) Q_s^H(k) \|, \qquad (29)$$

where $\|.\|$ denotes the Euclidean norm of a matrix. The quantity $J(k)$ measures the normalized error between the projector on the true signal subspace, i.e. $Q_s(k)Q_s(k)^H$, and an estimate of this projector at time k given by $U_s(k)U_s^H(k)$. Average learning curves are obtained by performing 40 independent experiments (with independent reinitialization).

Fig. 1 shows the average learning curves of the following algorithms under stationary conditions: perturbation algorithm (P1); covariance matrix gradient (YK1); instantaneous gradient (YK2). The parameter ε is set to 0.015, while the corresponding parameters of the other algorithms (i.e. μ_1 and μ_2) are adjusted to give the same steady-state normalized error. Under these conditions, the new algorithm converges more rapidly than the gradient-based algorithms. Fig. 2 shows the average learning curves of P1, YK1 and YK2 under the same conditions as in Fig. 1, except that μ_1 and μ_2 are now adjusted so that the convergence rate at the origin is fixed. In this case, the steady state normalized error of the new algorithm is the lowest. Fig. 3 illustrates the behavior of P1, YK1 and YK2 under non-stationary conditions: initially, the scenario is that of Fig. 2, but at time $k = 200$, θ_1 and θ_2 are changed suddenly. The approximate perturbation algorithm (P2) is compared to P1 in Fig.4. We note that both algorihtms have a similar behavior, except that the computational load of P2 is much lower. In fact, of all the algorithms considered, P2 has the smallest computational load, requiring on the order of $3LM$ complex multiplications. We have also found P2 to be more robust than P1 in situations where some eigenvalues get too close together, despite the assumption (26) made in the derivation of P2.

ACKNOWLEDGMENT

This work was supported by the Natural Sciences and Engineering Research Council of Canada.

REFERENCES

[1] S. U. Pillai, "Array Signal Processing," New York: Springer-Verlag, 1989.

[2] J.-F. Yang and M. Kaveh, "Adaptive eigensubspace algorithms for direction or frequency estimation and tracking," *IEEE Trans. on ASSP*, vol. 36, pp. 241-251, Feb. 1988.

[3] R. D. DeGroat and R. A. Roberts, "Efficient, numerically stabilized rank-one eigenstructure updating," *IEEE Trans. on ASSP*, vol. 38, pp. 301-316, Feb. 1990.

[4] F. Rellich, "Perturbation Theory of Eigenvalue Problems," New York: Gordon and Breach, 1969.

[5] B. Champagne, "Adaptive signal-subspace processing based on first-order perturbation analysis," *Proc. IEEE Pac. Rim Conf. on CCSP.*, Victoria, Canada, May 8-9, 1991, pp.120-123.

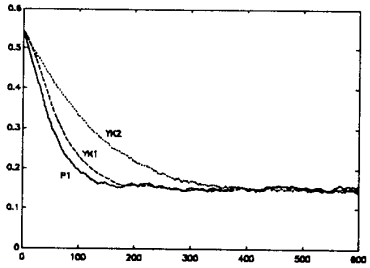

Fig. 1. Average learning curves of algorithms P1, YK1 and YK2 for fixed steady-state error ($\theta_1 = 9^o$, $\theta_2 = 12^o$, $\mu_1 = .0016$, $\mu_2 = .0009$).

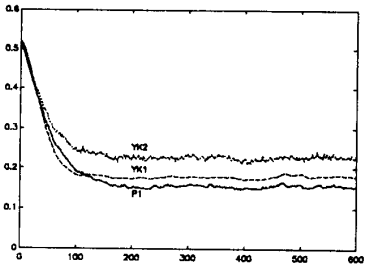

Fig. 2. Average learning curves of algorithms P1, YK1 and YK2 for fixed convergence rate at the origin ($\theta_1 = 9^o$, $\theta_2 = 12^o$, $\mu_1 = \mu_2 = .0025$).

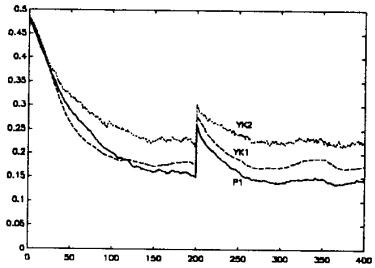

Fig. 3. Average learning curves of algorithms P1, YK1 and YK2 under non-stationary conditions ($\theta_1 = 9^o$, $\theta_2 = 12^o$ for $0 \le k < 200$, $\theta_1 = 11^o$, $\theta_2 = 14^o$ for $200 \le k < 400$, $\mu_1 = \mu_2 = .0025$).

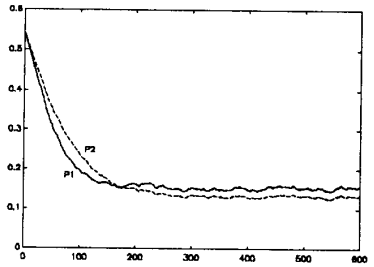

Fig. 4. Average learning curves of algorithms P1 and P2 ($\theta_1 = 9^o$, $\theta_2 = 12^o$).