Efficient Time-Recursive Implementation of Matched Filterbank Spectral Estimators

Stephen R. Alty, *Member, IEEE*, Andreas Jakobsson, *Member, IEEE*, and Erik G. Larsson, *Member, IEEE*

***Abstract*—In this paper, we present a computationally efficient sliding window time updating of the Capon and [AU: PLEASE SPELL OUT APES?] (APES) matched filterbank spectral estimators based on the time-variant displacement structure of the data covariance matrix. The presented algorithm forms a natural extension of the most computationally efficient algorithm to date, and offers a significant computational gain as compared to the computational complexity associated with the batch re-evaluation of the spectral estimates for each time-update. Furthermore, via simulations, the algorithm is found to be numerically superior to the time-updated spectral estimate formed from directly updating the data covariance matrix.**

***Index Terms*—Adaptive filters, computational complexity, covariance matrices, iterative methods, spectrum analysis, time-varying systems.**

I. Introduction

SPECTRAL estimation finds applications in a wide range of fields, and has received a vast amount of interest in the literature over the last century. Due to their inherent robustness to model assumptions, there has lately been a renewed interest in *nonparametric* spectral estimators. Among the nonparametric approaches, the *data-dependent* filterbank spectral estimators have many promising properties, allowing for very accurate, computationally efficient, high-resolution estimates (see, e.g., [1] and the references therein). Both the recent **[AU: PLEASE SPELL OUT APES?]** (APES) estimator [2] and the *amplitude spectrum Capon* (ASC) estimator, i.e., the estimator obtained when using the classical Capon filter [3] to estimate a sinusoidal component at the center frequency of the bandpass filter, can be seen as *matched* filterbank methods [4]. Given the excellent performance of these estimators, several authors have worked on finding efficient implementations (see [1] for references); the most efficient implementation to date was presented in [5]. This implementation is based on the evaluation of the inverse Cholesky factors of the sample covariance matrix using its inherent displacement structure [6]. Given its low displacement rank, these Cholesky factors can be obtained efficiently using the generalized Schur recursion. In several applications, one has an interest in time updating the spectral estimate as additional samples becomes available. Some effort has been made to introduce such an updating for the matched filterbank methods [7], [8], but neither of these methods exploit the full structure of the estimator, resulting in updates having higher computational complexity than that which is required to recompute the spectral estimate using the method in [5].

In this work, we present a novel computationally efficient approach to time updating the efficient estimator in [5] using a sliding window update of the measured data. The presented update is based on the time-variant displacement structure, allowing for the time updating of the inverse Cholesky factors of the (forward-backward averaged) covariance matrix estimate using the numerically robust time-variant generalized Schur algorithm presented in [9]. The resulting time-updated spectral estimates offers a significant computational gain as compared with the previously required recalculation of the Cholesky factors of the covariance matrix in each time step. Furthermore, via simulations, the method is found to be numerically superior, yielding a lower error propagation, as compared to a time-updated spectral estimate formed from a more direct (but computationally costlier) recursive updating of the sample covariance matrix.

The paper is organized as follows: in Section II, we briefly review the class of matched filterbank spectral estimators. In Section III, we discuss the proposed efficient time updating. Then, Section IV includes some numerical simulations indicating the computational gain and error propagation behavior of the proposed method, and Section V contains our conclusions.

II. Matched Filterbank Spectral Estimators

The matched filterbank spectral estimators are constructed from a set of data-adaptive frequency-dependent, L-tap finite-impulse response (FIR) filters $\mathbf{h}_\omega$, such that [2], [4]

$$\min_{\mathbf{h}_\omega} \mathbf{h}_\omega^* \mathbf{Q}_\omega \mathbf{h}_\omega \text{ subject to } \mathbf{h}_\omega^* \mathbf{a}_\omega = 1 \tag{1}$$

where $\mathbf{Q}_\omega$ is the $L \times L$ covariance matrix of the signal consisting of all frequencies except ω, $(\cdot)^*$ denotes the conjugate transpose, and $\mathbf{a}_\omega$ is an L-tap Fourier vector, i.e.,

$$\mathbf{a}_\omega = \begin{bmatrix} 1 & e^{i\omega} & \dots & e^{i\omega(L-1)} \end{bmatrix}^T. \tag{2}$$

The classical Capon filter is obtained by minimizing (1) using the covariance matrix of the measured data as an estimate of $\mathbf{Q}_\omega$, i.e.,

$$\mathbf{Q}_\omega^{\text{Capon}} = \mathbf{R}_x \triangleq E\{\mathbf{x}_t \mathbf{x}_t^*\} \tag{3}$$

Manuscript received April 28, 2004. This paper was recommended by Associate Editor H. Lev-Ari.

S. R. Alty is with the Centre for Digital Signal Processing Research, King's College London, London WC2R 2LS, U.K. (e-mail: steve.alty@kcl.ac.uk).

A. Jakobsson is with the Department of Electrical Engineering, Karlstad University, SE-651 88 Karlstad, Sweden (e-mail: andreas.jakobsson@ieee.org).

E. G. Larsson is with the Department of Electrical Engineering, George Washington University, Washington DC 20052 USA (e-mail: egl@gwu.edu).

Digital Object Identifier 10.1109/TCSI.2004.842876

where

$$\mathbf{x}_t = [\,x(t) \quad x(t+1) \quad \ldots \quad x(t+L-1)\,]^T. \tag{4}$$

Similarly, the APES filter is obtained by minimizing (1) using

$$\mathbf{Q}_\omega^{\mathrm{APES}} = \mathbf{R}_x - \mathbf{y}_\omega \mathbf{y}_\omega^* \tag{5}$$

where

$$\mathbf{y}_\omega = \frac{1}{M}\sum_{t=1}^{M} \mathbf{x}_t e^{-i\omega t}. \tag{6}$$

Here, $M = N - L + 1$, where N is the total number of samples used to form the spectral estimate. We remark that the choice of L is a compromise between resolution and statistical stability: the larger L, the better the resolution but the higher the variance. Furthermore, a larger L increases the dimension of $\mathbf{R}_x$, and thus, the computational burden of evaluating the spectral estimate. The corresponding (amplitude) spectral estimate is obtained as [2], [4]

$$\hat{\phi}_x(\omega) = \mathbf{h}_\omega^* \mathbf{y}_\omega = \frac{\mathbf{a}_\omega^* \mathbf{Q}_\omega^{-1} \mathbf{y}_\omega}{\mathbf{a}_\omega^* \mathbf{Q}_\omega^{-1} \mathbf{a}_\omega}. \tag{7}$$

It is worth noting that (7) in combination with (3), the so-called ASC estimator is in general different from the classical *power-spectrum Capon* (PSC) estimator, which is formed as

$$\hat{\phi}_x^{\mathrm{PSC}}(\omega) = \frac{1}{\mathbf{a}_\omega^* \mathbf{R}_x^{-1} \mathbf{a}_\omega}. \tag{8}$$

Further, using the matrix inversion formula, one may write both (7) and (8) using a number of matrix-vector multiplications and Fourier transforms of the inverse Cholesky factor of $\mathbf{R}_x$ (see also [1], [4]). This fact is exploited in the efficient implementation of (7) and (8) presented in [5], which is based on the inherent displacement structure of $\mathbf{R}_x$ to efficiently evaluate the inverse Cholesky factors of $\mathbf{R}_x$ using the generalized Schur algorithm (see, e.g., [6]); together with efficient matrix-vector multiplications and the fast Fourier transform (FFT) this forms the efficient implementation. As $\mathbf{R}_x$ is typically unknown, one needs to estimate it; this is commonly achieved using the forward–backward averaged outer-product estimate (see [10] for a more detailed discussion on the benefits of this estimator as compared to the forward-only estimator)

$$\hat{\mathbf{R}}_x^{fb} = \frac{1}{2}\left(\hat{\mathbf{R}}_x + \mathbf{J}\hat{\mathbf{R}}_x^T \mathbf{J}\right) \tag{9}$$

where $\mathbf{J}$ is the $L \times L$ exchange (or reversal) matrix formed as

$$\mathbf{J} = \begin{bmatrix} \mathbf{0} & & 1 \\ & \iddots & \\ 1 & & \mathbf{0} \end{bmatrix} \tag{10}$$

$$\hat{\mathbf{R}}_x = \frac{1}{M}\sum_{t=1}^{M} \mathbf{x}_t \mathbf{x}_t^*. \tag{11}$$

Herein, we consider the problem of time updating $\hat{\phi}_x(\omega)$ as additional data samples become available, by exploiting the time-invariant displacement structure of $\hat{\mathbf{R}}_x^{fb}$ to efficiently form a time-update of the inverse Cholesky factors. To find this updating, we first need an expression to update the Cholesky factor itself.

III. Time Updating Cholesky Factors

Numerous signal processing problems form matrices exhibiting a significant degree of *s*tructure. This structure can be exploited to reduce the computational burden as well as the memory requirements for operations on such matrices. In this work, we focus on the displacement structure of the sample covariance matrix to find an efficient *time updating* algorithm. A time-variant Toeplitz-like $L \times L$ matrix $\mathbf{R}_x(t)$ is said to have a time-variant displacement structure if the matrix difference $\nabla\mathbf{R}_x(t)$ defined by [6], [9]

$$\nabla\mathbf{R}_x(t) = \mathbf{R}_x(t) - \mathbf{F}(t)\mathbf{R}_x(t-\Delta)\mathbf{F}^*(t) \tag{12}$$

has low *rank*, say $r(t)$, where $r(t) \ll L$, for some lower triangular matrix $\mathbf{F}(t)$. The time-variant displacement rank, $r(t)$, provides a measure of the degree of structure present, with lower rank indicating stronger structure. Thus, if $r(t)$ is close to L, there is little point in pursuing the displacement framework. We note that the sliding window time updating of the estimated forward–backward covariance matrix estimate can be expressed as

$$\hat{\mathbf{R}}_x^{fb}(t) = \hat{\mathbf{R}}_x^{fb}(t-1) + \mathbf{G}(t)\mathbf{J}(t)\mathbf{G}^*(t) \tag{13}$$

where $\mathbf{G}(t)$ and $\mathbf{J}(t)$ are given below, allowing for a time-variant displacement structure with $\Delta = 1$ and $\mathbf{F}(t) = \mathbf{I}$. Combining (12) and (13), yields

$$\nabla\hat{\mathbf{R}}_x^{fb}(t) = \mathbf{G}(t)\mathbf{J}(t)\mathbf{G}^*(t) \tag{14}$$

where $\mathbf{G}(t)$ is an $L \times r(t)$ *generator* matrix and $\mathbf{J}(t)$ is an $r(t) \times r(t)$ full rank diagonal signature matrix with either ± 1 s along its diagonal. Here, (15), shown at the bottom of the page, holds, and

$$\mathbf{J}(t) = \begin{bmatrix} +1 & 0 & 0 & 0 \\ 0 & +1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & -1 \end{bmatrix}. \tag{16}$$

From (16), we note that $r(t) = 4$ for the forward-backward covariance matrix estimate. This value of $r(t)$ can often be significantly less than typical values of L, which depending on the

$$\mathbf{G}(t) = \begin{bmatrix} x^*(t+N) & x(t+M) & x^*(t+L-1) & x(t) \\ x^*(t+N-1) & x(t+M+1) & x^*(t+L-2) & x(t+1) \\ x^*(t+N-2) & x(t+M+2) & x^*(t+L-3) & x(t+2) \\ \vdots & \vdots & \vdots & \vdots \\ x^*(t+M) & x(t+N) & x^*(t) & x(t+L-1) \end{bmatrix} \tag{15}$$

application usually is very large. We note that the positive-definite nature of $\hat{\mathbf{R}}_x^{fb}(t)$ guarantees the existence of a unique (lower triangular) Cholesky factor, $\mathbf{L}(t)$, such that

$$\hat{\mathbf{R}}_x^{fb}(t) = \mathbf{L}(t)\mathbf{L}^*(t) \tag{17}$$

which, exploiting (13), can be expressed as [9]

$$\begin{aligned}[\mathbf{L}(t) \quad \mathbf{0}] &\begin{bmatrix} \mathbf{L}^*(t) \\ \mathbf{0} \end{bmatrix} \\ &= [\mathbf{L}(t-1) \quad \mathbf{G}(t)] \begin{bmatrix} \mathbf{I}_n & \mathbf{0} \\ \mathbf{0} & \mathbf{J}(t) \end{bmatrix} \begin{bmatrix} \mathbf{L}^*(t-1) \\ \mathbf{G}^*(t) \end{bmatrix}. \end{aligned} \tag{18}$$

Hence, it follows that there exists an $[\mathbf{I}_n \oplus \mathbf{J}(t)]$-unitary *rotation* matrix[1], $\mathbf{\Gamma}(t)$, such that

$$[\mathbf{L}(t) \quad \mathbf{0}] = [\mathbf{L}(t-1) \quad \mathbf{G}(t)]\mathbf{\Gamma}(t). \tag{19}$$

Note that $\mathbf{\Gamma}(t)$ has the effect of rotating the generator matrix onto the expression $\mathbf{L}(t-1)$ to produce the updated Cholesky factor $\mathbf{L}(t)$ and a block zero entry in the left-hand side of (19). The rotational transform $\mathbf{\Gamma}(t)$ is typically implemented as a sequence of elementary transforms, such that $\mathbf{\Gamma}(t) = \mathbf{\Gamma}^1(t)\mathbf{\Gamma}^2(t)\dots\mathbf{\Gamma}^L(t)$, where $\mathbf{\Gamma}^k(t)$ annihilates the kth row of the generator matrix, e.g.,[2]

$$\begin{aligned}\begin{bmatrix} l & 0 & 0 & g & g \\ l & l & 0 & g & g \\ l & l & l & g & g \end{bmatrix} &\xrightarrow{\mathbf{\Gamma}^1(t)} \begin{bmatrix} l' & 0 & 0 & 0 & 0 \\ l' & l & 0 & g' & g' \\ l' & l & l & g' & g' \end{bmatrix} \\ &\xrightarrow{\mathbf{\Gamma}^2(t)} \dots \xrightarrow{\mathbf{\Gamma}^L(t)} \begin{bmatrix} l' & 0 & 0 & 0 & 0 \\ l' & l' & 0 & 0 & 0 \\ l' & l' & l' & 0 & 0 \end{bmatrix}. \end{aligned} \tag{20}$$

Note how the sequence of rotations in (20) updates one column of the Cholesky factor at a time, leaving the lower ranks unchanged. Further, note that the remaining rows of the generator matrix are also updated, this to enable the updating of the next column of the Cholesky factor in turn. This procedure continues until all the L ranks of $\mathbf{L}(t-1)$ have been updated to $\mathbf{L}(t)$ and the entire generator matrix, $\mathbf{G}(t)$ has been completely nullified. In this way, the updated columns of the Cholesky factor are evolved in an efficient recursive manner. We remark that such a recursion is also beneficial for efficient use of memory allocation and numerical contraction during matrix-vector products.

[1]Here, a $\mathbf{J}$-unitary matrix $\mathbf{\Theta}$ is defined as any matrix $\mathbf{\Theta}$ such that $\mathbf{\Theta}\mathbf{J}\mathbf{\Theta}^* = \mathbf{J}$. Further, $\mathbf{a} \oplus \mathbf{b}$ denotes a matrix with the submatrices $\mathbf{a}\,\{n \times n\}$ and $\mathbf{b}\,\{m \times m\}$ concatenated to produce a matrix of size $\{(m+n) \times (m+n)\}$.

[2]In this example, (20) shows a rank-2 generator matrix, where $L = 3$ and x' indicating time-updated elements of a given matrix.

The rotation matrix $\mathbf{\Gamma}(t)$ can be formed in numerous different ways. Here, we will use a combination of the *Householder* and *Givens* rotations. Both of these transforms have the general form

$$[a \quad b]\mathbf{\Theta} = [\alpha \quad 0] \tag{21}$$

where $\alpha_H = \sqrt{|a|^2 - |b|^2}$ for a Householder and $\alpha_G = \sqrt{|a|^2 + |b|^2}$ for a Givens rotation. The corresponding rotation matrices are given as

$$\mathbf{\Theta}_H = \frac{1}{\sqrt{|a|^2 - |b|^2}} \begin{bmatrix} a & -b \\ -b^* & a \end{bmatrix} \tag{22}$$

$$\mathbf{\Theta}_G = \frac{1}{\sqrt{|a|^2 + |b|^2}} \begin{bmatrix} a & b \\ b^* & -a \end{bmatrix}. \tag{23}$$

The Givens rotation is used for "*updating*" the factor with new samples and the Householder rotation has the effect of "*downdating*" the factor by removing those samples which are no longer present in the time-updated sample frame. In this way, an appropriate combination of rotations can be determined to correctly time-update each Cholesky factor column vector in turn. One should note tha,t in practice, each column of the Cholesky factor is concatenated with the generator matrix to make an $\{L-k+1\} \times \{r(t)+1\}$ matrix, see (24), and as each vector is updated, this process is repeated whilst each row of the generator matrix is annihilated until all the column vectors of the new Cholesky factor are produced

$$\begin{aligned}&\begin{bmatrix} l & g & g \\ l & g & g \\ l & g & g \end{bmatrix} \xrightarrow{\mathbf{\Gamma}^1(t)} \begin{bmatrix} l' & 0 & 0 \\ l' & g' & g' \\ l' & g' & g' \end{bmatrix} \text{ followed by} \\ &\begin{bmatrix} l & g' & g' \\ l & g' & g' \end{bmatrix} \xrightarrow{\mathbf{\Gamma}^2(t)} \begin{bmatrix} l' & 0 & 0 \\ l' & g'' & g'' \end{bmatrix} \text{ etc.} \end{aligned} \tag{24}$$

Thus, an appropriate rotation matrix $\mathbf{\Gamma}(t)$ is an $\{r(t)+1\} \times \{r(t)+1\}$ matrix of the form (25), shown at the bottom of the page, where

$$\begin{aligned}\alpha_k &= \sqrt{|l(k,k)|^2 + |g(k,1)|^2} \\ \beta_k &= \sqrt{|\alpha_k|^2 + |g(k,2)|^2} \\ \gamma_k &= \sqrt{|\beta_k|^2 - |g(k,3)|^2} \\ \delta_k &= \sqrt{|\gamma_k|^2 - |g(k,4)|^2}. \end{aligned}$$

Whilst the time updating of the Cholesky factor is interesting in itself, for instance for solving sets of linear equations by simple back-substitution or Gaussian elimination, it can not be used to

$$\mathbf{\Gamma}^k(t) = \begin{bmatrix} \frac{l(k,k)}{\delta_k} & \frac{g(k,1)}{\alpha_k} & \frac{l(k,k)\cdot g(k,2)}{\alpha_k \cdot \beta_k} & -\frac{l(k,k)\cdot g(k,3)}{\beta_k \cdot \gamma_k} & -\frac{l(k,k)\cdot g(k,4)}{\gamma_k \cdot \delta_k} \\ \frac{g^*(k,1)}{\delta_k} & -\frac{l(k,k)}{\alpha_k} & \frac{g^*(k,1)\cdot g(k,2)}{\alpha_k \cdot \beta_k} & -\frac{g^*(k,1)\cdot g(k,3)}{\beta_k \cdot \gamma_k} & -\frac{g^*(k,1)\cdot g(k,4)}{\gamma_k \cdot \delta_k} \\ \frac{g^*(k,2)}{\delta_k} & 0 & -\frac{\alpha_k}{\beta_k} & -\frac{g^*(k,2)\cdot g(k,3)}{\beta_k \cdot \gamma_k} & -\frac{g^*(k,2)\cdot g(k,4)}{\gamma_k \cdot \delta_k} \\ -\frac{g^*(k,3)}{\delta_k} & 0 & 0 & \frac{\beta_k}{\gamma_k} & \frac{g^*(k,3)\cdot g(k,4)}{\gamma_k \cdot \delta_k} \\ -\frac{g^*(k,4)}{\delta_k} & 0 & 0 & 0 & \frac{\gamma_k}{\delta_k} \end{bmatrix} \tag{25}$$

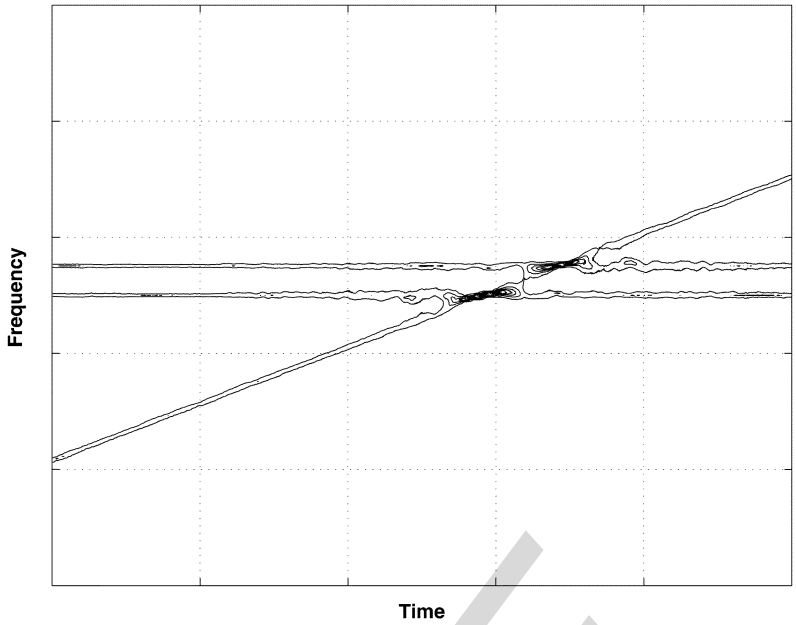

Fig. 1. ASC estimate of a time-varying signal.

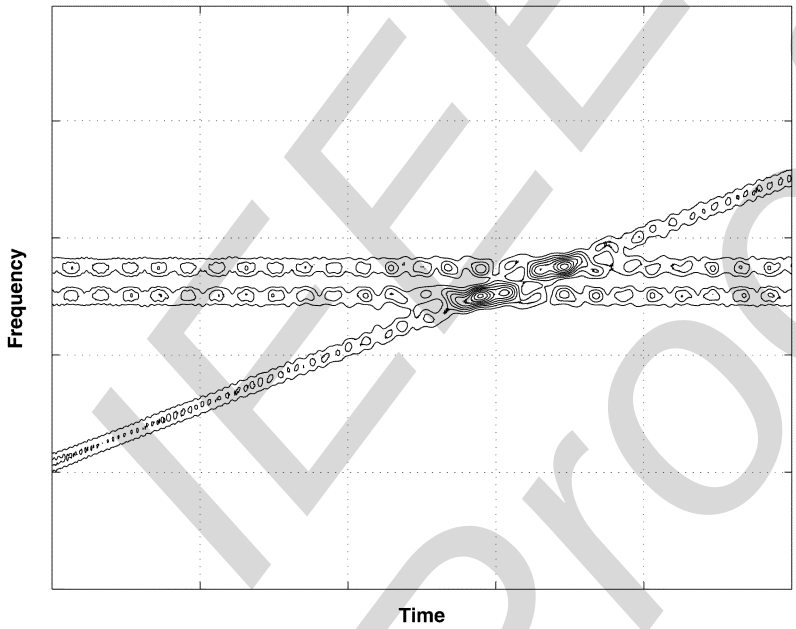

Fig. 2. Spectrogram estimate of a time-varying signal.

efficiently find the time-update of the filterbank spectral estimate. However, it is possible to extend the above procedure to also yield the *inverse* Cholesky factor by augmenting (20) as [9]

$$\begin{bmatrix} l & 0 & 0 & g & g \\ l & l & 0 & g & g \\ l & l & l & g & g \\ l^{-*} & l^{-*} & l^{-*} & 0 & 0 \\ 0 & l^{-*} & l^{-*} & 0 & 0 \\ 0 & 0 & l^{-*} & 0 & 0 \end{bmatrix}$$

$$\xrightarrow{\mathbf{\Gamma}^1(t)} \begin{bmatrix} l' & 0 & 0 & 0 & 0 \\ l' & l & 0 & g' & g' \\ l' & l & l & g' & g' \\ l'^{-*} & l^{-*} & l^{-*} & w & w \\ 0 & l^{-*} & l^{-*} & 0 & 0 \\ 0 & 0 & l^{-*} & 0 & 0 \end{bmatrix}$$

$$\xrightarrow{\mathbf{\Gamma}^2(t)} \cdots \xrightarrow{\mathbf{\Gamma}^L(t)} \begin{bmatrix} l' & 0 & 0 & 0 & 0 \\ l' & l' & 0 & 0 & 0 \\ l' & l' & l' & 0 & 0 \\ l'^{-*} & l'^{-*} & l'^{-*} & w'' & w'' \\ 0 & l'^{-*} & l'^{-*} & w' & w' \\ 0 & 0 & l'^{-*} & w & w \end{bmatrix}.$$

Here, the upper triangular (or transpose conjugate) of the inverse Cholesky factor, $\mathbf{L}^{-*}(t-1)$, have been appended below the ma-

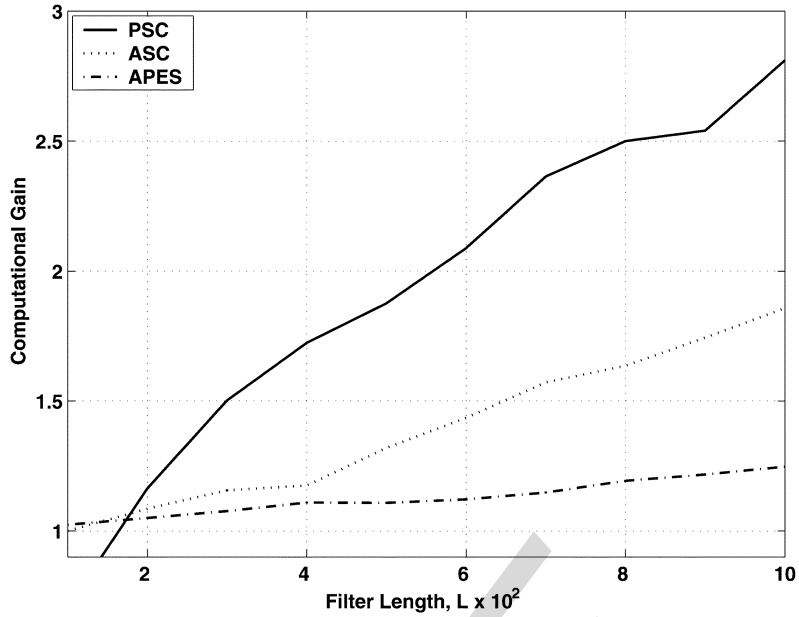

Fig. 3. Computational gain of the proposed time-updated PSC, ASC, and APES spectral estimators as compared to the brute-force re-evaluation using [5].

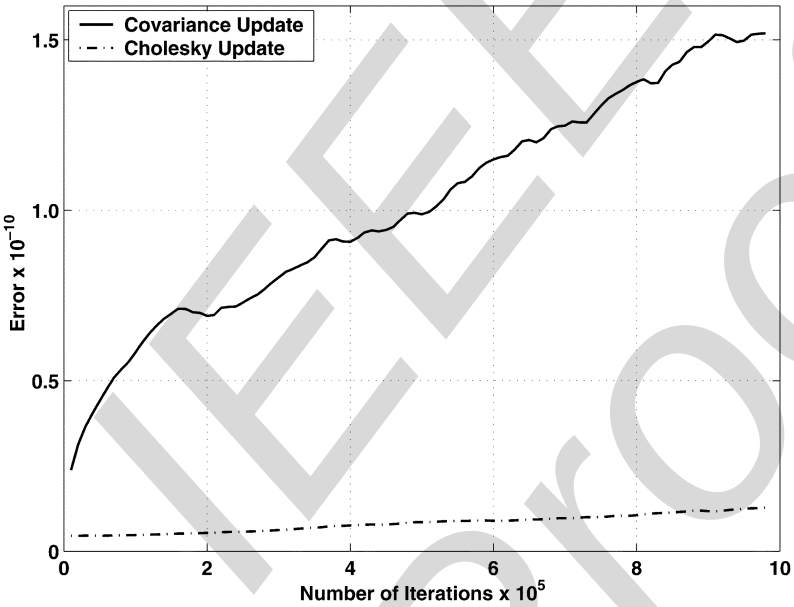

Fig. 4. Error propagation of the ASC spectral estimate using covariance updating vs. Cholesky updating.

trix in (20). Further, an $L \times r(t)$ matrix of zeros is also appended to produce the above $2L \times \{L + r(t)\}$ matrix. By applying the exact same rotation, $\mathbf{\Gamma}(t)$, we thus find an efficient time updating of the inverse Cholesky factor, yielding one column vector per iteration. As a side effect, the iteration produces an inverse generator matrix ($\mathbf{W}(t)$ in the above example) which must be stored as it is required to correctly update each vector in turn.

IV. Numerical Example

As an illustration of the superior quality of the filterbank spectral estimators, the ASC estimate is shown in Fig. 1 for a time-varying signal consisting of 3 complex-valued sinusoids with time-varying frequency. Here, the estimates are computed for 256 frequency points using $N = 64$ samples for each time step, using a $L = 16$-tap adaptive filter. As a comparison, Fig. 2 shows the (unwindowed) Spectrogram spectral estimate for the same data set, clearly illustrating the superior resolution of the ASC estimator. Fig. 3 illustrates the *computational gain* of the proposed time updating of the PSC, the ASC, and the APES spectral estimators for varying filter lengths as compared to the "brute-force" re-evaluation of the spectral estimates for each time-update using the efficient implementation in [5]. The presented measurements are calculated from the *relative* time taken to execute 100 evaluations, this is to minimize the variance associated with spurious processor loading from the operating system (though this effect is still somewhat evident). As seen from the figure, quite significant gains are achieved even for relatively short filter lengths.

Further, we define the spectral error, $e_\phi(t)$, as the distance between the spectral estimates evaluated over P frequency bins, i.e.,

$$e_\phi(t) = \sum_{k=0}^{P-1} \left| \phi_x(\omega_k) - \hat{\phi}_x^t(\omega_k) \right| \tag{26}$$

where $\phi_x(\omega)$ is the true spectrum, $\hat{\phi}_x^t(\omega)$ the time-updated spectral estimate at time t, and ω_k denotes the kth frequency bin. Fig. 4 shows the spectral error propagation of the ASC spectral estimate[3] using a sliding window covariance updating as compared to the proposed Cholesky updating, clearly showing the robustness of the suggested updating. This robustness is expected, as the Cholesky updating operates on the matrix square root instead of the matrix itself, and is thus preferable [11].

V. Conclusion

In this paper, we have proposed a time updating of the Capon and the APES spectral estimators based on the updating of the inverse Cholesky factor of the forward-backward averaged sample covariance matrix. This updating can be found via the numerically robust time-variant generalized Schur algorithm, providing a natural extension of the current most efficient batch implementation of the estimators. Numerical simulations indicate a significant computational gain over the batch estimation methods for larger filter lengths. Further, studies of the error propagation shows the proposed method to be superior to a sliding window update of the covariance matrix estimate.

References

[1] E. G. Larsson, J. Li, and P. Stoica, "High-resolution nonparametric spectral analysis: Theory and applications," in *High-Resolution and Robust Signal Processing*, Y. Hua, A. B. Gershman, and Q. Cheng, Eds. New York: Marcel-Dekker, 2003.

[2] J. Li and P. Stoica, "Adaptive filtering approach to spectral estimation and SAR imaging," *IEEE Trans. Signal Processing*, vol. 44, no. 6, pp. 1469–1484, Jun. 1996.

[3] J. Capon, "Maximum-likelihood spectral estimation," in *Nonlinear Methods of Spectral Analysis*, 2nd ed, S. Haykin, Ed. New York: Springer-Verlag, 1983, ch. 5.

[4] P. Stoica, A. Jakobsson, and J. Li, "Matched-filterbank interpretation of some spectral estimators," *Signal Processing*, vol. 66, no. 1, pp. 45–59, 1998.

[5] E. G. Larsson and P. Stoica, "Fast implementation of two-dimensional APES and capon spectral estimators," *Multidim. Syst. Signal Process.*, vol. 13, no. 1, pp. 35–54, 2002.

[6] T. Kailath and A. H. Sayed, *Fast Reliable Algorithms for Matrices With Structure*. Philadelphia: SIAM, 1999.

[7] R. Wu, Z.-S. Liu, and J. Li, "Time-varying complex spectral analysis via recursive APES," *Proc. IEE Radar, Sonar, Navigation*, vol. 145, no. 12, pp. 354–360, Dec. 1998.

[8] Z.-S. Liu, R. Wu, and J. Li, "Complex ISAR imaging of maneuvering targets via the capon estimator," *IEEE Trans. Signal Process.*, vol. 47, no. 5, pp. 1262–1271, May 1999.

[9] A. H. Sayed, H. Lev-Ari, and T. Kailath, "Time-variant displacement structure and triangular arrays," *IEEE Trans. Signal Process.*, vol. 42, no. 5, pp. 1052–1062, May 1994.

[10] M. Jansson and P. Stoica, "Forward-only and forward-backward sample covariances—A comparative study," *Signal Process.*, vol. 77, no. 3, pp. 235–245, 1999.

[11] S. Haykin, *Adaptive Filter Theory*, 4th ed. Englewood Cliffs, NJ: Prentice-Hall, 2002.

[3]The PSC and the APES estimators show similar performance.

Stephen R. Alty (M'00) received the B.Eng.(Hons) degree in electronic engineering from the University of Liverpool, Liverpool, U.K., and the Ph.D. degree in signal processing from Liverpool John Moores University, Liverpool, U.K., in 1992 and 1998, respectively.

He was an Assistant Professor at the Liverpool John Moores University. Since 2002, he has been an Assistant Professor within the Centre for Digital Signal Processing Research,King's College London, London, U.K. His research interests include adaptive signal processing and machine learning with application to spectral analysis, speech and biomedical pattern recognition. He is a member of the IEEE Signal Processing and Biomedical Engineering Societies.

Andreas Jakobsson (M'02) received the M.Sc. degree from Lund Institute of Technology, Lund, Sweden, and the Ph.D. degree in signal processing from Uppsala University, Uppsala, Sweden, in 1993, and 2000, respectively.

Since, he has held research and teaching positions with Global IP Sound (**[AU: CITY/country?]**), the Swedish Royal Institute of Technology (**[AU: CITY?]**, Sweden), and King's College London (London, U.K.). During his doctoral studies, he was a Visiting Researcher at Brigham Young University (Provo, UT), Stanford University (Stanford, CA), Katholieke Universiteit Leuven (Leuven, Belgium), and the University of California San Diego. Since January 2004, he has been an Associate Professor in the Department of Electrical Engineering, Karlstad University, Karlstad, Sweden. He also holds a Honorary Research Fellowship at Cardiff University, Cardiff, U.K. His research interests include statistical and array signal processing, detection and estimation theory, and related applications in remote sensing, telecommunication, and biomedicine.

Erik G. Larsson (S'99–M'01) received the Ph.D. degree in electrical engineering from Uppsala University, Uppsala, Sweden, in 2002.

He has held research and teaching positions with Ericsson Radio Systems AB (Stockholm, Sweden), Uppsala University, and the University of Florida (Gainesville, FL). Since August 2003, he has been an Assistant Professor in the Department of Electrical and Computer Engineering, George Washington University, Washington DC. His research interests and experience include space-time diversity for wireless communications, multichannel communication theory, signal processing for communications and radar, and location services for E-911. He has authored and coauthored some 20 papers in IEEE Transactions and Journals and other international journals on these topics. He holds several U.S. patents, and is a coauthor of the textbook *Space-Time Block Coding for Wireless Communications* (Cambridge University Press, 2003, with P. Stoica).

Dr. Larsson is an Associate Editor for the IEEE Transactions on Vehicular Technology.