Time-Recursive IAA Spectral Estimation

G. O. Glentis*, *Member, IEEE*, and A. Jakobsson†, *Senior Member, IEEE*

***Abstract*— This paper presents computationally efficient time-updating algorithms of the recent Iterative Adaptive Approach (IAA) spectral estimation technique. By exploiting the inherently low displacement rank, together with the development of suitable Gohberg-Semencul (GS) representations, and the use of data dependent trigonometric polynomials, the proposed time-recursive IAA algorithm offers a reduction of the necessary computational complexity with at least one order of magnitude. The resulting complexity can also be reduced further by allowing for approximate solutions. Numerical simulations together with theoretical complexity measures illustrate the achieved performance gain.**

***Index Terms*— Adaptive spectral estimation, Iterative Adaptive Approach (IAA), fast algorithms.**

I. INTRODUCTION

ADAPTIVE spectral estimation is of great interest in a wide range of applications, and the topic has received substantial and well-justified interest in the literature during recent decades. Often, when studying time-varying signals, one assumes that the signal of interest can be well approximated as being stationary during a short time interval, allowing the use of techniques such as the popular short-time Fourier transform (STFT). Regrettably, the resulting estimates will suffer from either low time- or frequency resolution as well as high leakage effects [1], [2]. As a result, there has recently been a notable interest in finding alternative time-varying spectral estimation techniques. Due to their excellent performance and inherent robustness to model assumptions, there has been a particular interest in data-dependent filterbank-based algorithms. Notably, both the Capon and APES spectral estimation techniques have recently been extended to allow for time-varying signals [3]–[7]. Generally, these spectral estimates offer higher resolution as compared to the STFT, although this advantage comes at the cost of an increased computational complexity. Recently, an alternative data-dependent approach has been proposed for signals exhibiting a sparse signal representation; in [8], the so-called iterative adaptive approach (IAA) was proposed for passive sensing, channel estimation, and single-antenna radar applications. There, the technique was shown to outperform both the Capon and APES estimators, and, as a result, the technique has since attracted significant interest in a variety of topics [9]–[13]. In these works, the examined signals are assumed to be (reasonably) stationary over the time period of interest, whereas in [14], a sliding window IAA algorithm was proposed for Doppler spectrogram analysis of human gait. As can be seen in these papers, the IAA-based estimation techniques are able to provide highly accurate estimates even when only a few data snapshots are available, although the improved performance comes at a cost of a considerable computational complexity. In this paper, building on our earlier work on computationally efficient time-updating of the Capon and APES spectral estimates [5], [7], as well as our recent work on computationally efficient implementations of the various IAA-based batch spectral estimation techniques [15], we present computationally efficient time-updating techniques for the IAA algorithm. The presented implementations exploiting an efficient implementation of the necessary data dependent trigonometric polynomials as well as the development of suitable GS representations of the estimated inverse covariance matrix (see also [16], [17]). The rest of the paper is organized as follows: in the following section, we review the IAA algorithm and introduce our notation. Then, in Section III, we propose a fast IAA implementation, as well as two approximate implementations. Finally, Section IV, contains our conclusions.

This work was supported in part by the Swedish Research Council and Carl Trygger's foundation.

*G. O. Glentis is with the Department of Science and Technology of Telecommunications, University of Peloponnese, Tripolis, 22100 Greece, email: gglentis@uop.gr.

†A. Jakobsson is with the Dept. of Mathematical Statistics, Lund University, P.O. Box 118, SE-221 00 Lund, Sweden, email: aj@maths.lth.se.

II. PROBLEM FORMULATION

Let $y(n) \in \mathbb{C}$, for $n = 0, 1, \ldots$, represent a data sequence under consideration, whereof subsets of length L may be assumed to be (reasonably) stationary. Define the data and frequency vectors

$$\mathbf{y}_L(n) = [\, y(n) \quad \ldots \quad y(n+L-1)\,]^T, \tag{1}$$
$$\mathbf{f}_L(\omega_k) = [\, 1 \quad e^{\jmath\omega_k} \quad \ldots \quad e^{\jmath\omega_k(L-1)}\,]^T \tag{2}$$

where $(\cdot)^T$ denotes the transpose, and where $\omega_k = 2\pi\frac{k}{K}$, $k = 0, 1, \ldots, K-1$, typically[1] with $K > L$. Let $\Phi_n(\omega_k) = |\alpha_n(\omega_k)|^2$ denote the power of the signal at frequency ω_k at time instant n, where $\alpha_n(\omega_k)$ is the complex-valued spectral amplitude at frequency ω_k. Let $\mathbf{R}_L(n)$ denote an estimate of the $L \times L$ sample covariance matrix at time instant n. Then, a time recursive, sliding window, IAA spectral estimate is formed by iteratively estimating at each time instant $\alpha_n(\omega_k)$ and $\mathbf{R}_L(n)$, until practical convergence, as (see [8], [10], [14] for details)

$$\alpha_n(\omega_k) = \frac{\mathbf{f}_L^H(\omega_k)\mathbf{R}_L^{-1}(n)\mathbf{y}_L(n)}{\mathbf{f}_L^H(\omega_k)\mathbf{R}_L^{-1}(n)\mathbf{f}_L(\omega_k)}, \tag{3}$$
$$\mathbf{R}_L(n) = \sum_{k=0}^{K-1} |\alpha_n(\omega_k)|^2 \mathbf{f}_L(\omega_k)\mathbf{f}_L^H(\omega_k) \tag{4}$$

for $k = 0, 1, \ldots, K-1$, where $(\cdot)^H$ denotes the conjugate transpose, with $\mathbf{R}_L(n)$ initialized to the identity matrix $\mathbf{I}_L$.

[1]For $K = L$, IAA yields the DFT spectrum for all iterations.

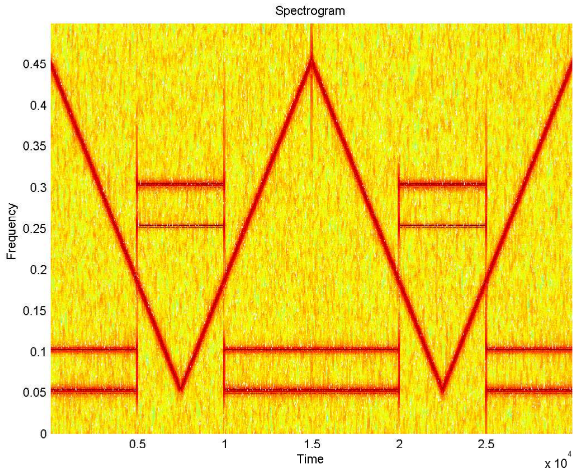

Fig. 1. An example of the spectrogram resulting from the proposed fast, time-recursive, sliding window, TRIAA algorithm.

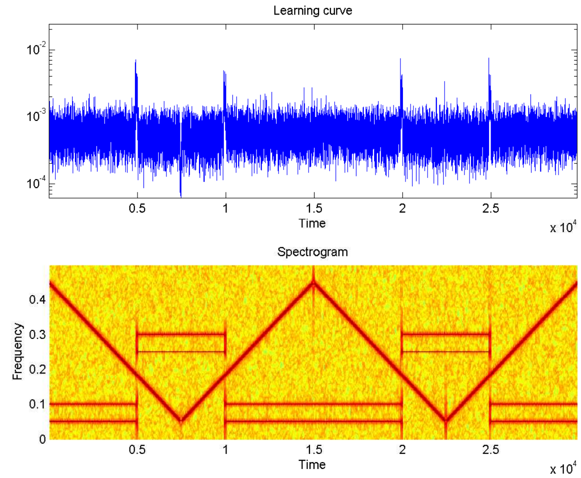

Fig. 2. The learning curve and spectrogram estimate resulting from the approximative TRIAA-a1 estimator.

The brute force implementation of the time recursive IAA method (TRIAA-bf) as described by (3) and (4) results in a computational cost of $m(2KL^2 + KL + L^3)$ operations per processed subset, where m is the number of IAA iterations necessary to allow for convergence; typically, no more than 10 or 15 iterations are needed [8]. However, this figure can be drastically improved by taking into account the special structure of $\mathbf{R}_L(n)$ and the operations required to update $\alpha_n(\omega_k)$ and $\mathbf{R}_L(n)$ each iteration.

III. Fast time-recursive IAA implementation

Define $\mathbf{F}_{LK} \triangleq [\, \mathbf{f}_L(\omega_0) \quad \ldots \quad \mathbf{f}_L(\omega_{K-1}) \,]$ and express the sample covariance matrix estimate in (4) as

$$\mathbf{R}_L(n) = \mathbf{F}_{LK}\, \mathrm{diag}\,\{\, |\alpha_n(\omega_0)|^2 \quad \ldots \quad |\alpha_n(\omega_{K-1})|^2 \,\}\, \mathbf{F}_{LK}^H,$$

where $\mathrm{diag}\{\mathbf{x}\}$ denotes a diagonal matrix formed with the vector $\mathbf{x}$ along the diagonal. As shown in [15], $\mathbf{R}_L(n)$ is a Toeplitz matrix which can be extracted from a circular matrix of higher dimensions, such that

$$\begin{aligned} \mathbf{C}_L(n) &\triangleq \mathbf{W}_K^H \mathrm{diag}\left\{|\alpha_n(\omega_0)|^2, \ldots, |\alpha_n(\omega_{K-1})|^2\right\} \mathbf{W}_K \\ &= \begin{bmatrix} \mathbf{R}_L(n) & \times \\ \times & \times \end{bmatrix}, \end{aligned}$$

where $\mathbf{W}_K$ is the Discrete Fourier Transform (DFT) matrix of size $K \times K$, and the symbol $\times$ denotes unspecified terms of no relevance. Thus, the first column of $\mathbf{R}_L(n)$, denoted by $\mathbf{r}_L(n)$, is obtained by the partition $\mathbf{c}_L(n) = [\mathbf{r}_L^T(n) \;\; \times]^T$, where $\mathbf{c}_L(n)$ is the first column of $\mathbf{C}_L(n)$, and can be computed using the Inverse DFT (IDFT) as $\mathbf{c}_L(n) = \mathbf{W}_K^H \boldsymbol{\alpha}_n$, where $\boldsymbol{\alpha}_n = [\, |\alpha_n(\omega_0)|^2 \quad \ldots \quad |\alpha_n(\omega_{K-1})|^2 \,]^T$. It is worth noting that $\mathbf{r}_L(n)$ can be computed for a subset output points using an incomplete IDFT [18]. Consider the partitioning

$$\begin{aligned} \mathbf{R}_L(n) &= \begin{bmatrix} r_0(n) & \mathbf{r}_{L-1}^{fH}(n) \\ \mathbf{r}_{L-1}^f(n) & \mathbf{R}_{L-1}(n) \end{bmatrix} && (5) \\ &= \begin{bmatrix} \mathbf{R}_{L-1}(n) & \mathbf{J}_{L-1}\mathbf{r}_{L-1}^{f*}(n) \\ \mathbf{r}_{L-1}^{fT}(n)\mathbf{J}_{L-1} & r_0(n) \end{bmatrix}, && (6) \end{aligned}$$

where $\mathbf{J}_{L-1}$ is the exchange matrix. Application of the matrix inversion lemma for partitioned matrices [2] results in

$$\begin{aligned} \mathbf{R}_L^{-1}(n) &= \begin{bmatrix} 0 & \mathbf{0}^T \\ \mathbf{0} & \mathbf{R}_{L-1}^{-1}(n) \end{bmatrix} + \mathbf{A}_L(n)\mathbf{A}_L^H(n) && (7) \\ &= \begin{bmatrix} \mathbf{R}_{L-1}^{-1}(n) & \mathbf{0} \\ \mathbf{0}^T & 0 \end{bmatrix} + \mathbf{J}_L\mathbf{A}_L^*(n)\mathbf{A}_L^T(n)\mathbf{J}_L && (8) \end{aligned}$$

where

$$\mathbf{A}_L(n) = \begin{bmatrix} 1 \\ \mathbf{a}_{L-1}(n) \end{bmatrix} / \sqrt{\alpha_L^f(n)}, \qquad (9)$$

and

$$\begin{aligned} \mathbf{a}_{L-1}(n) &= -\mathbf{R}_{L-1}^{-1}(n)\mathbf{r}_{L-1}^f(n) && (10) \\ \alpha_L^f(n) &= r_0(n) + \mathbf{r}_{L-1}^{fH}(n)\mathbf{a}_{L-1}(n). && (11) \end{aligned}$$

Define the operator

$$\mathbf{Z}_L(\nu) = \begin{bmatrix} \mathbf{0}^T & \nu \\ \mathbf{I}_{L-1} & \mathbf{0} \end{bmatrix} \qquad (12)$$

where $\mathbf{I}_{L-1}$ represents the identity matrix. Clearly, $(\mathbf{Z}_L(\nu))^L = \nu \mathbf{I}_L$. Using (7) and (8) yields

$$\begin{aligned} \mathbf{R}_L^{-1}(n) - \mathbf{Z}_L(\nu_1)\mathbf{R}_L^{-1}(n)\mathbf{Z}_L^T(\nu_2) = \mathbf{A}_L(n)\mathbf{A}_L^H(n) \\ -\mathbf{Z}_L(\nu_1)\mathbf{J}_L\mathbf{A}_L^*(n)\mathbf{A}_L^T(n)\mathbf{J}_L\mathbf{Z}_L^T(\nu_2), \qquad (13) \end{aligned}$$

which, via repeated multiplications by the down-shifting operator, defined in (12), and by the subsequent summation of the resulting equations, leads to an efficient way for the calculation of $\mathbf{R}_L^{-1}(n)$. First, consider the case when $\nu_1 = \nu_2 = 0$. Then, the well-known GS factorization of $\mathbf{R}_L^{-1}(n)$ is obtained [16]

$$\mathbf{R}_L^{-1}(n) = \sum_{i=1}^{2} \sigma_i \mathbf{L}(\mathbf{t}_L^i(n))\mathbf{L}^H(\mathbf{t}_L^i(n)) \qquad (14)$$

where $\mathbf{t}_L^1(n) \triangleq \mathbf{A}_L(n)$ and $\mathbf{t}_L^2(n) \triangleq \mathbf{Z}_L(0)\mathbf{J}_L\mathbf{A}_L^*(n)$, $\sigma_1 = 1$ and $\sigma_2 = -1$, and where $\mathbf{L}(\boldsymbol{\xi}_L)$ denotes a $L \times L$ Toeplitz lower matrix with $\boldsymbol{\xi}_L$ along its first column. The structure in (14) allows for an efficient way to compute the coefficients of the trigonometric polynomial appearing in the denominator

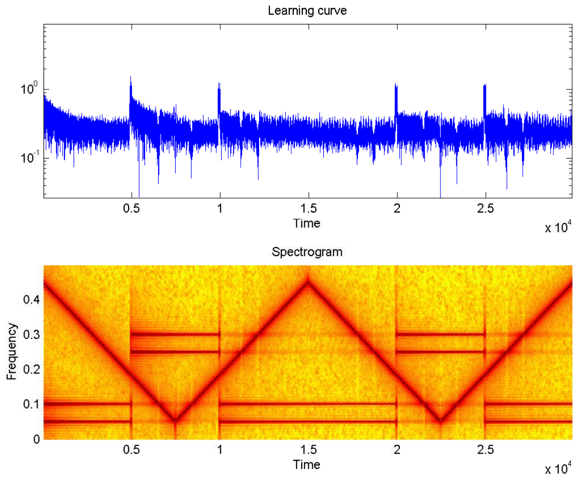

Fig. 3. The learning curve and spectrogram estimate resulting from the approximative TRIAA-a2 estimator using $\mu = 0.001$.

of (3), at a cost of $\mathcal{O}\{L \log_2(L)\}$ operation (see, e.g. [19]). Next, consider $\nu_1 = 0$ and $\nu_2 = 1$. Then, a variant of the GS factorization is obtained as [20]

$$\mathbf{R}_L^{-1}(n) = \sum_{i=1}^{2} \sigma_i \mathbf{L}(\mathbf{t}_L^i(n)) \mathbf{C}^H(\bar{\mathbf{t}}_L^i(n)) \tag{15}$$

where $\bar{\mathbf{t}}_L^1(n) \triangleq \mathbf{t}_L^1(n)$ and $\bar{\mathbf{t}}_L^2(n) \triangleq \mathbf{Z}_L(1)\mathbf{J}_L\mathbf{A}_L^*(n)$, with $\mathbf{C}(\boldsymbol{\xi}_L)$ denoting a $L \times L$ circulant matrix having $\boldsymbol{\xi}_L$ along its first column. This particular GS decomposition of $\mathbf{R}_L^{-1}(n)$ allows for an improved implementation of the matrix-vector product in the numerator of (3), defined as

$$\mathbf{d}_L(n) \triangleq \mathbf{R}_L^{-1}(n)\mathbf{y}_L(n). \tag{16}$$

Finally, the linear system of equations given by (10) can be solved using the Levinson-Durbin algorithm (see, e.g., [2]). The resulting algorithm, termed the fast time-recursive IAA (TRIAA), is tabulated in Table I. Interestingly, one may reduce the complexity even further by allowing for an approximate solution. Noting that, upon convergence, $\mathbf{R}_L^{\ell}(n) \approx \mathbf{R}_L^{\ell-1}(n)$ and $\mathbf{R}_L(n) \approx \mathbf{R}_L(n-1)$ suggest that an approximate solution may be found by setting $m = 1$ and using $\mathbf{R}_L^0(n) = \mathbf{R}_L^1(n-1)$. As shown in Figures 1 and 2, the resulting approximation does not induce any substantial loss in performance. We term the resulting approximative estimate the TRIAA-a1 estimator. Alternatively, one may instead use a single step steepest descent scheme for the computation of $\mathbf{a}_{L-1}(n)$. Steepest descent algorithms have been proved to be a well behaved and powerful tool for the iterative solution of linear (and non-linear) systems of equations, and has successfully been applied to many batch and adaptive signal processing applications (see, e.g., [21]–[23]). This version, here termed the TRIAA-a2 estimator, is detailed in Table II, with $\mu > 0$ denoting a trimming parameter that control the convergence rate of the algorithm. To speed up convergence, the algorithm of Table I is used for initialization. The complexity of the resulting estimator is now reduced even further, although, as shown in Figure 3, this computational reduction comes at the price of a somewhat worse spectral estimate, where it is clear that the TRIAA-a2 estimator will yield estimates with a somewhat wider mainlobe as compared to the TRIAA-a1 estimator. Here, the results in Figures 1-3 are obtained from the data set described in [7]. The convergence rates of the TRIAA-a1 and TRIAA-a2 estimators are also illustrated in Figures 2 and 3, using a learning curve, defined as

$$\mathcal{C}_n = \frac{1}{K} \sum_{k=0}^{K-1} \left| \Phi_n(\omega_k) - \Phi_{n-1}(\omega_k) \right| .$$

TABLE I

TIME RECURSIVE IAA ALGORITHM (TRIAA)

Initialization:

$$\begin{aligned} \mathbf{y}_L(n) &= [\, y(n) \quad y(n+1) \quad \ldots \quad y(n_L-1) \,] \\ \mathbf{R}_L^0(n) &= \mathbf{I}_L \end{aligned}$$

Then, for iteration $\ell = 1, \ldots, m$,

$$\begin{aligned} \mathbf{a}_{L-1}(n) &= -\left[\mathbf{R}_{L-1}^{\ell-1}(n)\right]^{-1} \mathbf{r}_{L-1}^{f(\ell-1)}(n) \\ \left[\mathbf{R}_L^{\ell}(n)\right]^{-1} &= \sum_{i=1}^{2} \sigma_i \mathbf{L}(\mathbf{t}_L^i(n)) \mathbf{C}^H(\bar{\mathbf{t}}_L^i(n)) \\ \mathbf{d}_L(n) &= [\mathbf{R}_L^{\ell}(n)]^{-1}\mathbf{y}_L(n) \\ \psi_n(\omega_k) &= \mathbf{f}_L^H(\omega_k)\mathbf{d}_L(n) \\ \varphi_n(\omega_k) &= \sum_{i=-L+1}^{L-1} \varphi_i(n) e^{\jmath \frac{2\pi k}{K} i} \\ \alpha_n(\omega_k) &= \frac{\psi_n(\omega_k)}{\varphi_n(\omega_k)} \\ \begin{bmatrix} \mathbf{r}_L^{\ell}(n) \\ \times \end{bmatrix} &= \mathbf{W}_K^H \left[\, |\alpha_n(\omega_0)|^2 \quad \ldots \quad |\alpha_n(\omega_{K-1})|^2 \,\right]^T \end{aligned}$$

We proceed to examine the computational cost of the discussed time-recursive estimators. Consider first the computation of the numerator of (3), which, using zero padding, can be expressed as $\psi_n(\omega_k) \triangleq \mathbf{f}_K^H(\omega_k) \left[\, \mathbf{d}_L^T(n) \quad \mathbf{0}_{K-L}^T \,\right]^T$ which can be evaluated using an incomplete input FFT, where L out of K samples are assumed to be nonzero, at a computational cost of $\phi(L, K) \approx \frac{K}{L}\phi(L)$ [18], where $\phi(L, K)$ denotes the cost of performing a FFT (or an Inverse FFT, IFFT) of length L, being zeropadded to length K, and where $\phi(K, K) \equiv \phi(K)$.

Moreover, the denominator of (3) can be expressed as $\varphi_n(\omega_k) = e^{-\jmath \frac{2\pi k}{K}(L-1)} \sum_{i=0}^{2L-1} \varphi_i(n) e^{\jmath \frac{2\pi k}{K} i}$ which can be evaluated at a cost of $\phi(2L, K) + K$ operations. Furthermore, $\mathbf{r}_L(n)$ can be computed using an incomplete output FFT, where L out of K samples are produced at a cost of $\phi(L, K)$ operations [18], and the solution of computing (10) can be done in about $(L-1)^2$ operations. Computing $\mathbf{d}_L(n)$ using (15) and (16) can be done in no more than $9\phi(L)$ operations [20], and the coefficients $\varphi_i(n)$ of the trigonometric polynomial $\varphi_n(\omega_k)$ that appears in the denominator of (4) are computed at a cost of $10\phi(L)$ using the method presented in [19]. However, this figure can be reduced to $6\phi(L)$ since some of the computations have already be performed at a previous step of the algorithm. Finally, the Toeplitz-vector products required for the computation of variables $\mathbf{e}_{L-1}(n)$ and $\beta(n)$ that appear in Table II result in an additional cost of $8\phi(L)$. In

TABLE II

TIME RECURSIVE APPROXIMATE IAA ALGORITHM (TRIAA$-a2$)

Initialization:

$$\mathbf{y}_L(n) = [\,y(n) \quad y(n+1) \quad \ldots \quad y(n+L-1)\,]$$

Then, if n=0, follow Table I for a given m, otherwise

$$\begin{aligned}
\mathbf{e}_{L-1}(n) &= -\mathbf{r}^f_{L-1}(n) - \mathbf{R}_{L-1}(n)\mathbf{a}_{L-1}(n-1)\\
\beta(n) &= \mathbf{e}^H_{L-1}(n)\mathbf{R}_{L-1}(n)\mathbf{e}_{L-1}(n)\\
\rho(n) &= \mathbf{e}^H_{L-1}(n)\mathbf{e}_{L-1}(n)\\
\alpha(n) &= \frac{\rho(n)}{\beta(n)}\\
\mathbf{a}_{L-1}(n) &= \mathbf{a}_{L-1}(n-1) + \mu\alpha(n)\mathbf{e}_{L-1}(n)\\
\mathbf{R}^{-1}_L(n) &= \sum_{i=1}^{2}\sigma_i\mathbf{L}(\mathbf{t}^i_L(n))\mathbf{C}^H(\bar{\mathbf{t}}^i_L(n))\\
\mathbf{d}_L(n) &= \mathbf{R}^{-1}_L(n)\mathbf{y}_L(n)\\
\psi_n(\omega_k) &= \mathbf{f}^H_L(\omega_k)\mathbf{d}_L(n)\\
\varphi_n(\omega_k) &= \sum_{i=-L+1}^{L-1}\varphi_i(n)e^{\jmath\frac{2\pi k}{K}i}\\
\alpha_n(\omega_k) &= \frac{\psi_n(\omega_k)}{\varphi_n(\omega_k)}\\
\begin{bmatrix}\mathbf{r}_L(n+1)\\ \times\end{bmatrix} &= \mathbf{W}^H_K\,[\,|\alpha_n(\omega_0)|^2 \quad \ldots \quad |\alpha_n(\omega_{K-1})|^2\,]^T
\end{aligned}$$

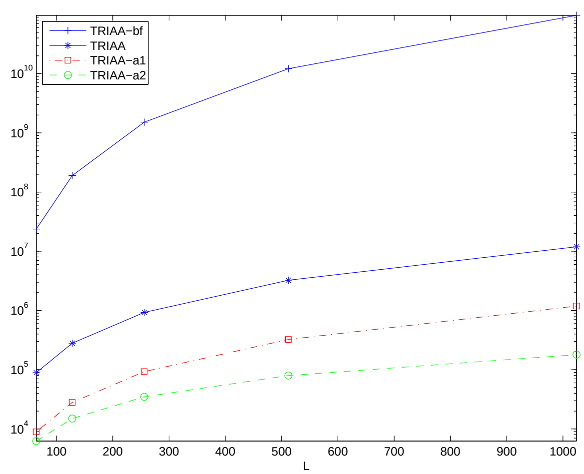

Fig. 4. Approximative computational complexity of the discussed estimators.

conclusion, the computational cost of the fast time-recursive algorithms presented in this paper are:

$$\begin{aligned}
C^{TRIAA} &= m[(L-1)^2 + 15\phi(L) + 4\phi(L,K)],\\
C^{TRIAA-a1} &= (L-1)^2 + 15\phi(L) + 4\phi(L,K),\\
C^{TRIAA-a2} &= 23\phi(L) + 4\phi(L,K).
\end{aligned}$$

The computational complexity of the discussed methods is illustrated in Figure 4, for $K = 4L$.

IV. CONCLUSIONS

In this paper, we have presented three computationally efficient implementations of the IAA spectral estimation technique. The techniques exploit the rich structure of the estimator to substantially reduce the necessary complexity, as well as form approximate solutions of notably lower complexity, while only suffering minor degradation of the resulting estimates.

REFERENCES

[1] L. Cohen, *Time-Frequency Analysis*, Prentice Hall, Englewood Cliffs, N.J., 1994.

[2] P. Stoica and R. Moses, *Spectral Analysis of Signals*, Prentice Hall, Upper Saddle River, N.J., 2005.

[3] R. Wu, Z.-S. Liu, and J. Li, "Time-varying complex spectral analysis via recursive APES," *IEE Proc. Radar, Sonar and Navigation*, vol. 145, pp. 354–360, December 1998.

[4] M. T. Özgen, "Extension of the capons spectral estimator to timefrequency analysis and to the analysis of polynomial-phase signals," *Signal Processing*, vol. 83, pp. 575–592, 2003.

[5] S. R. Alty, A. Jakobsson, and E. G. Larsson, "Efficient Time-Recursive Implementation of Matched Filterbank Spectral Estimators," *IEEE Trans. on Circuits and Systems I: Regular Papers*, vol. 52, no. 3, pp. 516–521, March 2005.

[6] J. Benesty, J. Chen, and Y. Huang, "Recursive and Fast Recursive Capon Spectral Estimators," *EURASIP Journal on Advances in Signal Processing*, vol. 2007, pp. Article ID 45194, 12 pages, 2007.

[7] G. O. Glentis, "Efficient Algorithms for Adaptive Capon and APES Spectral Estimation," *IEEE Trans. Signal Processing*, vol. 58, no. 1, pp. 84–96, Jan. 2010.

[8] T. Yardibi, J. Li, P. Stoica, M. Xue, and A. B. Baggeroer, "Source Localization and Sensing: A Nonparametric Iterative Approach Based on Weighted Least Squares," *IEEE Trans. on Aerospace and Electronic Systems*, vol. 46, no. 1, pp. 425–443, January 2010.

[9] W. Roberts, P. Stoica, J. Li, T. Yardibi, and F. A. Sadjadi, "Iterative Adaptive Approaches to MIMO Radar Imaging," *IEEE Journal of Selected Topics in Signal Processing*, vol. 4, no. 1, pp. 5–20, Feb 2010.

[10] P. Stoica, Jian Li, and Jun Ling, "Missing Data Recovery Via a Nonparametric Iterative Adaptive Approach," *IEEE Signal Processing Letters*, vol. 16, no. 4, pp. 241–244, April 2009.

[11] P. Stoica, Jian Li, and Hao He, "Spectral Analysis of Nonuniformly Sampled Data: A New Approach Versus the Periodogram," *IEEE Trans. on Signal Processing*, vol. 57, no. 3, pp. 843–858, March 2009.

[12] N. R. Butt and A. Jakobsson, "Coherence spectrum estimation from nonuniformly sampled sequences," *IEEE Signal Processing Letters*, vol. 17, no. 4, pp. 339–342, April 2010.

[13] E. Gudmundson, P. Stoica, Jian Li, A. Jakobsson, M. D. Rowe, J. A. S. Smith, and Jun Ling, "Spectral Estimation of Irregularly Sampled Exponentially Decaying Signals with Applications to RF Spectroscopy," *Journal of Magnetic Resonance*, vol. 203, no. 1, pp. 167–176, March 2010.

[14] L. Du, J. Li, P. Stoica, H. Ling, and S. S. Ram, "Doppler spectrogram analysis of human gait via iterative adaptive approach," *Electronics Letters*, vol. 45, no. 3, pp. 186–188, 2009.

[15] G.-O. Glentis and A. Jakobsson, "Efficient implementation of iterative adaptive approach spectral estimation techniques," *Submitted to IEEE Trans. Signal Processing*.

[16] I. C. Gohberg and A. A. Semencul, "On the Inversion of Finite Toeplitz Matrices and their Continuous Analogs," *Mat. Issled.*, vol. 2, pp. 201–233, 1972, (in Russian).

[17] T. Kailath and A. H. Sayed, "Displacement Structure: Theory and Applications," *SIAM Review*, vol. 37, no. 3, pp. 297–386, September 1995.

[18] H. V. Sorensen and C. S. Burrus, "Efficient Computation of the DFT with Only a Subset of Input or Output Points," *IEEE Trans. Signal Processing*, vol. 41, no. 3, pp. 1184–1200, March 1993.

[19] G.-O. Glentis, "A Fast Algorithm for APES and Capon Spectral Estimation," *IEEE Transactions on Signal Processing*, vol. 56, no. 9, pp. 4207 –4220, sept. 2008.

[20] G. Ammar and P. Gader, "A Variant of the Gohberg–Semencul Formula Involving Circulant Matrices," *SIAM Journal on Matrix Analysis and Applications*, vol. 12, no. 3, pp. 534–540, July 1991.

[21] D. G. Luenberger, *Optimization by Vector Space Methods*, John Wiley and Sons, New York, 1969.

[22] L. Ljung and T. Söderström, *Theory and Practice of Recursive Identification*, The MIT Press, Cambridge, M.A., 1983.

[23] G. O. Glentis, K. Berberidis, and S. Theodoridis, "Efficient Least Squares Adaptive Algorithms for FIR Transversal Filtering," *IEEE Signal Processing Magazine*, vol. 16, no. 4, pp. 13–41, July 1999.