Design of Third-Order Tensorial RLS Adaptive Filtering Algorithms

Ionuţ-Dorinel Fîciu
Telecommunications Department
POLITEHNICA Bucharest
Bucharest, Romania
email: ionut.ficiu22@gmail.com

Camelia Elisei-Iliescu
CNS
ROMATSA
Bucharest, Romania
email: camelia.elisei@romatsa.ro

Laura-Maria Dogariu
Telecommunications Department
POLITEHNICA Bucharest
Bucharest, Romania
email: ldogariu@comm.pub.ro

Constantin Paleologu
Telecommunications Department
POLITEHNICA Bucharest
Bucharest, Romania
email: pale@comm.pub.ro

Cristian-Lucian Stanciu
Telecommunications Department
POLITEHNICA Bucharest
Bucharest, Romania
email: cristian@comm.pub.ro

Cristian Anghel
Telecommunications Department
POLITEHNICA Bucharest
Bucharest, Romania
email: canghel@comm.pub.ro

***Abstract*—A recently developed Recursive Least-Squares (RLS) adaptive filter based on a Third-Order Tensor (TOT) decomposition technique, namely RLS-TOT, has proved to be efficient in system identification problems that target the estimation of long length impulse responses. This solution fits very well in echo cancellation scenarios, where the associated impulse response of the echo path can reach hundreds or even thousands of coefficients. In this short paper, we further discuss several strategies for improving the performance of RLS-TOT, focusing on its main parameters that control the convergence features.**

***Index Terms*—adaptive filter; recursive least-squares algorithm; echo cancellation; tensor decomposition; convergence parameters**

I. Introduction

Adaptive filtering algorithms are frequently involved in many real-word system identification problems [1]. Among them, the Recursive Least-Squares (RLS) algorithm represents a very appealing choice due to its fast convergence rate, which can be achieved even for highly correlated input signals [2]. However, the price to pay is a high computational complexity.

In this framework, the overall difficulty increases when dealing with the identification of long length impulse responses, which raise significant challenges in terms of the complexity, convergence/tracking, and accuracy of the solution. Even the "fast" (i.e., less complex) versions of the RLS algorithm face performance limitations in such scenarios [3]. A well-known example is related to echo cancellation, where the echo paths are usually modeled as finite impulse response filters that can reach hundreds/thousands of coefficients [4].

Exploiting the characteristics of the systems to be identified represents a natural path to follow, in order to overcome the main challenges related to the estimation of long length impulse responses. In this context, several recent works have focused on decomposition-based techniques that involve the Nearest Kronecker Product (NKP) and low-rank approximations [5]–[8]. In this framework, the NKP-based approach in conjunction with a Third-Order Tensor (TOT) decomposition has been addressed in [9] and [10]. The resulting RLS-TOT algorithm combines the coefficients of three adaptive filters of much shorter lengths, which leads to important advantages in terms of the main performance criteria.

In this short paper, we explore several upcoming developments related to the RLS-TOT adaptive filtering algorithm, aiming to improve the overall performance by tuning its convergence features. In this context, we target the development of improved versions of this algorithm, by using variable forgetting factors and variable regularization parameters.

In the following, Section II presents the RLS-TOT algorithm. Next, Section III is dedicated to the results and discussions. Finally, Section IV concludes this paper.

II. RLS-TOT Algorithm

Let us consider a single-input single-output scenario with real-valued signals. The main goal is to identify an unknown impulse response with L real-valued coefficients, which are grouped into the column vector $\mathbf{h}$. Thus, at the discrete-time index n, the reference signal results in

$$d(n) = \mathbf{h}^T \mathbf{x}(n) + w(n) = y(n) + w(n), \tag{1}$$

where the superscript ^{T} denotes the transpose operator, the column vector $\mathbf{x}(n)$ contains the most recent L samples of the zero-mean input signal $x(n)$, $y(n) = \mathbf{h}^T \mathbf{x}(n)$ is the output signal, and $w(n)$ is a zero-mean additive noise, which is uncorrelated with $x(n)$.

Also, let us consider that the length of $\mathbf{h}$ can be expressed as $L = L_{11} L_{12} L_2$, with $L_{11} \geq L_{12}$ and $L_{11} L_{12} \gg L_2$. Thus, the impulse response can be equivalently decomposed as [9]

$$\mathbf{h} = \sum_{l=1}^{L_2} \sum_{p=1}^{P} \mathbf{h}_2^l \otimes \mathbf{h}_{12}^{lp} \otimes \mathbf{h}_{11}^{lp}, \tag{2}$$

where $P < L_{12}$ and the (shorter) component impulse responses $\mathbf{h}_{11}^{lp}$, $\mathbf{h}_{12}^{lp}$, and $\mathbf{h}_2^l$ have the lengths L_{11}, L_{12}, and

Copyright (c) IARIA, 2024. ISBN: 978-1-68558-142-8

L_2, respectively, while $\otimes$ denotes the Kronecker product. The particular subscript/superscript notation indicates different set of vectors related to the component impulse responses, as shown in Table I (which summarizes the specific data structures). The development from [9] that led to (2) is based on the low-rank approach [5]. This is a reasonable approach, since in practice most of the system impulse responses are low rank, especially in the context of room acoustics [7].

In this decomposition framework, the main focus is the identification of the component impulse responses from (2). Thus, the identification of the impulse response $\mathbf{h}$ (with $L = L_{11}L_{12}L_2$ coefficients) is reformulated as a combination of three sets of coefficients, i.e., $\mathbf{h}_2^l$ of length L_2 (with $l = 1, 2, \ldots, L_2$), $\mathbf{h}_{12}^{lp}$ of length L_{12}, and $\mathbf{h}_{11}^{lp}$ of length L_{11} (with $l = 1, 2, \ldots, L_2$ and $p = 1, 2, \ldots, P$). Correspondingly, related to these three sets, we need to estimate L_2^2, $PL_{12}L_2$, and $PL_{11}L_2$ coefficients, respectively. For the common decomposition setup that involves $L_{11}L_{12} \gg L_2$ and $P \ll L_{12}$ [9], this represents an important dimensionality reduction, especially for large values of L.

In this context, let us consider that $\widehat{\mathbf{h}}(n)$ is an estimate of the impulse response $\mathbf{h}$ at the discrete-time index n, so that the a priori error signal results in

$$e(n) = d(n) - \widehat{\mathbf{h}}^T(n-1)\mathbf{x}(n). \tag{3}$$

On the other hand, using the impulse response decomposition from (2), we can also express its estimate as $\widehat{\mathbf{h}}(n) = \sum_{l=1}^{L_2}\sum_{p=1}^{P} \widehat{\mathbf{h}}_2^l(n) \otimes \widehat{\mathbf{h}}_{12}^{lp}(n) \otimes \widehat{\mathbf{h}}_{11}^{lp}(n)$, where $\widehat{\mathbf{h}}_{11}^{lp}(n)$, $\widehat{\mathbf{h}}_{12}^{lp}(n)$, and $\widehat{\mathbf{h}}_2^l(n)$ are three shorter impulse responses of length L_{11}, L_{12}, and L_2, respectively. Thus, in order to construct the cost functions of the RLS-TOT algorithm, we can rewrite $e(n)$ in three equivalent ways, with the purpose of "extracting" each individual component. As a result, following the development from [10], the equations that defined the RLS-TOT are:

$$\mathbf{k}_{12,11}(n) = \frac{\mathbf{P}_{12,11}(n-1)\mathbf{x}_{12,11}(n)}{\lambda_2 + \mathbf{x}_{12,11}^T(n)\mathbf{P}_{12,11}(n-1)\mathbf{x}_{12,11}(n)},$$

$$\mathbf{k}_{2,11}(n) = \frac{\mathbf{P}_{2,11}(n-1)\mathbf{x}_{2,11}(n)}{\lambda_{12} + \mathbf{x}_{2,11}^T(n)\mathbf{P}_{2,11}(n-1)\mathbf{x}_{2,11}(n)},$$

$$\mathbf{k}_{2,12}(n) = \frac{\mathbf{P}_{2,12}(n-1)\mathbf{x}_{2,12}(n)}{\lambda_{11} + \mathbf{x}_{2,12}^T(n)\mathbf{P}_{2,12}(n-1)\mathbf{x}_{2,12}(n)},$$

$$\mathbf{P}_{12,11}(n) = \lambda_2^{-1}\left[\mathbf{I}_{L_2^2} - \mathbf{k}_{12,11}(n)\mathbf{x}_{12,11}^T(n)\right]\mathbf{P}_{12,11}(n-1),$$

$$\mathbf{P}_{2,11}(n) = \lambda_{12}^{-1}\left[\mathbf{I}_{PL_{12}L_2} - \mathbf{k}_{2,11}(n)\mathbf{x}_{2,11}^T(n)\right]\mathbf{P}_{2,11}(n-1),$$

$$\mathbf{P}_{2,12}(n) = \lambda_{11}^{-1}\left[\mathbf{I}_{PL_{11}L_2} - \mathbf{k}_{2,12}(n)\mathbf{x}_{2,12}^T(n)\right]\mathbf{P}_{2,12}(n-1),$$

$$\underline{\widehat{\mathbf{h}}}_2(n) = \underline{\widehat{\mathbf{h}}}_2(n-1) + \mathbf{k}_{12,11}(n)e(n),$$

$$\underline{\widehat{\mathbf{h}}}_{12}(n) = \underline{\widehat{\mathbf{h}}}_{12}(n-1) + \mathbf{k}_{2,11}(n)e(n),$$

$$\underline{\widehat{\mathbf{h}}}_{11}(n) = \underline{\widehat{\mathbf{h}}}_{11}(n-1) + \mathbf{k}_{2,12}(n)e(n),$$

where λ_2, λ_{12}, and λ_{11} are the forgetting factors, which are positive constants smaller than or equal to one. The rest of notation is detailed in Table I, where $\mathbf{I}_{L_\bullet}$ denotes the identity matrix of size $L_\bullet \times L_\bullet$ and $\mathbf{0}_{L_\bullet}$ generally denotes an all-zeros column vector of length $L_\bullet$. As shown in Table I, the three

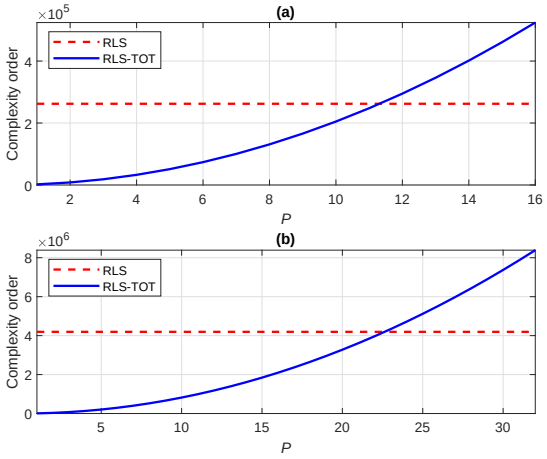

Figure 1. Complexity order of the conventional RLS algorithm and RLS-TOT for two impulse responses, with lengths (a) $L = 512$ and (b) $L = 2048$.

component impulse responses are connected via the Kronecker product, so that they are interdependent. The initialization setup is $\mathbf{P}_{12,11}(0) = \delta^{-1}\mathbf{I}_{L_2^2}$, $\mathbf{P}_{2,11}(0) = \delta^{-1}\mathbf{I}_{PL_{12}L_2}$, and $\mathbf{P}_{2,12}(0) = \delta^{-1}\mathbf{I}_{PL_{11}L_2}$, where $\delta > 0$ is the regularization parameter. Also, the components of the filters are initialized with $\widehat{\mathbf{h}}_2^l(0) = \begin{bmatrix} 1 & \mathbf{0}_{L_2-1}^T \end{bmatrix}^T$, $\widehat{\mathbf{h}}_{12}^{lp}(0) = \begin{bmatrix} 1 & \mathbf{0}_{L_{12}-1}^T \end{bmatrix}^T$, and $\widehat{\mathbf{h}}_{11}^{lp}(0) = \begin{bmatrix} 1 & \mathbf{0}_{L_{11}-1}^T \end{bmatrix}^T$.

The computational complexity of the RLS-based algorithms is proportional to the square of the filter length [2]. Thus, the conventional RLS algorithm requires a computational amount proportional to $\mathcal{O}(L^2) = \mathcal{O}(L_{11}L_{12}L_2)^2$. On the other hand, the RLS-TOT combines three much shorter filters (for the common setup of the decomposition parameters), so that it has a lower computational complexity order, i.e., $\mathcal{O}\left[L_2^4 + (PL_{11}L_2)^2 + (PL_{12}L_2)^2\right]$. This aspect is indicated in Figure 1, where the complexity orders of the conventional RLS algorithm and RLS-TOT are plotted, considering two impulse responses with $L = 512$ and $L = 2048$. For $L = 512$, the decomposition setup of the RLS-TOT involves $L_2 = 2$ and $L_{11} = L_{12} = 16$, while for $L = 2048$ we use $L_2 = 2$ and $L_{11} = L_{12} = 32$. Note that the value of P is usually significantly smaller than L_{12}, while $L_{12} \ll L$. Consequently, the computational complexity order of the RLS-TOT can be much lower as compared to the conventional RLS algorithm.

In the analysis reported in [10], the RLS-TOT was compared to the conventional RLS algorithm, but also with a previously developed decomposition-based version that exploits a second-order decomposition level, which is referred as the RLS algorithm using the NKP decomposition, i.e., RLS-NKP [6]. This counterpart combines the estimates provided by two adaptive filters of lengths $P^*L_1^*$ and $P^*L_2^*$, with $L = L_1^*L_2^*$ and $P^* < L_2^*$. While the RLS-TOT is able to outperform the conventional RLS benchmark and also the RLS-NKP, there is still room for improvements, as outlined in the next section.

Copyright (c) IARIA, 2024. ISBN: 978-1-68558-142-8

TABLE I
NOTATION USED FOR THE RLS-TOT ALGORITHM.

Indices: $l = 1, 2, \ldots, L_2,\ p = 1, 2, \ldots, P$

Initialization:

$$\underline{\widehat{\mathbf{h}}}_2(0) = \left[\begin{array}{ccc} \left(\widehat{\mathbf{h}}_2^1\right)^T(0) & \cdots & \left(\widehat{\mathbf{h}}_2^{L_2}\right)^T(0) \end{array} \right]^T$$

$$\underline{\widehat{\mathbf{h}}}_{12}^{l}(0) = \left[\begin{array}{ccc} \left(\widehat{\mathbf{h}}_{12}^{l1}\right)^T(0) & \cdots & \left(\widehat{\mathbf{h}}_{12}^{lP}\right)^T(0) \end{array} \right]^T$$

$$\underline{\widehat{\mathbf{h}}}_{12}(0) = \left[\begin{array}{ccc} \left(\widehat{\underline{\mathbf{h}}}_{12}^{1}\right)^T(0) & \cdots & \left(\widehat{\underline{\mathbf{h}}}_{12}^{L_2}\right)^T(0) \end{array} \right]^T$$

$$\underline{\widehat{\mathbf{h}}}_{11}^{l}(0) = \left[\begin{array}{ccc} \left(\widehat{\mathbf{h}}_{11}^{l1}\right)^T(0) & \cdots & \left(\widehat{\mathbf{h}}_{11}^{lP}\right)^T(0) \end{array} \right]^T$$

$$\underline{\widehat{\mathbf{h}}}_{11}(0) = \left[\begin{array}{ccc} \left(\widehat{\underline{\mathbf{h}}}_{11}^{1}\right)^T(0) & \cdots & \left(\widehat{\underline{\mathbf{h}}}_{11}^{L_2}\right)^T(0) \end{array} \right]^T$$

For discrete-time index $n = 1, 2, \ldots$

$$\widehat{\mathbf{H}}_{12,11}^{lp}(n-1) = \mathbf{I}_{L_2} \otimes \widehat{\mathbf{h}}_{12}^{lp}(n-1) \otimes \widehat{\mathbf{h}}_{11}^{lp}(n-1)$$

$$\underline{\widehat{\mathbf{H}}}_{12,11}^{l}(n-1) = \sum_{p=1}^{P} \widehat{\mathbf{H}}_{12,11}^{lp}(n-1)$$

$$\underline{\widehat{\mathbf{H}}}_{12,11}(n-1) = \left[\begin{array}{ccc} \underline{\widehat{\mathbf{H}}}_{12,11}^{1}(n-1) & \cdots & \underline{\widehat{\mathbf{H}}}_{12,11}^{L_2}(n-1) \end{array} \right]$$

$$\widehat{\mathbf{H}}_{2,11}^{lp}(n-1) = \widehat{\mathbf{h}}_{2}^{l}(n-1) \otimes \mathbf{I}_{L_{12}} \otimes \widehat{\mathbf{h}}_{11}^{lp}(n-1)$$

$$\underline{\widehat{\mathbf{H}}}_{2,11}^{l}(n-1) = \left[\begin{array}{ccc} \widehat{\mathbf{H}}_{2,11}^{l1}(n-1) & \cdots & \widehat{\mathbf{H}}_{2,11}^{lP}(n-1) \end{array} \right]$$

$$\underline{\widehat{\mathbf{H}}}_{2,11}(n-1) = \left[\begin{array}{ccc} \underline{\widehat{\mathbf{H}}}_{2,11}^{1}(n-1) & \cdots & \underline{\widehat{\mathbf{H}}}_{2,11}^{L_2}(n-1) \end{array} \right]$$

$$\widehat{\mathbf{H}}_{2,12}^{lp}(n-1) = \widehat{\mathbf{h}}_{2}^{l}(n-1) \otimes \widehat{\mathbf{h}}_{12}^{lp}(n-1) \otimes \mathbf{I}_{L_{11}}$$

$$\underline{\widehat{\mathbf{H}}}_{2,12}^{l}(n-1) = \left[\begin{array}{ccc} \widehat{\mathbf{H}}_{2,12}^{l1}(n-1) & \cdots & \widehat{\mathbf{H}}_{2,12}^{lP}(n-1) \end{array} \right]$$

$$\underline{\widehat{\mathbf{H}}}_{2,12}(n-1) = \left[\begin{array}{ccc} \underline{\widehat{\mathbf{H}}}_{2,12}^{1}(n-1) & \cdots & \underline{\widehat{\mathbf{H}}}_{2,12}^{L_2}(n-1) \end{array} \right]$$

$$\mathbf{x}_{12,11}(n) = \underline{\widehat{\mathbf{H}}}_{12,11}^{T}(n-1)\mathbf{x}(n)$$

$$\mathbf{x}_{2,11}(n) = \underline{\widehat{\mathbf{H}}}_{2,11}^{T}(n-1)\mathbf{x}(n)$$

$$\mathbf{x}_{2,12}(n) = \underline{\widehat{\mathbf{H}}}_{2,12}^{T}(n-1)\mathbf{x}(n)$$

$$\underline{\widehat{\mathbf{h}}}_2(n) = \left[\begin{array}{ccc} \left(\widehat{\mathbf{h}}_2^1\right)^T(n) & \cdots & \left(\widehat{\mathbf{h}}_2^{L_2}\right)^T(n) \end{array} \right]^T$$

$$\underline{\widehat{\mathbf{h}}}_{12}(n) = \left[\begin{array}{ccc} \left(\underline{\widehat{\mathbf{h}}}_{12}^{1}\right)^T(n) & \cdots & \left(\underline{\widehat{\mathbf{h}}}_{12}^{L_2}\right)^T(n) \end{array} \right]^T$$

$$\underline{\widehat{\mathbf{h}}}_{12}^{l}(n) = \left[\begin{array}{ccc} \left(\widehat{\mathbf{h}}_{12}^{l1}\right)^T(n) & \cdots & \left(\widehat{\mathbf{h}}_{12}^{lP}\right)^T(n) \end{array} \right]^T$$

$$\underline{\widehat{\mathbf{h}}}_{11}(n) = \left[\begin{array}{ccc} \left(\underline{\widehat{\mathbf{h}}}_{11}^{1}\right)^T(n) & \cdots & \left(\underline{\widehat{\mathbf{h}}}_{11}^{L_2}\right)^T(n) \end{array} \right]^T$$

$$\underline{\widehat{\mathbf{h}}}_{11}^{l}(n) = \left[\begin{array}{ccc} \left(\widehat{\mathbf{h}}_{11}^{l1}\right)^T(n) & \cdots & \left(\widehat{\mathbf{h}}_{11}^{lP}\right)^T(n) \end{array} \right]^T$$

$$\widehat{\mathbf{h}}(n) = \sum_{l=1}^{L_2} \sum_{p=1}^{P} \widehat{\mathbf{h}}_2^l(n) \otimes \widehat{\mathbf{h}}_{12}^{lp}(n) \otimes \widehat{\mathbf{h}}_{11}^{lp}(n)$$

III. RESULTS AND DISCUSSIONS

The main parameters of the RLS-based algorithms are the forgetting factors. While the conventional RLS algorithm involves a single forgetting factor (denoted by λ), the RLS-TOT algorithm requires three forgetting factors, i.e., λ_{11}, λ_{12}, and λ_2. Also, the RLS-NKP algorithm [6] uses two forgetting factors, λ_1^* and λ_2^*, which correspond to the two adaptive filters. It is known that choosing the value of a forgetting factor involves a compromise between the main performance criteria, i.e., fast convergence/tracking and low misalignment. In general, the value of a forgetting factor $\lambda_\star$ can be related to the associated filter length (generally denoted by $L_\star$), according to the relation [6]:

$$\lambda_\star = 1 - \frac{1}{KL_\star}, \tag{4}$$

with $K \geq 1$. A higher value of K leads to $\lambda_\star$ closer to one, which improves the accuracy of the solution, but sacrificing in terms of the tracking behavior. We should also outline that the initial convergence rate is not always relevant for the RLS-based algorithms, while the tracking is the true assessment [3].

As compared to the conventional RLS algorithm, the RLS-TOT provides a better flexibility related to the choice of these parameters, since it combines three adaptive filters of different lengths, but much shorter as compared to the length of the global filter. Since a shorter adaptive filter is usually related to a faster tracking capability, we could increase the values of RLS-TOT forgetting factors, aiming to improve its accuracy, while slightly sacrificing in terms of tracking. On the other hand, the forgetting factors of the conventional RLS algorithm and the RLS-NKP should be increased in order to improve their tracking behavior, while paying in terms of accuracy.

Such an approach is considered in Figure 2, where the forgetting factors of the comparing algorithms are set based on (4), using $K = 5$ for the conventional RLS algorithm and $K = 45$ for the decomposition-based versions, i.e., RLS-NKP and RLS-TOT. The experimental framework is echo cancellation, aiming to identify a network echo path from G168 Recommendation [11], with the length $L = 512$ (the

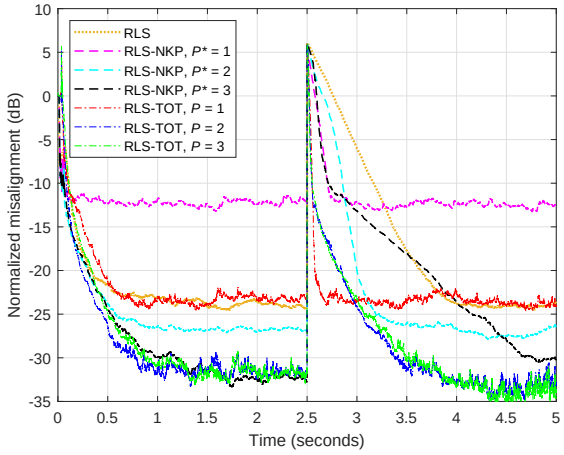

Figure 2. Misalignment of the RLS-based algorithms for the identification of a network impulse response of length $L = 512$. The forgetting factors are set based on (4), using $K = 5$ for the conventional RLS algorithm, and $K = 45$ for the RLS-NKP and RLS-TOT. The input signal is an AR(1) process and SNR = 20 dB.

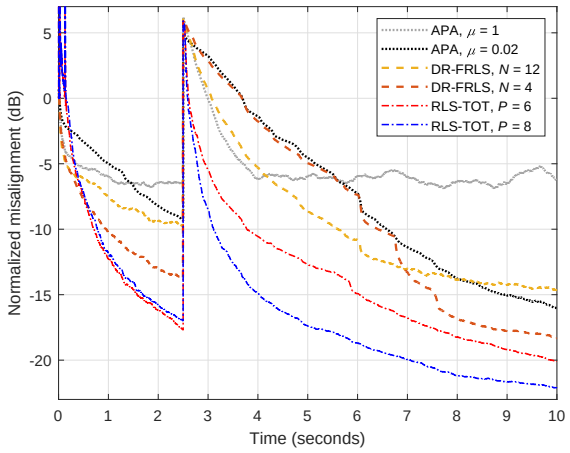

Figure 3. Misalignment of the APA, DR-FRLS algorithm, and RLS-TOT, for the identification of an acoustic impulse response of length $L = 2048$. The RLS-TOT uses λ_{11} and λ_{12} set based on (4), with $K = 100$, while $\lambda_2 = 1$. The input signal is speech and SNR $= 10$ dB.

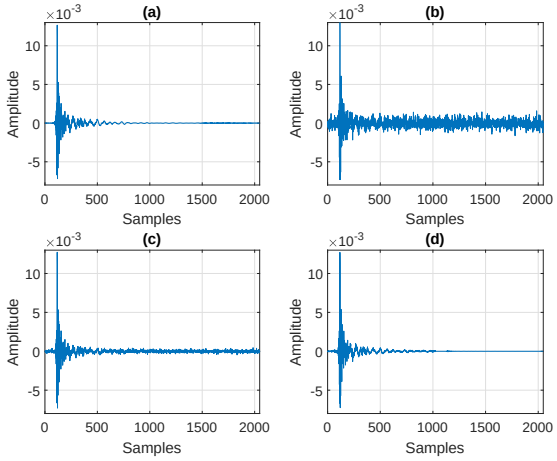

Figure 4. Impulse responses related to the experiment reported in Figure 3: (a) true acoustic impulse response $\mathbf{h}$; (b) $\widehat{\mathbf{h}}(n)$ obtained by APA using $\mu = 1$; (c) $\widehat{\mathbf{h}}(n)$ obtained by DR-FRLS using $N = 12$; and (d) $\widehat{\mathbf{h}}(n)$ obtained by RLS-TOT using $P = 8$.

sampling rate is 8 kHz). Thus, the decomposition setup uses $L_1^* = 32$ and $L_2^* = 16$ for the RLS-NKP, and $L_{11} = L_{12} = 16$ and $L_2 = 2$ for the RLS-TOT. The input signal $x(n)$ is a first order autoregressive process, AR(1), which is obtained by filtering a white Gaussian noise through an AR(1) transfer function with the pole at 0.8. The output of the echo path, $y(n)$, is corrupted by a white Gaussian noise, $w(n)$, while the signal-to-noise ratio (SNR) is set to 20 dB. In order to test the tracking capabilities, an abrupt change of the impulse response is considered, by changing the sign of the coefficients after 2.5 seconds. The performance measure is the normalized misalignment (in dB), which is evaluated as $20\log_{10}\left[\left\|\mathbf{h} - \widehat{\mathbf{h}}(n)\right\| / \|\mathbf{h}\|\right]$, where $\|\cdot\|$ denotes the Euclidean norm. As we can notice in Figure 2, the performance gain (i.e., misalignment/tracking) is clear in the favor of RLS-TOT, as compared to the RLS-NKP counterpart, for $P^* = P$. Besides, even if the tracking capability of the conventional RLS algorithm is improved when using a smaller value of K, it is significantly slower as compared to the RLS-TOT.

Another strategy that should be considered in case of the RLS-TOT is to set the maximum value of the forgetting factor (i.e., equal to 1) for the shortest filter of length L_2^2. In the common setups, this filter has only a few coefficients (e.g., 4 in our scenario), so that the tracking behavior of the global filter will be slightly affected, while still improving its misalignment. This is supported in Figure 3, in the framework of acoustic echo cancellation. This second experiment is dedicated to the identification of an acoustic impulse response of length $L = 2048$, using a speech signal as input and operating in a noisy environment with SNR $= 10$ dB. Therefore, the decomposition setup of the RLS-NKP considers $L_1^* = 64$ and $L_2^* = 32$, while the RLS-TOT uses $L_{11} = L_{12} = 32$ and $L_2 = 2$. Due to its high computational complexity, the conventional RLS algorithm is prohibitive in such scenarios that involve very long length impulse responses. Hence, other more practical algorithms are used in this simulation. First, we involve the Affine Projection Algorithm (APA) [12], which is a very popular choice in the framework of acoustic echo cancellation. The main parameters of APA are the step-size ($0 < \mu \leq 1$) and the projection order ($1 \leq M \ll L$). Higher values of μ and M improve the convergence rate and tracking of the algorithm, but increase the misalignment. In our scenario, we set $M = 8$ and use different values of μ to illustrate this behavior. Second, we involve in the experiment the recently developed Data-Reuse Fast RLS (DR-FRLS) algorithm [13]. This algorithm can operate with the maximum value of the forgetting factor (i.e., $\lambda = 1$), while the tracking capability is tuned based on the data-reuse parameter, denoted by N. As we can notice in Figure 3, the RLS-TOT outperforms both APA and the DR-FRLS algorithm, in terms of convergence rate/tracking and misalignment level.

The most relevant estimated impulse responses related to the experiment reported in Figure 3 are included in Figure 4. These are compared to the true acoustic impulse response $\mathbf{h}$ of length $L = 2048$ [shown in Figure 4(a)]. Here, we compare the estimates provided by three algorithms: i) APA using $\mu = 1$ [Figure 4(b)], ii) DR-FRLS using $N = 12$ [Figure 4(c)], and iii) RLS-TOT using $P = 8$ [in Figure 4(d)]. As we can notice, the accuracy of the estimated impulse response of RLS-TOT is significantly better as compared to the estimates obtained by the comparing algorithms.

In future works, it would be interesting to investigate time-dependent (i.e., variable) forgetting factors. In this manner, by varying these parameters within the iterations of the main algorithm, we can target a better compromise between the performance criteria. In echo cancellation, this translates in addressing several challenging situations, like the echo path change, the double-talk scenario (i.e., the two speakers talk simultaneously), and the background noise variation.

Copyright (c) IARIA, 2024. ISBN: 978-1-68558-142-8

In terms of double-talk robustness, an appealing approach would be to design variable regularization parameters for the RLS-TOT. To this purpose, the main relations of the algorithm (presented in Section II) need to be reformulated, by considering the covariance matrices

$$\mathbf{R}_{12,11}(n) = \sum_{k=1}^{n} \lambda_2^{n-k} \mathbf{x}_{12,11}(k) \mathbf{x}_{12,11}^T(k), \quad (5)$$

$$\mathbf{R}_{2,11}(n) = \sum_{k=1}^{n} \lambda_{12}^{n-k} \mathbf{x}_{2,11}(k) \mathbf{x}_{2,11}^T(k), \quad (6)$$

$$\mathbf{R}_{2,12}(n) = \sum_{k=1}^{n} \lambda_{11}^{n-k} \mathbf{x}_{2,12}(k) \mathbf{x}_{2,12}^T(k). \quad (7)$$

Their inverses are equivalent to the matrices $\mathbf{P}_{12,11}(n)$, $\mathbf{P}_{2,11}(n)$, and $\mathbf{P}_{2,12}(n)$. They can be iteratively evaluated as

$$\mathbf{R}_{12,11}(n) = \lambda_2 \mathbf{R}_{12,11}(n-1) + \mathbf{x}_{12,11}(n) \mathbf{x}_{12,11}^T(n), \quad (8)$$

$$\mathbf{R}_{2,11}(n) = \lambda_{12} \mathbf{R}_{2,11}(n-1) + \mathbf{x}_{2,11}(n) \mathbf{x}_{2,11}^T(n), \quad (9)$$

$$\mathbf{R}_{2,12}(n) = \lambda_{11} \mathbf{R}_{2,12}(n-1) + \mathbf{x}_{2,12}(n) \mathbf{x}_{2,12}^T(n), \quad (10)$$

so that the final updates of the filters result in

$$\widehat{\underline{\mathbf{h}}}_2(n) = \widehat{\underline{\mathbf{h}}}_2(n-1) + \mathbf{M}_{12,11}^{-1}(n) \mathbf{x}_{12,11}(n) e(n), \quad (11)$$

$$\widehat{\underline{\mathbf{h}}}_{12}(n) = \widehat{\underline{\mathbf{h}}}_{12}(n-1) + \mathbf{M}_{2,11}^{-1}(n) \mathbf{x}_{2,11}(n) e(n), \quad (12)$$

$$\widehat{\underline{\mathbf{h}}}_{11}(n) = \widehat{\underline{\mathbf{h}}}_{11}(n-1) + \mathbf{M}_{2,12}^{-1}(n) \mathbf{x}_{2,12}(n) e(n), \quad (13)$$

where $\mathbf{M}_{12,11}(n) = \mathbf{R}_{12,11}(n) + \delta_2 \mathbf{I}_{L_2^2}$, $\mathbf{M}_{2,11}(n) = \mathbf{R}_{2,11}(n) + \delta_{12} \mathbf{I}_{PL_{12}L_2}$, and $\mathbf{M}_{2,12}(n) = \mathbf{R}_{2,12}(n) + \delta_{11} \mathbf{I}_{PL_{11}L_2}$. The regularization parameters δ_2, δ_{12}, and δ_{11} can be designed in a time-dependent manner, as a function of the estimated SNR. In other words, a lower SNR should be associated to higher values of the regularization terms, which further slow down the adaptation process. This is the desired behavior in double-talk situations or noisy environments, where a low SNR level could significantly disturb the echo canceler.

IV. Conclusions

In this short paper, we have presented a tensorial RLS-based algorithm, which follows a recently developed method that splits the impulse response of the system based on a third-order tensor decomposition. The resulting RLS-TOT combines the estimates provided by three shorter adaptive filters, so that it is suitable for the identification of long length impulse responses. We further investigated different strategies for this algorithm, focusing on its main parameters, i.e., the forgetting factors and the regularization terms. Simulations performed in echo cancellation scenarios support the performance gain, in terms of converge/tracking and accuracy of the solution.

Acknowledgments

This work was supported by a grant of the Ministry of Research, Innovation and Digitization, CNCS–UEFISCDI, project number PN-III-P4-PCE-2021-0438, within PNCDI III. Dr. Camelia Elisei-Iliescu would like to acknowledge the support of ROMATSA. The authors would like to thank the three reviewers for the valuable comments and suggestions, which significantly improved the content of this paper.

References

[1] J. Benesty and Y. Huang, Eds., *Adaptive Signal Processing–Applications to Real-World Problems*. Berlin, Germany: Springer-Verlag, 2003.

[2] S. Haykin, *Adaptive Filter Theory*. Fourth Edition, Upper Saddle River, NJ, USA: Prentice-Hall, 2002.

[3] Y. V. Zakharov and V. H. Nascimento, "DCD-RLS adaptive filters with penalties for sparse identification," *IEEE Trans. Signal Processing*, vol. 61, pp. 3198–3213, June 2013.

[4] S. L. Gay and J. Benesty, Eds., *Acoustic Signal Processing for Telecommunication*. Boston, MA: Kluwer Academic Publisher, 2000.

[5] C. Paleologu, J. Benesty, and S. Ciochină, "Linear system identification based on a Kronecker product decomposition," *IEEE/ACM Trans. Audio, Speech, Language Processing*, vol. 26, pp. 1793–1808, Oct. 2018.

[6] C. Elisei-Iliescu et al., "Recursive least-squares algorithms for the identification of low-rank systems," *IEEE/ACM Trans. Audio, Speech, Language Processing*, vol. 27, pp. 903–918, May 2019.

[7] L.-M. Dogariu, J. Benesty, C. Paleologu, and S. Ciochină, "Identification of room acoustic impulse responses via Kronecker product decompositions," *IEEE/ACM Trans. Audio, Speech, Language Processing*, vol. 30, pp. 2828–2841, Sept. 2022.

[8] H. He, J. Chen, J. Benesty, and Y. Yu, "A frequency-domain recursive least-squares adaptive filtering algorithm based on a Kronecker product decomposition," in *Proc. IEEE ICASSP*, 2023, 5 pages.

[9] J. Benesty, C. Paleologu, and S. Ciochină, "Linear system identification based on a third-order tensor decomposition," *IEEE Signal Processing Lett.*, vol. 30, pp. 503–507, May 2023.

[10] C. Paleologu et al., "Recursive least-squares algorithm based on a third-order tensor decomposition for low-rank system identification," *Signal Processing*, vol. 213, id. 109216, Dec. 2023.

[11] *Digital Network Echo Cancellers*, ITU-T Recommendation G.168, 2012.

[12] K. Ozeki and T. Umeda, "An adaptive filtering algorithm using an orthogonal projection to an affine subspace and its properties," *Electron. Commun. Jpn.*, vol. 67-A, pp. 19–27, May 1984.

[13] C. Paleologu, J. Benesty, and S. Ciochină, "Data-reuse recursive least-squares algorithms," *IEEE Signal Processing Lett.*, vol. 29, pp. 752–756, Mar. 2022.

Copyright (c) IARIA, 2024. ISBN: 978-1-68558-142-8