HRHATRAC Algorithm for Spectral Line Tracking of Musical Signals

Bertrand David, Roland Badeau, Gael Richard

HRHATRAC ALGORITHM FOR SPECTRAL LINE TRACKING OF MUSICAL SIGNALS

Bertrand DAVID, Roland BADEAU and Gaël RICHARD

GET-Télécom Paris - Département TSI
46 rue Barrault - 75634 PARIS Cedex 13 FRANCE
bertrand.david@enst.fr

ABSTRACT

HRHATRAC combines the last improvements regarding the fast subspace tracking algorithms with a gradient update for adapting the signal poles estimates. It leads to a line spectral tracker which is able to robustly estimate the frequencies, even in a noisy context, when the lines are close to each other and when a modulation occurs. HRATRAC is also successfully applied in this paper to a piano note recording.

1. INTRODUCTION

HRHATRAC stands for *High Resolution HArmonics TRACking* and denotes an algorithm aiming at modeling musical sounds as multiple, slowly varying, spectral lines surrounded by an additive noise. Numerous existing systems devoted to this task are Fourier based, similarly to the classical works of McAuley and Quatieri [1] and Serra [2]. They often *a posteriori* process a set of frequency candidates obtained from a short time representation, in order to link them from frame to frame and finally extract long-term sinusoidal components with varying amplitude and frequency. Several techniques have been utilized for this purpose, such as dynamic programming [3], HMM [4] or particle filtering [5].

To avoid the inherent trade-off of Fourier based methods between observation length and accuracy, the algorithm proposed in this paper relies on subspace analysis (sometimes referred to as High Resolution methods in the context of time series spectral analysis), which locally precisely matches the model of a sum of sinusoidal components and white noise. These methods are sparsely used in the context of audio signal processing partly because of their high computational cost. However, a lot of progress has been made in the last decade to design low-cost, adaptive subspace trackers, reaching a linear complexity [6, 7, 8]. It is thus possible to efficiently update the major subspace (*e.g.* the signal subspace) and moreover, to update a so called spectral matrix whose eigenvalues are the signal poles [9]. HRHATRAC is a whole processing system for spectral line tracking, including an *ad hoc* prefiltering, an adaptive ESPRIT method, and a gradient descent for adapting the frequency estimates.

2. PRINCIPLE

Let $z_k = e^{-\delta_k + j2\pi f_k}, k = 0 \dots r-1$ be the complex poles of a noiseless harmonic signal, $(\delta_k, f_k) \in \mathbb{R}^+ \times [-0.5\; 0.5]$, and $\mathbf{x}(t)$ an n-dimensional vector ($n \geq r$) of data at time $t \in \mathbb{Z}$. Subspace analysis relies on the following property of the vector $\mathbf{x}(t)$: $\forall t \in \mathbb{Z}$ it belongs to the r-dimensional subspace spanned by the basis $\{\mathbf{v}(z_k)\}_{k=0\dots r-1}$; $\mathbf{v}(z) = \begin{bmatrix} 1 & z & \dots & z^{n-1} \end{bmatrix}^T$ being the Vandermonde vector associated with a non-zero complex number z. As a corollary, $\mathbf{v}(z_k) \perp \operatorname{span}(\mathbf{W}_\perp)$ where $\mathbf{W}$ denotes an $n \times r$ matrix spanning the signal subspace and $\mathbf{W}_\perp$ an $n \times (n-r)$ matrix spanning its orthogonal complement, referred to as the noise subspace.

A number of subspace analysis methods obtain the matrix $\mathbf{W}$ from observed data by means of a Singular Value Decomposition (SVD) of the covariance matrix. More generally, even if no structural property is required, the convenient choice of an orthonormal matrix is often made. Let $\mathbf{V} = \begin{bmatrix} \mathbf{v}(z_0) & \mathbf{v}(z_1) & \dots & \mathbf{v}(z_{r-1}) \end{bmatrix}$ be the Vandermonde matrix of the z_k's. $\mathbf{V}$ and $\mathbf{W}$ span the same subspace but $\mathbf{V}$ also satisfies the Rotational Invariance Property: $\mathbf{V}_\downarrow \mathbf{D} = \mathbf{V}_\uparrow$ where $\mathbf{D} = \operatorname{diag}(z_0, \dots, z_{r-1})$ and $\mathbf{V}_\downarrow$(*resp.* $\mathbf{V}_\uparrow$) contains the $n-1$ first (*resp.* last) rows of $\mathbf{V}$. This leads to the well known ESPRIT method in which the z_k's are estimated as the eigenvalues of the spectral matrix $\mathbf{\Phi} = \mathbf{W}_\downarrow^\dagger \mathbf{W}_\uparrow$ where the superscript † denotes the pseudo-inverse.

In the context of musical signal processing, the frequency parameters f_k are assumed to evolve slowly with time and the data are corrupted by an additive noise (at first considered as white). Consequently, the signal subspace becomes a function of t, and the corresponding matrix will be noted $\mathbf{W}(t)$. HRHATRAC implements a linear complexity subspace tracker (FAPI, *cf.* [7]) to update $\mathbf{W}(t)$ and $\mathbf{\Phi}(t)$ and tracks the pole estimates with the help of a gradient descent precisely initialized.

3. THE SUBSPACE TRACKER

The Fast Approximated Power Iteration algorithm described in [7] declined in its exponential version starts by a rank-one update of the covariance matrix:

$$\mathbf{C}_{xx}(t) = \beta\mathbf{C}_{xx}(t-1) + \mathbf{x}(t)\mathbf{x}(t)^H \qquad (1)$$

where the superscript ^{H} stands for the transpose conjugate. The power iteration method tracks the rectangular matrix $\mathbf{W}(t)$ spanning the dominant subspace of $\mathbf{C}_{xx}(t)$, using for each time increment a two steps iteration:

1. (C) compression: $\mathbf{C}_{xy}(t) = \mathbf{C}_{xx}(t)\mathbf{W}(t-1)$,
2. (O) orthonormalization: $\mathbf{W}(t)\mathbf{R}(t) = \mathbf{C}_{xy}(t)$,

where the step (O) decomposes $\mathbf{C}_{xy}(t)$ into the product between the orthonormal matrix $\mathbf{W}(t)$ and a square matrix $\mathbf{R}(t)$, which can be triangular or positive definite for example. The latter form is used in FAPI and $\mathbf{R}(t)^H$ is derived as the positive definite square root of $\mathbf{C}_{xy}(t)^H\mathbf{C}_{xy}(t)$. When $\mathbf{W}(t)$ and $\mathbf{W}(t-1)$ both span the range space of $\mathbf{C}_{xx}(t)$ this leads to the polar decomposition

$$\mathbf{R}(t)^H = (\mathbf{W}(t-1)^H\mathbf{C}_{xx}(t)\mathbf{W}(t-1))\mathbf{\Theta}(t) \qquad (2)$$

where

$$\mathbf{\Theta}(t) \overset{\Delta}{=} \mathbf{W}(t-1)^H\mathbf{W}(t). \qquad (3)$$

When dealing with noisy, possibly varying signals, $\mathbf{W}(t-1)$ does not span exactly the range space of $\mathbf{C}_{xx}(t)$. The relation (2) becomes an approximation and

$$\mathbf{W}(t) \approx \mathbf{W}(t-1)\mathbf{\Theta}(t). \qquad (4)$$

Equation (4) is interpreted as *a projection approximation* since $\mathbf{\Theta}$ defined as in (3) yields to the best approximation in the sense of the Frobenius norm of $\mathbf{W}(t)$ in the range space of $\mathbf{W}(t-1)$. The rank-one update (1) is then propagated through the equations stream leading to a rank-one update of $\mathbf{W}(t)$ as described in table 1.

Table 1. Exponential window Fast API (FAPI) algorithm

Initialization :

$$\boldsymbol{W}(0) = \begin{bmatrix} \boldsymbol{I}_r \\ \mathbf{0}_{(n-r)\times r} \end{bmatrix}, \ \boldsymbol{Z}(0) = \boldsymbol{I}_r$$

For each time step do

Input vector : $\boldsymbol{x}(t)$

FAPI main section	**Cost**
$\boldsymbol{y}(t) = \boldsymbol{W}(t-1)^H\boldsymbol{x}(t)$	nr
$\boldsymbol{h}(t) = \boldsymbol{Z}(t-1)\,\boldsymbol{y}(t)$	$r2$
$\boldsymbol{g}(t) = \frac{\boldsymbol{h}(t)}{\beta+\boldsymbol{y}(t)^H\boldsymbol{h}(t)}$	$2r$
$\varepsilon^2(t) = \Vert\boldsymbol{x}(t)\Vert^2 - \Vert\boldsymbol{y}(t)\Vert^2$	$n+r$
$\tau(t) = \frac{\varepsilon^2(t)}{1+\varepsilon^2(t)\Vert\boldsymbol{g}(t)\Vert^2+\sqrt{1+\varepsilon^2(t)\Vert\boldsymbol{g}(t)\Vert^2}}$	r
$\eta(t) = 1 - \tau(t)\,\Vert\boldsymbol{g}(t)\Vert^2$	1
$\boldsymbol{y}'(t) = \eta(t)\,\boldsymbol{y}(t) + \tau(t)\,\boldsymbol{g}(t)$	$2r$
$\boldsymbol{h}'(t) = \boldsymbol{Z}(t-1)^H\boldsymbol{y}'(t)$	r^2
$\boldsymbol{\epsilon}(t) = \frac{\tau(t)}{\eta(t)}\left(\boldsymbol{Z}(t-1)\boldsymbol{g}(t) - \left(\boldsymbol{h}'(t)^H\boldsymbol{g}(t)\right)\boldsymbol{g}(t)\right)$	r^2+3r
$\boldsymbol{Z}(t) = \frac{1}{\beta}\left(\boldsymbol{Z}(t-1) - \boldsymbol{g}(t)\,\boldsymbol{h}'(t)^H + \boldsymbol{\epsilon}(t)\,\boldsymbol{g}(t)^H\right)$	$2r^2$
$\boldsymbol{e}(t) = \eta(t)\,\boldsymbol{x}(t) - \boldsymbol{W}(t-1)\,\boldsymbol{y}'(t)$	$nr+n$
$\boldsymbol{W}(t) = \boldsymbol{W}(t-1) + \boldsymbol{e}(t)\,\boldsymbol{g}(t)^H$	nr

4. TRACKING OF THE SPECTRAL MATRIX

Taking the rotational invariance property of the signal subspace into account leads Roy [10] to the design of the ESPRIT method. In [9], the spectral matrix $\mathbf{\Phi}(t)$, defined in section 2, whose eigenvalues are the z_k's, is rewritten as

$$\mathbf{\Phi}(t) = \underbrace{(\mathbf{W}_\downarrow(t)^H\mathbf{W}_\downarrow(t))^{-1}}_{\mathbf{\Omega(t)}}\underbrace{\mathbf{W}_\downarrow(t)^H\mathbf{W}_\uparrow(t)}_{\mathbf{\Psi(t)}}.$$

Following the results of the preceding section, the update of $\mathbf{W}(t)$ is of the form $\mathbf{W}(t) = \mathbf{W}(t-1) + \mathbf{e}(t)\,\mathbf{g}(t)^H$ (see table 1). Let define the intermediary vectors

$$\begin{aligned} \mathbf{e}_-(t) &= \mathbf{W}_\downarrow(t-1)^H\mathbf{e}_\uparrow(t) \\ \mathbf{e}_+(t) &= \mathbf{W}_\uparrow(t-1)^H\mathbf{e}_\downarrow(t) \\ \mathbf{e}'_+(t) &= \mathbf{e}_+(t) + \mathbf{g}(t)(\mathbf{e}_\uparrow(t)^H\mathbf{e}_\downarrow(t)) \end{aligned}$$

and $\boldsymbol{\nu}(t)$ the transpose conjugate of the last row of $\mathbf{W}(t)$. The matrix $\mathbf{\Psi}(t)$ is recursively calculated according to

$$\mathbf{\Psi}(t) = \mathbf{\Psi}(t-1) + \mathbf{e}_-(t)\mathbf{g}(t)^H + \mathbf{g}(t)\mathbf{e}'_+(t).$$

This leads to an update formula for $\mathbf{\Phi}(t)$:

$$\mathbf{\Phi}(t) = \mathbf{\Psi}(t) + \frac{1}{1-\parallel \boldsymbol{\nu}(t) \parallel^2}\boldsymbol{\nu}(t)\mathbf{\Psi}(t)\boldsymbol{\nu}(t)^H.$$

These results are gathered in table 2.

Table 2. Adaptive computation of the spectral matrix

	Cost
$\boldsymbol{e}_-(t) = \boldsymbol{W}_\downarrow(t-1)^H\boldsymbol{e}_\uparrow(t)$	nr
$\boldsymbol{e}_+(t) = \boldsymbol{W}_\uparrow(t-1)^H\boldsymbol{e}_\downarrow(t)$	nr
$\boldsymbol{e}'_+(t) = \boldsymbol{e}_+(t) + \boldsymbol{g}(t)\ \left(\boldsymbol{e}_\uparrow(t)^H\boldsymbol{e}_\downarrow(t)\right)$	n
$\boldsymbol{\Psi}(t) = \boldsymbol{\Psi}(t-1) + \boldsymbol{e}_-(t)\,\boldsymbol{g}(t)^H + \boldsymbol{g}(t)\,\boldsymbol{e}'_+(t)^H$	$2r^2$
$\boldsymbol{\varphi}(t) = \boldsymbol{\Psi}(t)^H\boldsymbol{\nu}(t)$	r^2
$\boldsymbol{\Phi}(t) = \boldsymbol{\Psi}(t) + \frac{1}{1-\Vert\boldsymbol{\nu}(t)\Vert^2}\,\boldsymbol{\nu}(t)\,\boldsymbol{\varphi}(t)^H$	r^2

5. ADAPTING THE POLES

An EigenValue Decomposition (EVD) of $\mathbf{\Phi}(t)$ gives the estimates of the z_k's at each time step. A postprocessing is nevertheless necessary to compute the spectral lines, *i.e.* grouping the r frequency estimates $f_k = \angle z_k/(2\pi)$ into r classes. To avoid dilemnae on the choice of appropriate heuristics for modeling the spectral trajectories or for adjusting a weight function to link the r new $f_k(t)$ to the r old values, a gradient approach is adopted below to directly update the z_k's. It consists of the sequential tracking of the diagonal matrix $\mathbf{\Lambda}(t)$ containing the eigenvalues of $\mathbf{\Phi}(t)$ in first place, then the tracking of the matrix $\mathbf{V}(t)$ containing the corresponding eigenvectors.

5.1. Iteration for $\mathbf{\Lambda}(t)$

The cost function to be minimized for estimating $\mathbf{\Lambda}(t)$ is defined as the squared Frobenius norm of the estimation error:

$$J(\mathbf{\Lambda}) = \operatorname{tr}\{\mathbf{E}_L(\mathbf{\Lambda})^H \mathbf{E}_L(\mathbf{\Lambda})\}$$

where the estimation error is simply given by

$$\mathbf{E}_L(\mathbf{\Lambda}) = \mathbf{\Lambda} - \operatorname{diag}\left(\mathbf{V}(t-1)^{-1}\mathbf{\Phi}(t)\mathbf{V}(t-1)\right).$$

The gradient descent of $J(\mathbf{\Lambda})$ leads to an adaptation in the direction opposite to the estimation error:

$$\mathbf{\Lambda}(t) = (1-\mu_L)\mathbf{\Lambda}(t-1) + \mu_L \operatorname{diag}\left(\mathbf{V}(t-1)^{-1}\mathbf{\Phi}(t)\mathbf{V}(t-1)\right)$$

where $0 < \mu_L < 1$.

5.2. Iteration for $\mathbf{V}(t)$

Once $\mathbf{\Lambda}(t)$ is estimated, the cost function to be minimized for the estimation of $\mathbf{V}(t)$ is defined as the squared Frobenius norm of the estimation error: $J(\mathbf{V}) = \operatorname{tr}\{\mathbf{E}_V(\mathbf{V})^H \mathbf{E}_V(\mathbf{V})\}$ where the estimation error is now defined as

$$\mathbf{E}_V(\mathbf{V}) = \mathbf{V} - \mathbf{\Phi}(t)\mathbf{V}\mathbf{\Lambda}(t)^{-1}.$$

The expression of the gradient of $J(\mathbf{V})$ with respect to $\mathbf{V}$ (element by element, including the complex case) leads to the following recursion[1]:

$$\mathbf{V}(t) = (1-\mu_V)\mathbf{V}(t-1) +$$
$$\mu_V\left(\mathbf{\Phi}(t)\mathbf{V}(t-1)\mathbf{\Lambda}(t)^{-1} + \mathbf{\Phi}(t)^H \mathbf{E}_V(\mathbf{V}(t-1))\mathbf{\Lambda}(t)^{-H}\right)$$

where $0 < \mu_V < 1$. Note that for better numerical stability, the columns of $\mathbf{V}(t)$ can be normalized after each iteration.

6. SIMULATIONS AND PERFORMANCE

The whole algorithm has a linear complexity of $5nr$. This makes the system suitable for real time applications. Since musical signal analysis is targeted, the testing scenario is based on modulations and frequency resolution trade-off encountered in the field. The results are compared to those of an exact EVD of the spectral matrix $\mathbf{\Phi}$, denoted by the superscript $^{\text{EVD}}$. For instance, $\hat{f}_k^{\text{EVD}}$ is the f_k estimate using an exact EVD. Note that this estimation leads to a *pointwise* representation of the frequency estimates which in this testing example will be used as the baseline estimation.

The signal is a sum of two close modulated components and an additive white complex noise $b(t)$:

$$x(t) = U(t-t_0)(\exp(j\phi_1(t) + \exp(j\phi_2(t)) + b(t), \quad (5)$$

where $f_1(t) = \phi_1'(t)/(2\pi) = 0.1(1 + 0.1\cos(2\pi f_m t))$, $f_m = 5{\scriptstyle 10^{-4}}$ and $f_2(t) = 1.1 f_1(t)$. $U(t)$ denotes the unit step function. The SNR is fixed to 9 dB. The subspace tracker uses the parameters $n = 31$, $r = 2$ and $\beta = 0.99$ while the gradient steps μ_L and μ_V both equal 0.99. The results are represented in figure 1. The spectral lines obtained by HRHATRAC nearly coincide with the $\hat{f}_k^{\text{EVD}}$ estimates in the time range where a sufficient degree of convergence is reached. The noticeable differences with the ground truth can thus be attributed to the deviation of the observed data from the model of stationary sinusoidal components.

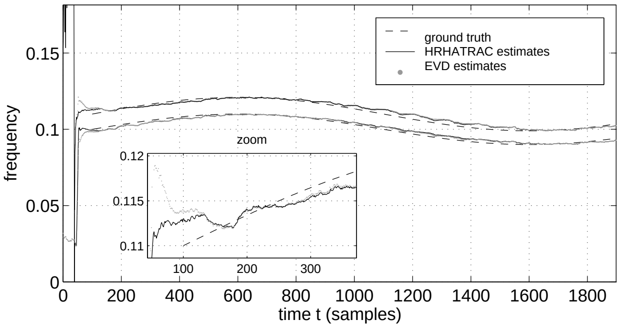

Fig. 1. Results for 2 modulated sinusoids.

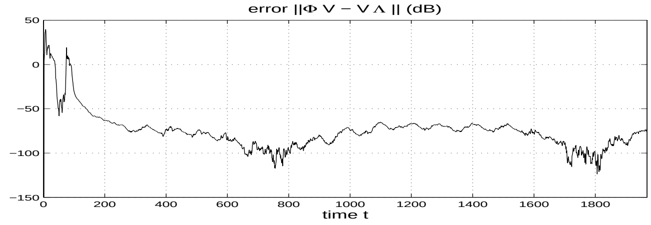

Fig. 2. Representation of the error $\|\mathbf{\Phi}(t)\mathbf{V}(t) - \mathbf{V}(t)\mathbf{\Lambda}(t)\|$ in the coupled estimation of the eigenvalues matrix $\mathbf{\Lambda}(t)$ and the eigenbasis $\mathbf{V}(t)$.

Figure 2 shows the gradient capability to accurately track the eigenvalues and eigenvectors of the spectral matrix. This plot demonstrates the fast convergence property of the gradient method[2], since the error on the eigenparameters abruptly falls after the start of both sinusoids. The frequency estimation is then comparable to that of an exact ESPRIT solution. Moreover, some favorable aspects of the whole algorithm performance are revealed in this example: despite a tight frequency discrepency between the two components (under the Fourier resolution) and the low SNR compared to that of usual audio signals, the estimates remain accurately locked on the true spectral lines.

[1] in this demonstration, it is worth noting that for two matrices $\mathbf{A}$ and $\mathbf{B}$ $\operatorname{tr}(\mathbf{A}^H\mathbf{B})$ defines a scalar product $\mathbf{a}^H\mathbf{b}$ where $\mathbf{a}$ et $\mathbf{b}$ are the column vectors obtained by rearranging columnwise the coefficients of $\mathbf{A}$ and $\mathbf{B}$; the result can then be obtained by a first order perturbation of $J(\mathbf{V})$.

[2] We can obtain nearly the same results for μ_L and μ_V around 0.5

7. A REAL WORLD CASE

In this section, an E6 piano note is analyzed. The critical part of the work lies in the deviation that audio signals present from the hypothesis of white surrounding noise. A carefully designed preprocessing is presented below to circumvent this difficulty.

7.1. Preprocessing

The whitening of the noise is achieved in 4 steps:

1- a median filtering of the periodogram of the signal which leads to an estimate of the noise spectral density,

2- an inverse Fourier transform which provides an estimate of the autocovariance sequence,

3- an AR12 modeling of the noise spectrum,

4- a filtering of the data by a FIR filter defined as the inverse of the estimated AR model.

The signal power spectral density after whitening is represented in figure 3.

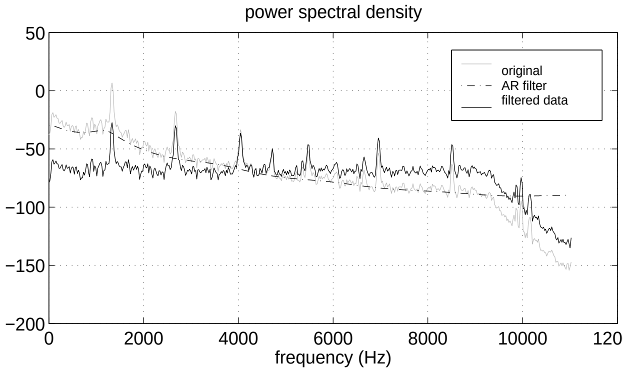

Fig. 3. Power spectral density before and after preprocessing.

7.2. Results

The figures 4 and 5 show the spectrogram and the HRHA-TRAC analysis of the piano note. The tracking is applied to the analytic complex signal associated to the preprocessed real signal with $r = 9, n = 101$. The spectral line tracking leads to 6 lines clearly identified as the 6 harmonics of the note and one line not in relation with the others (perhaps a compression harmonic). The eighth and ninth lines (in gray) are irregular enough to be labeled as "spurious".

8. CONCLUSIONS

A whole spectral line tracker has been presented. It benefits from the high resolution property of the subspace methods while being adaptive with a linear complexity. A successful application to a piano sound ends the presentation.

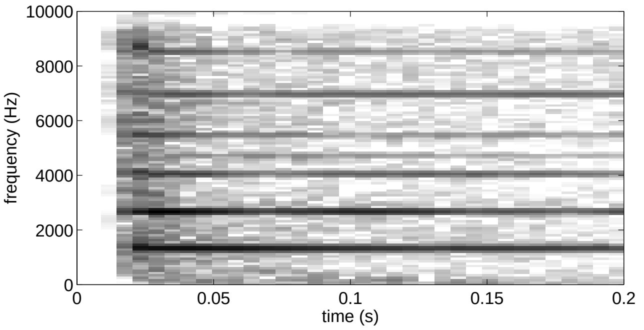

Fig. 4. Spectrogram of an E6 piano note for $N = 256$ and Hann windowing with 50% overlap.

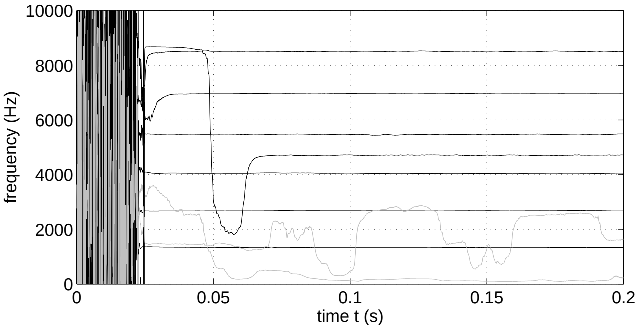

Fig. 5. Spectral line tracking for a E6 piano note.

9. REFERENCES

[1] R. J. McAulay and T. F. Quatieri, "Speech analysis/synthesis based on a sinusoidal representation," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-34, no. 4, pp. 744–754, Aug 1986.

[2] X. Serra and J. Smith, "Spectral modeling synthesis: A sound analysis-/synthesis system based on a deterministic plus stochastic decomposition," *Computer Music J.*, vol. 14, no. 4, pp. 12–24, Winter 1990.

[3] Mathieu Lagrange, Sylvain Marchand, and Jean-Bernard Rault, "Tracking partials for the sinusoidal modeling of polyphonic sounds," *Proc. IEEE ICASSP-05*, vol. III, pp. 229–232, 2005.

[4] P. Depalle, G. Garcia, and X Rodet, "Tracking of partials for additive sound synthesis using hidden Markov models," *Proc. IEEE ICASSP-93*, vol. 1, Apr 1993.

[5] C. Dubois, M. Davy, and J. Idier, "Tracking of time-frequency components using particle filtering," in *Proc. IEEE ICASSP*, Philadelphia, PA, USA, Mar. 2005.

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

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

[8] R. Badeau, B. David, and G. Richard, "Yet another subspace tracker," in *ICASSP'05*, Philadelphie, Pennsylvanie, USA, mar 2005, vol. 4, pp. 329–332.

[9] R. Badeau, G. Richard, and B. David, "Fast adaptive esprit algorithm," in *SSP'05*, Bordeaux, France, jul 2005.

[10] 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.