FAST ADAPTIVE ESPRIT ALGORITHM

Roland Badeau, Gaël Richard and Bertrand David

École Nationale Supérieure des Télécommunications
Département de Traitement du Signal et des Images
46 rue Barrault, 75634 Paris Cedex 13 France
roland.badeau, gael.richard, bertrand.david@enst.fr

ABSTRACT

The ESPRIT algorithm[1] is a subspace-based high resolution method used in source localization and spectral analysis. It relies on the rotational invariance property of the signal subspace, and provides accurate estimates of the signal parameters. However, its main drawback is a high computational cost.

In an adaptive context, some very fast algorithms were proposed to robustly track the variations of the signal subspace. Based on these subspace trackers, we propose a new adaptive implementation of the ESPRIT algorithm, faster than the existing methods.

1. INTRODUCTION

The ESPRIT algorithm [1] first consists in estimating the signal subspace. Then a so-called *spectral matrix* is derived from this subspace, whose eigenvalues characterize the signal parameters.

Adaptive implementations of the ESPRIT algorithm for tracking the spectral matrix were already proposed in the literature, based on specific subspace trackers. For example, the technique presented in [2] relies on a *Singular Value Decomposition* (SVD) updating algorithm [3], and that proposed in [4] is based on the rank-revealing URV decomposition [5]. The complexity of these methods is proportional to $O(n^2)$ operations for each time update, where n is the dimension of the data vectors. In [6], other adaptive algorithms were proposed for the LORAF [7] and the Bi-SVD [8] subspace trackers. In comparison with [2] and [4], the complexity of these algorithms is reduced to $O(nr^2)$ or $O(nr)$, where r is the dimension of the signal subspace (in practice $r << n$). However we observed that only the $O(nr^2)$ algorithms in [6] are of practical interest, due to the degraded performance of the $O(nr)$ subspace trackers. In [9], we proposed a new adaptive algorithm based on the PAST [10] and related subspace trackers, whose complexity is $O(nr)$. Its performance is the same as that of the $O(nr^2)$ algorithms in [6]. Below, we propose a more efficient implementation of the approach introduced in [9].

The second step of the ESPRIT algorithm consists in computing the eigenvalues of the $r \times r$ estimated spectral matrix. They can be obtained via an *Eigen-Value Decomposition* (EVD), whose complexity is $O(r^3)$, as proposed in [4]. However they can also be recursively updated. A first approach proposed in [11] relies on perturbation theory, but we observed that it fails to robustly track frequency variations. Another technique proposed in [6] relies on the *split-Schur* theorem, but proves to be less precise than an EVD,

for a similar computational cost. We propose in this paper a new method wich exactly and efficiently updates the eigenvalues of the spectral matrix at each time step.

This paper is organized as follows. Section 2 briefly describes the ESPRIT method. Section 3 introduces a new algorithm for tracking the spectral matrix. Then our adaptive EVD algorithm is presented in section 4. The performance of this fast adaptive ESPRIT method is compared to that of existing methods in section 5. Section 6 presents the analysis of an audio signal. Finally, section 7 summarizes the main conclusions of this paper.

2. ESTIMATION OF SIGNAL PARAMETERS VIA ROTATIONAL INVARIANCE TECHNIQUES

In source localization and spectral analysis, the observed data can be arranged into a sequence of n-dimensional vectors $\{\boldsymbol{x}(t)\}$, where $t \in \mathbb{Z}$. If there is no noise, it can be shown that these vectors belong to the *signal subspace* spanned by the Vandermonde matrix

$$\boldsymbol{V} = \begin{bmatrix} 1 & \cdots & 1 \\ z_1 & \cdots & z_r \\ \vdots & \cdots & \vdots \\ z_1^{n-1} & \cdots & z_r^{n-1} \end{bmatrix},$$

where r is the model order and $\forall k \in \{1 \ldots r\}$, $z_k \in \mathbb{C} - \{0\}$ is a complex pole of the signal. In practice, the matrix $\boldsymbol{V}$ is unknown. However, in presence of white noise, a $n \times r$ matrix $\boldsymbol{W}(t)$ spanning the signal subspace can be estimated from the vectors observed in the analysis window. This operation involves the SVD of a data matrix, or the EVD of a correlation matrix, which requires $O(n^3)$ operations. In an adaptive context, the signal variations can be tracked by time-shifting the analysis window. The complexity of the subspace estimation can be reduced by using a fast subspace tracker which efficiently updates $\boldsymbol{W}(t)$.

The ESPRIT algorithm relies on the rotational invariance property of the Vandermonde matrix $\boldsymbol{V}$. Let $\boldsymbol{V}_{\downarrow}$ be the matrix extracted from $\boldsymbol{V}$ by deleting the last row. Similarly, let $\boldsymbol{V}_{\uparrow}$ be the matrix extracted from $\boldsymbol{V}$ by deleting the first row. Then it can be shown that $\boldsymbol{V}_{\uparrow} = \boldsymbol{V}_{\downarrow} \boldsymbol{D}$, where $\boldsymbol{D} = \text{diag}(z_1 \ldots z_r)$. The estimated matrix $\boldsymbol{W}(t)$ satisfies a similar invariance property: $\boldsymbol{W}(t)_{\uparrow} = \boldsymbol{W}(t)_{\downarrow} \boldsymbol{\Phi}(t)$, where the eigenvalues of the $r \times r$ matrix $\boldsymbol{\Phi}(t)$, herein called *spectral matrix*, are the complex poles z_k.

---

[1]ESPRIT stands for *Estimation of Signal Parameters via Rotational Invariance Techniques*.

Consequently, the LS-ESPRIT algorithm [1] consists in:

1. estimating the subspace weighting matrix $\boldsymbol{W}(t)$,
2. computing $\boldsymbol{\Phi}(t)$ by means of the *Least Squares* (LS) method:

$$\boldsymbol{\Phi}(t) = \boldsymbol{W}(t)_{\downarrow}^{\dagger} \boldsymbol{W}(t)_{\uparrow} \qquad (1)$$

(where $\dagger$ denotes the Moore-Penrose pseudo-inverse),

3. estimating the complex poles z_k as the eigenvalues of $\boldsymbol{\Phi}(t)$.

3. SPECTRAL MATRIX TRACKING

As mentioned in section 2, the first step of the ESPRIT algorithm consists in estimating the subspace weighting matrix $\boldsymbol{W}(t)$. Although no assumption is made about the structure of this matrix, many subspace estimation techniques produce an orthonormal matrix $\boldsymbol{W}(t)$. In section 3.1, we show that this property can be taken into account in the second step of the ESPRIT method, in order to efficiently compute the spectral matrix. Besides, in an adaptive context, the complexity can be further reduced by exploiting the structure of the update involved in the subspace tracker, as shown in section 3.2. We thus obtain a fast algorithm for tracking the spectral matrix. Its cost is lower than that of the algorithm that we proposed in [9], which did not exploit the orthonormality of $\boldsymbol{W}(t)$.

3.1. Fast computation of the spectral matrix

If the matrix $\boldsymbol{W}(t)_{\downarrow}$ is full-rank (which is always the case in practice), equation (1) can be written in the form

$$\boldsymbol{\Phi}(t) = \boldsymbol{\Omega}(t)\,\boldsymbol{\Psi}(t) \qquad (2)$$

where $\boldsymbol{\Omega}(t)$ and $\boldsymbol{\Psi}(t)$ are the $r \times r$ matrices

$$\boldsymbol{\Omega}(t) \triangleq \left(\boldsymbol{W}_{\downarrow}(t)^H \boldsymbol{W}_{\downarrow}(t)\right)^{-1}, \qquad (3)$$

$$\boldsymbol{\Psi}(t) \triangleq \boldsymbol{W}_{\downarrow}(t)^H \boldsymbol{W}_{\uparrow}(t). \qquad (4)$$

The computation of $\boldsymbol{\Psi}(t)$ involves nr^2 MAC[2], and that of $\boldsymbol{\Omega}(t)$ involves $nr^2 + O(r^3)$ MAC. The overall computation of $\boldsymbol{\Phi}(t)$ as defined in equation (2) therefore involves $2nr^2 + O(r^3)$ MAC. It will be shown below that this cost can be reduced by assuming that the matrix $\boldsymbol{W}(t)$ is orthonormal.

Indeed, if $\boldsymbol{W}(t)$ is orthonormal, $\boldsymbol{\Omega}(t)$ can be easily calculated. The equation $\boldsymbol{W}(t)^H \boldsymbol{W}(t) = \boldsymbol{I}_r$ shows that $\boldsymbol{W}_{\downarrow}(t)^H \boldsymbol{W}_{\downarrow}(t) = \boldsymbol{I}_r - \boldsymbol{\nu}(t)\boldsymbol{\nu}(t)^H$, where $\boldsymbol{\nu}(t)^H$ is the last row of $\boldsymbol{W}(t)$. Thus $\boldsymbol{W}_{\downarrow}(t)^H \boldsymbol{W}_{\downarrow}(t)$ is a rank-one modification of the identity matrix. Finally, the matrix inversion lemma [12, pp. 18-19] shows that

$$\boldsymbol{\Omega}(t) = \boldsymbol{I}_r + \frac{1}{1 - \|\boldsymbol{\nu}(t)\|^2}\,\boldsymbol{\nu}(t)\,\boldsymbol{\nu}(t)^H.$$

Substituting this last equation into equation (2) yields

$$\boxed{\boldsymbol{\Phi}(t) = \boldsymbol{\Psi}(t) + \frac{1}{1 - \|\boldsymbol{\nu}(t)\|^2}\,\boldsymbol{\nu}(t)\,\boldsymbol{\varphi}(t)^H} \qquad (5)$$

where the r-dimensional vector $\boldsymbol{\varphi}(t)$ is defined as

$$\boldsymbol{\varphi}(t) = \boldsymbol{\Psi}(t)^H \boldsymbol{\nu}(t). \qquad (6)$$

Thus the matrix $\boldsymbol{\Phi}(t)$ is a rank-one modification of $\boldsymbol{\Psi}(t)$. By successively computing equations (4), (6) and (5), the calculation of the spectral matrix involves only $nr^2 + O(r^2)$ MAC.

---

[2] In this paper, the computational costs are expressed in terms of Multiply-ACcumulate (MAC) operations.

Table 1. Adaptive computation of the spectral matrix

	Eq.	Cost
$\boldsymbol{e}_-(t) = \boldsymbol{W}_{\downarrow}(t-1)^H \boldsymbol{e}_{\uparrow}(t)$	(8)	nr
$\boldsymbol{e}_+(t) = \boldsymbol{W}_{\uparrow}(t-1)^H \boldsymbol{e}_{\downarrow}(t)$	(9)	nr
$\boldsymbol{e}'_+(t) = \boldsymbol{e}_+(t) + \boldsymbol{g}(t)\left(\boldsymbol{e}_{\uparrow}(t)^H \boldsymbol{e}_{\downarrow}(t)\right)$	(10)	n
$\boldsymbol{\Psi}(t) = \boldsymbol{\Psi}(t-1) + \boldsymbol{e}_-(t)\boldsymbol{g}(t)^H + \boldsymbol{g}(t)\boldsymbol{e}'_+(t)^H$	(11)	$2r^2$
$\boldsymbol{\varphi}(t) = \boldsymbol{\Psi}(t)^H \boldsymbol{\nu}(t)$	(6)	r^2
$\boldsymbol{\Phi}(t) = \boldsymbol{\Psi}(t) + \frac{1}{1-\|\boldsymbol{\nu}(t)\|^2}\,\boldsymbol{\nu}(t)\,\boldsymbol{\varphi}(t)^H$	(5)	r^2

3.2. Adaptive computation of the spectral matrix

Several subspace trackers such as PAST [10] and NIC [13] provide a rank-one update for $\boldsymbol{W}(t)$:

$$\boldsymbol{W}(t) = \boldsymbol{W}(t-1) + \boldsymbol{e}(t)\boldsymbol{g}(t)^H \qquad (7)$$

where $\boldsymbol{e}(t)$ and $\boldsymbol{g}(t)$ are n and r-dimensional vectors. Here we focus on those which additionally guarantee the orthonormality of $\boldsymbol{W}(t)$, such as in [14–17], in order to use the results in section 3.1. In this case, it will be shown below that $\boldsymbol{\Psi}(t)$ satisfies a similar recursion. Indeed, let

$$\boldsymbol{e}_-(t) = \boldsymbol{W}_{\downarrow}(t-1)^H \boldsymbol{e}_{\uparrow}(t), \qquad (8)$$

$$\boldsymbol{e}_+(t) = \boldsymbol{W}_{\uparrow}(t-1)^H \boldsymbol{e}_{\downarrow}(t), \qquad (9)$$

$$\boldsymbol{e}'_+(t) = \boldsymbol{e}_+(t) + \boldsymbol{g}(t)\left(\boldsymbol{e}_{\uparrow}(t)^H \boldsymbol{e}_{\downarrow}(t)\right). \qquad (10)$$

Substituting equations (7), (8), (9) and (10) into equation (4) shows that $\boldsymbol{\Psi}(t)$ satisfies a rank-two recursion:

$$\boxed{\boldsymbol{\Psi}(t) = \boldsymbol{\Psi}(t-1) + \boldsymbol{e}_-(t)\boldsymbol{g}(t)^H + \boldsymbol{g}(t)\boldsymbol{e}'_+(t)^H.} \qquad (11)$$

This recursion leads to an adaptive computation of the spectral matrix, as shown in table 1. The global cost is only $n(2r+1) + 4r^2$ MAC (instead of in $8nr + O(r^2)$ in [9]).

Besides, the last row of equation (7) can be written in the form:

$$\boldsymbol{\nu}(t) = \boldsymbol{\nu}(t-1) + \boldsymbol{g}(t)e_n(t)^* \qquad (12)$$

where $e_n(t)$ is the last coefficient of $\boldsymbol{e}(t)$. Substituting equations (11) and (12) into (5) yields a rank-three recursion for $\boldsymbol{\Phi}(t)$:

$$\begin{aligned}\boldsymbol{\Phi}(t) = {} & \boldsymbol{\Phi}(t-1) + \boldsymbol{g}(t)\boldsymbol{e}''_+(t)^H \\ & + \boldsymbol{e}_-(t)\boldsymbol{g}(t)^H + \boldsymbol{\nu}(t-1)\Delta\boldsymbol{\varphi}(t)^H,\end{aligned} \qquad (13)$$

where

$$\boldsymbol{e}''_+(t) = \boldsymbol{e}'_+(t) + \frac{e_n(t)}{1 - \|\boldsymbol{\nu}(t)\|^2}\,\boldsymbol{\varphi}(t),$$

$$\Delta\boldsymbol{\varphi}(t) = \frac{\boldsymbol{\varphi}(t)}{1 - \|\boldsymbol{\nu}(t)\|^2} - \frac{\boldsymbol{\varphi}(t-1)}{1 - \|\boldsymbol{\nu}(t-1)\|^2}.$$

Below, equation (13) will be compacted in the form

$$\boldsymbol{\Phi}(t) = \boldsymbol{\Phi}(t-1) + \underline{\boldsymbol{a}}(t)\,\underline{\boldsymbol{b}}(t)^H \qquad (14)$$

where $\underline{\boldsymbol{a}}(t)$ and $\underline{\boldsymbol{b}}(t)$ are $r \times 3$ matrices. Note that the additional calculation of $\boldsymbol{e}''_+(t)$ and $\Delta\boldsymbol{\varphi}(t)$ only involves $O(r)$ MAC.

4. EIGENVALUES TRACKING

The eigenvalues of $\mathbf{\Phi}(t)$ can be obtained by computing its EVD, which involves $O(r^3)$ MAC. However, in an adaptive context, we propose in this section a fast algorithm for diagonalizing $\mathbf{\Phi}(t)$, which exploits the structure of the update in equation (14).

Suppose that an EVD of the $r \times r$ matrix $\mathbf{\Phi}(t-1)$ is known:

$$\boldsymbol{D}(t-1) = \boldsymbol{G}'(t-1)^H \, \mathbf{\Phi}(t-1) \, \boldsymbol{G}(t-1) \qquad (15)$$

where $\boldsymbol{D}(t-1) = \text{diag}\{z_1(t-1) \ldots z_r(t-1)\}$, $\boldsymbol{G}(t-1)$ is non-singular, and $\boldsymbol{G}'(t-1) = \boldsymbol{G}(t-1)^{-H}$. Then define the $r \times r$ matrix

$$\widetilde{\boldsymbol{D}}(t) \triangleq \boldsymbol{G}'(t-1)^H \, \mathbf{\Phi}(t) \, \boldsymbol{G}(t-1). \qquad (16)$$

Note that the eigenvalues of $\widetilde{\boldsymbol{D}}(t)$ are the same as those of $\mathbf{\Phi}(t)$. Substituting equations (14) and (15) into equation (16) yields

$$\widetilde{\boldsymbol{D}}(t) = \boldsymbol{D}(t-1) + \underline{\widetilde{\boldsymbol{a}}}(t) \, \underline{\widetilde{\boldsymbol{b}}}(t)^H, \qquad (17)$$

where

$$\underline{\widetilde{\boldsymbol{a}}}(t) = \boldsymbol{G}'(t-1)^H \underline{\boldsymbol{a}}(t) \qquad (18)$$
$$\underline{\widetilde{\boldsymbol{b}}}(t) = \boldsymbol{G}(t-1)^H \underline{\boldsymbol{b}}(t). \qquad (19)$$

Thus the matrix $\widetilde{\boldsymbol{D}}(t)$ is a low-rank modification of $\boldsymbol{D}(t-1)$. Now suppose that $\widetilde{\boldsymbol{D}}(t)$ is diagonalizable in the form

$$\boldsymbol{D}(t) = \widetilde{\boldsymbol{G}}'(t)^H \, \widetilde{\boldsymbol{D}}(t) \, \widetilde{\boldsymbol{G}}(t), \qquad (20)$$

where $\boldsymbol{D}(t) = \text{diag}\{z_1(t) \ldots z_r(t)\}$, $\widetilde{\boldsymbol{G}}(t)$ is non-singular, and $\widetilde{\boldsymbol{G}}'(t) = \widetilde{\boldsymbol{G}}(t)^{-H}$. Then an EVD of $\mathbf{\Phi}(t)$ can be derived from this last EVD. Indeed, substituting equation (20) into (16) yields

$$\boldsymbol{D}(t) = \boldsymbol{G}'(t)^H \, \mathbf{\Phi}(t) \, \boldsymbol{G}(t),$$

where the $r \times r$ matrices $\boldsymbol{G}(t)$ and $\boldsymbol{G}(t)$ are non-singular:

$$\boldsymbol{G}(t) = \boldsymbol{G}(t-1) \, \widetilde{\boldsymbol{G}}(t), \qquad (21)$$
$$\boldsymbol{G}'(t) = \boldsymbol{G}'(t-1) \, \widetilde{\boldsymbol{G}}'(t). \qquad (22)$$

Note that the column vectors $\{\widetilde{\boldsymbol{g}}_1(t), \ldots, \widetilde{\boldsymbol{g}}_r(t)\}$ of the matrix $\widetilde{\boldsymbol{G}}(t)$ are the right eigenvectors of the matrix $\widetilde{\boldsymbol{D}}(t)$. In the same way, the column vectors $\{\widetilde{\boldsymbol{g}}'_0(t), \ldots, \widetilde{\boldsymbol{g}}'_{r-1}(t)\}$ of the matrix $\widetilde{\boldsymbol{G}}'(t)$ are the left eigenvectors of the matrix $\widetilde{\boldsymbol{D}}(t)$. Below, proposition 1 describes the eigenstructure of the matrix $\widetilde{\boldsymbol{D}}(t)$. Its proof can be found in appendix A.

Proposition 1. *Suppose that all eigenvalues of $\mathbf{\Phi}(t-1)$ and $\mathbf{\Phi}(t)$ are simple, and that all eigenvalues of $\mathbf{\Phi}(t)$ differ from those of $\mathbf{\Phi}(t-1)$. Consider the 3×3 matrix*

$$\underline{\phi}(z;t) \triangleq \underline{I} - \underline{\widetilde{\boldsymbol{b}}}(t)^H \, (z\boldsymbol{I}_r - \boldsymbol{D}(t-1))^{-1} \, \underline{\widetilde{\boldsymbol{a}}}(t), \quad (23)$$

defined for all $z \in \mathbb{C} - \{\text{eig}(\mathbf{\Phi}(t-1))\}$.

Then the r eigenvalues of $\mathbf{\Phi}(t)$ match the roots of the equation

$$\det(\underline{\phi}(z;t)) = 0. \qquad (24)$$

If $z_k(t)$ is such an eigenvalue, the corresponding right eigenvector $\widetilde{\boldsymbol{g}}_k(t)$ and left eigenvector $\widetilde{\boldsymbol{g}}'_k(t)$ of $\widetilde{\boldsymbol{D}}(t)$ can be written in the form

$$\widetilde{\boldsymbol{g}}_k(t) = (z_k(t)\boldsymbol{I}_r - \boldsymbol{D}(t-1))^{-1} \, \underline{\widetilde{\boldsymbol{a}}}(t) \, \underline{v}_k(t), \quad (25)$$
$$\widetilde{\boldsymbol{g}}'_k(t) = (z_k(t)\boldsymbol{I}_r - \boldsymbol{D}(t-1))^{-1*} \, \underline{\widetilde{\boldsymbol{b}}}(t) \, \underline{v}'_k(t), \quad (26)$$

where

$$\underline{v}_k(t) \in \text{ker}\left\{\underline{\phi}(z_k(t);t)\right\}, \qquad (27)$$
$$\underline{v}'_k(t) \in \text{ker}\left\{\underline{\phi}(z_k(t);t)^H\right\}. \qquad (28)$$

Proposition 1 suggests a fast way of diagonalizing $\widetilde{\boldsymbol{D}}(t)$, from which the EVD of $\mathbf{\Phi}(t)$ can be deduced. The pseudo-code of the complete algorithm is summarized in table 2. Its complexity is $2r^3 + O(r^2)$, instead of $O(r^3)$ for a full EVD[3].

Table 2. Adaptive computation of the eigenvalues

	Eq.	Cost
$\underline{\widetilde{\boldsymbol{a}}}(t) = \boldsymbol{G}'(t-1)^H \underline{\boldsymbol{a}}(t)$	(18)	$3r^2$
$\underline{\widetilde{\boldsymbol{b}}}(t) = \boldsymbol{G}(t-1)^H \underline{\boldsymbol{b}}(t)$	(19)	$3r^2$
Compute[4] the roots $\{z_k(t)\}$ of $\det\left(\underline{\phi}(z;t)\right) = 0$	(24)	$O(r^2)$
For $k = 1$ to r do		$r\times$
$\quad \underline{\phi}_k(t) = \underline{\phi}(z_k(t);t)$	(23)	$O(1)$
$\quad$ Extract $\underline{v}_k(t)$ from $\text{ker}\left\{\underline{\phi}_k(t)\right\}$	(27)	$O(1)$
$\quad \widetilde{\boldsymbol{g}}_k(t) \propto (z_k(t)\boldsymbol{I}_r - \boldsymbol{D}(t-1))^{-1} \underline{\widetilde{\boldsymbol{a}}}(t)\underline{v}_k(t)$	(25)	$3r$
$\quad$ Extract $\underline{v}'_k(t)$ from $\text{ker}\left\{\underline{\phi}_k(t)^H\right\}$	(28)	$O(1)$
$\quad \widetilde{\boldsymbol{g}}'_k(t) \propto (z_k(t)\boldsymbol{I}_r - \boldsymbol{D}(t-1))^{-1*} \underline{\widetilde{\boldsymbol{b}}}(t)\underline{v}'_k(t)$	(26)	$3r$
End for		
Re-normalize the columns of $\widetilde{\boldsymbol{G}}(t)$ and $\widetilde{\boldsymbol{G}}'(t)$		
$\quad$ in order to obtain $\widetilde{\boldsymbol{G}}'(t)^H \widetilde{\boldsymbol{G}}(t) = \boldsymbol{I}_r$		$4r^2$
$\boldsymbol{G}(t) = \boldsymbol{G}(t-1) \widetilde{\boldsymbol{G}}(t)$	(21)	r^3
$\boldsymbol{G}'(t) = \boldsymbol{G}'(t-1) \widetilde{\boldsymbol{G}}'(t)$	(22)	r^3

5. SIMULATION RESULTS

In this section, the performance of the tracking is illustrated in the context of frequency estimation. The test signal is a sum of $r = 4$ complex sinusoidal sources plus a complex white gaussian noise (the SNR is 5.7 dB). The frequencies of the sinusoids vary according to a scenario illustrated in figure 1-a: two of them are constant and have close values (below the Fourier resolution limit), the third one is sinusoidally modulated, and the last one is a discontinuous, piecewise linear function. The tracking of these frequencies is performed by means of three algorithms, whose computational costs are summarized in table 3: a non-adaptive estimation scheme based on the EVD of the signal correlation matrix, the adaptive ES-PRIT algorithm proposed in [6], and our new methods for tracking the spectral matrix and its eigenvalues, associated to the YAST subspace tracker [16].

Figures 1-b to 1-d show the frequency tracking results, obtained with parameters[5] $n = 80$ and $\beta = 0.99$. The dotted lines

---

[3]The computation of an EVD is an iterative process whose exact operation count depends on the particular matrix to be decomposed. However, the number of MAC involved is much greater than $2r^3$ in usual cases.

[4]It can be shown that $\det(\underline{\phi}(z;t))$ is proportional to the characteristic polynomial of $\widetilde{\boldsymbol{D}}(t)$. The roots of this equation can be computed via standard numerical methods.

[5]β is the forgetting factor of the exponential window used for estimating the signal correlation matrix.
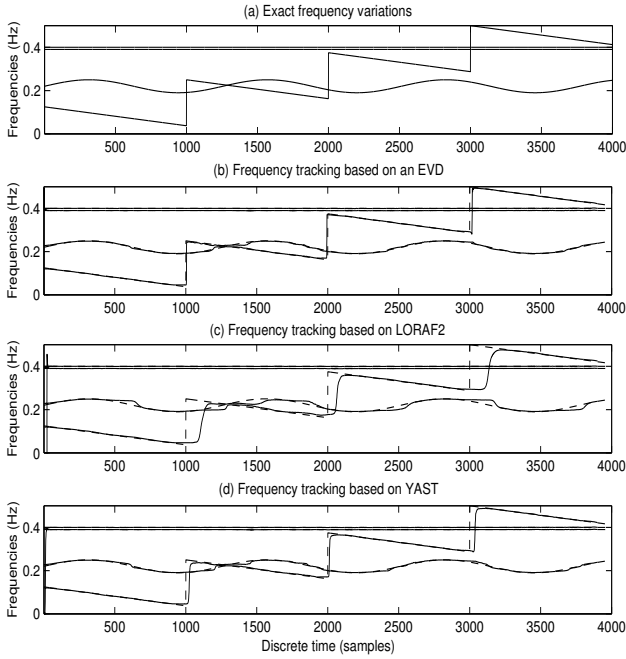
Fig. 1. Frequency estimation and tracking

Table 3. Complexities of various adaptive ESPRIT algorithms

Step	*Algo. 1*	*Algo. 2*	*Algo. 3*
1. Subspace estimation	EVD $O(n^3)$	LORAF2 [7] nr^2	YAST [16] $4nr$
2. Spectral matrix	eq. (1) $2nr^2$	ref. [6] nr^2	Table 1 $2nr$
3. Complex poles	EVD $O(r^3)$	EVD $O(r^3)$	Table 2 $2r^3$

indicate the true frequency parameters, while the solid lines indicate the estimated frequencies. The frequency trajectories obtained with the EVD-based algorithm are displayed in figure 1-b. This result is to be compared to that obtained with the adaptive ESPRIT algorithm [6] associated with the LORAF2 subspace tracker [7], represented in figure 1-c. The response of this algorithm to frequency variations is slower. Finally, figure 1-d shows the trajectories estimated by the tracking methods presented in this paper, associated to the YAST subspace tracker [16]. The performance of this technique is closer to that of the EVD-based algorithm. Besides, it is the less computationally demanding approach.

6. APPLICATION TO AUDIO

This section illustrates the application of our adaptive ESPRIT algorithm to a musical signal. The study deals with two piano tones, C5 and G5, sampled at 11025 Hz.

First, it must be mentioned that audio signals often require some pre-processing before estimating and tracking the sinusoids. For example, signals with a high number of sinusoids (typically low-pitched sounds) may first be decomposed into several sub-

band signals (via filtering/decimating, as proposed in [18]). In the example proposed below, this pre-processing is not used, since the chosen piano tones have few sinusoidal components. In addition, it is well known that the energy of audio signals is not evenly distributed over frequencies. To compensate the energy decrease, we use a pre-emphasis filter obtained by linear prediction at order 4.
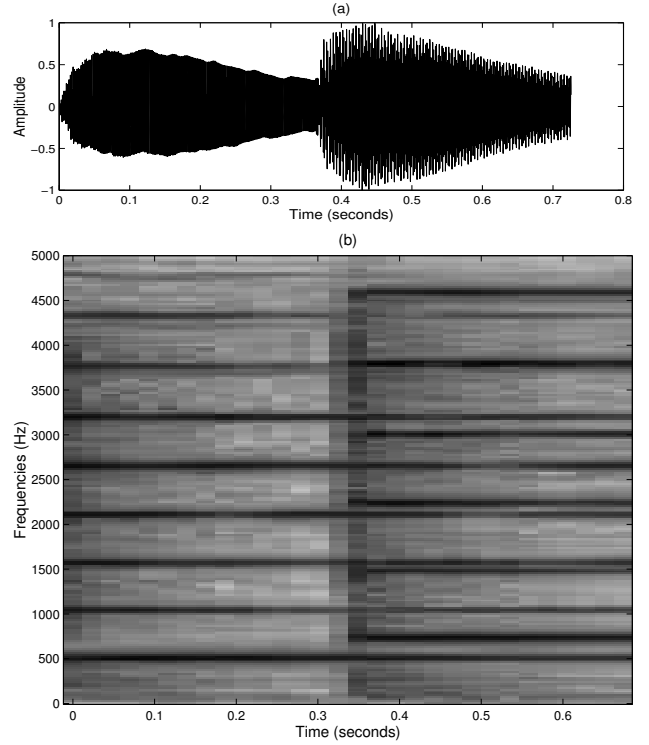

Fig. 2. *Time waveform (a) and spectrogram (b) of the piano signal for the frequency tracking test.*

The frequency tracking method presented in sections 3 and 4 is applied to the following signal: the C5 tone is played at time $t = 0s$, then the G5 is played at time $t = 0.36s$, while the C5 is maintained (the time waveform and the spectrogram of this signal are represented in figure 2). Figure 3 shows the result of the tracking. The analysis parameters are $r = 28$ (*i.e.* 14 real sinusoids), $n = 280$ (*i.e.* 25ms), and $\beta = 0.99$, and the sinusoids energies (estimated by means of the LS method) are represented on a logarithmic scale using gray levels for the plot. Since the number of sinusoids is over-estimated in the first half of the sound, some spurious poles are detected. However, the algorithm proves to be robust to the attack of the second tone.

7. CONCLUSIONS

In this paper, we proposed a fully adaptive implementation of the ESPRIT algorithm. This new approach can be used in conjunction with any subspace tracker which involves a rank-one update and guarantees the orthonormality of the subspace weighting matrix, such as [14–17]. These two conditions are necessary for efficiently updating the spectral matrix $\mathbf{\Phi}(t)$, by means of a rank-three update. The resulting tracking method is faster than the algorithm that we proposed in [9], and than any other existing method. Be-
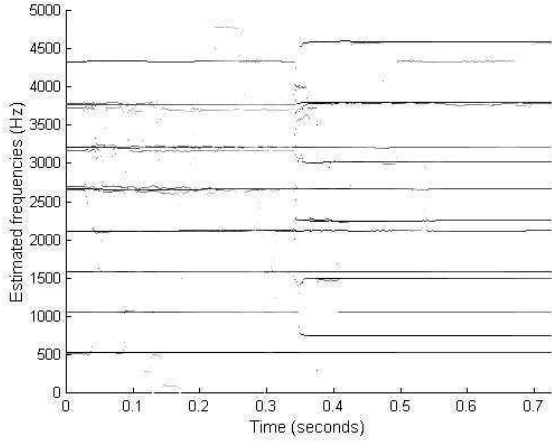
Fig. 3. *Frequency tracking of the piano signal.*

sides, the EVD of the spectral matrix can be updated by taking the structure of the recursion for $\boldsymbol{\Phi}(t)$ into account. The overall complexity of our new adaptive ESPRIT algorithm is only $2nr + 2r^3$ MAC. Our simulation results show that this algorithm robustly tracks abrupt frequency variations, and can be successfully applied to audio signals. It reaches the best performance when used in conjunction with the YAST subspace tracker [16].

A. APPENDIX

Proof of proposition 1. Let $z_k(t)$ be an eigenvalue of $\widetilde{\boldsymbol{D}}(t)$, and $\widetilde{\boldsymbol{g}}_k(t)$ and $\widetilde{\boldsymbol{g}}_k'(t)$ the corresponding right and left eigenvectors. Then

$$(z\boldsymbol{I}_r - \boldsymbol{D}(t-1))\,\widetilde{\boldsymbol{g}} = \widetilde{\underline{\boldsymbol{a}}}(t)\left(\widetilde{\underline{\boldsymbol{b}}}(t)^H\widetilde{\boldsymbol{g}}\right), \qquad (29)$$

$$(z\boldsymbol{I}_r - \boldsymbol{D}(t-1))^*\,\widetilde{\boldsymbol{g}}' = \widetilde{\underline{\boldsymbol{b}}}(t)\left(\widetilde{\underline{\boldsymbol{a}}}(t)^H\widetilde{\boldsymbol{g}}'\right). \qquad (30)$$

Define

$$\underline{v}_k(t) \triangleq \widetilde{\underline{\boldsymbol{b}}}(t)^H\widetilde{\boldsymbol{g}}_k(t), \qquad (31)$$

$$\underline{v}_k'(t) \triangleq \widetilde{\underline{\boldsymbol{a}}}(t)^H\widetilde{\boldsymbol{g}}_k'. \qquad (32)$$

If $z_k(t)$ is not an eigenvalue of $\boldsymbol{\Phi}(t-1)$, the matrix $z_k(t)\boldsymbol{I}_r - \boldsymbol{D}(t-1)$ is non-singular. Then equation (25) is obtained by left-multiplying equation (29) by $(z\boldsymbol{I}_r - \boldsymbol{D}(t-1))^{-1}$. Substituting equation (25) into equation (31) yields equation (27).

Since the vector $\widetilde{\boldsymbol{g}}_k(t)$ is non-zero, $\underline{v}_k(t)$ is also non-zero, thus the matrix $\phi(z_k(t); t)$ is singular and its determinant is zero. Then equation (26) is obtained by left-multiplying equation (30) by $(z\boldsymbol{I}_r - \boldsymbol{D}(t-1))^{-1*}$. Substituting equation (26) into equation (32) finally yields equation (28). Conversely, if $z_k(t)$ is defined as a root of the equation (24), then the vectors $\underline{v}_k(t)$ and $\underline{v}_k'(t)$ can be defined as in equations (27) and (28), and the vectors $\widetilde{\boldsymbol{g}}_k(t)$ and $\widetilde{\boldsymbol{g}}_k(t)$ can be defined as in equations (25) and (26). Then it can be readily verified that $\widetilde{\boldsymbol{g}}_k(t)$ and $\widetilde{\boldsymbol{g}}_k(t)$ are the right and left eigenvectors of $\widetilde{\boldsymbol{D}}(t)$ associated to the eigenvalue $z_k(t)$. Thus the eigenvalues of $\widetilde{\boldsymbol{D}}(t)$ match the roots of equation (24). $\qquad\square$

B. REFERENCES

[1] 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.

[2] M. Moonen, F. J. Vanpoucke, and E. F. Deprettere, "Parallel and adaptive high-resolution direction finding," *IEEE Trans. Signal Processing*, vol. 42, no. 9, pp. 2439–2448, Sept. 1994.

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

[4] K. J. R. Liu, D. P. O'Leary, G. W. Stewart, and Y-J. J. Wu, "URV ESPRIT for tracking time-varying signals," *IEEE Trans. Signal Processing*, vol. 42, no. 12, pp. 3443–3448, Dec. 1994.

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

[6] P. Strobach, "Fast recursive subspace adaptive ESPRIT algorithms," *IEEE Trans. Sig. Proc.*, vol. 46, no. 9, pp. 2413–2430, Sept. 1998.

[7] P. Strobach, "Low-rank adaptive filters," *IEEE Trans. Signal Processing*, vol. 44, no. 12, pp. 2932–2947, Dec. 1996.

[8] P. Strobach, "Bi-iteration SVD subspace tracking algorithms," *IEEE Trans. Signal Proc.*, vol. 45, no. 5, pp. 1222–1240, May 1997.

[9] R. Badeau, G. Richard, and B. David, "Adaptive ESPRIT algorithm based on the PAST subspace tracker," in *Proc. of ICASSP'03*, Hong Kong, China, Apr. 2003, vol. 6, pp. 229–232, IEEE.

[10] B. Yang, "Projection Approximation Subspace Tracking," *IEEE Trans. Signal Processing*, vol. 44, no. 1, pp. 95–107, Jan. 1995.

[11] Q. G. Liu and B. Champagne, "An adaptive ESPRIT algorithm based on perturbation of unsymmetrical matrices," in *Proc. 8th EUSIPCO*, Trieste, Italy, Sept. 1996, pp. 539–542.

[12] R. A. Horn and C. R. Johnson, *Matrix analysis*, Cambridge University Press, Cambridge, 1985.

[13] Y. Miao and Y. Hua, "Fast subspace tracking and neural network learning by a novel information criterion," *IEEE Trans. Signal Processing*, vol. 46, no. 7, pp. 1967–1979, July 1998.

[14] K. Abed, A. Chkeif, and Y. Hua, "Fast orthonormal PAST algorithm," *IEEE Sig. Proc. Letters*, vol. 7, no. 3, pp. 60–62, Mar. 2000.

[15] S. C. Douglas, "Numerically-robust adaptive subspace tracking using Householder transformations," in *SAM'00*. 2000, pp. 499–503, IEEE.

[16] R. Badeau, B. David, and G. Richard, "Yet Another Subspace Tracker," in *Proc. of ICASSP'05*. Mar. 2005, vol. 4, pp. 329–332, IEEE.

[17] R. Badeau, B. David, and G. Richard, "Fast Approximated Power Iteration Subspace Tracking," *IEEE Trans. Signal Processing*, vol. 53, no. 8, Aug. 2005, to be published.

[18] J. Laroche, "The use of the Matrix Pencil method for the spectrum analysis of musical signals," *Journal of the Acoustical Society of America*, vol. 94, no. 4, pp. 1958–1965, Oct. 1993.