POLITEHNICA UNIVERSITY OF BUCHAREST

Doctoral School of Electronics, Telecommunications and Information Technology

Decision No. 884 **from** 20-07-2022

Ph.D. THESIS SUMMARY

Alexandru-George RUSU

ALGORITMI ADAPTIVI CU APLICAȚII ÎN COMUNICAȚIILE AUDIO ȘI DE VOCE

ADAPTIVE ALGORITHMS WITH APPLICATIONS IN AUDIO AND VOICE COMMUNICATIONS

THESIS COMMITTEE

Prof. Dr. Eng. Ion MARGHESCU Politehnica Univ. of Bucharest	President
Prof. Dr. Eng. Silviu CIOCHINĂ Politehnica Univ. of Bucharest	PhD Supervisor
Prof. Dr. Eng. Doru-Florin CHIPER Gheorghe Asachi Technical Univ. of Iași	Referee
Prof. Dr. Eng. Dinu COLŢUC Valahia Univ. of Târgovişte	Referee
Prof. Dr. Eng. Constantin PALEOLOGU Politehnica Univ. of Bucharest	Referee

BUCHAREST 2022

Table of contents

Chapter 1

Introduction to Research Field

1.1 Presentation of the field of the doctoral thesis

The echo cancellation is one of the most interesting applications of signal processing. Even if its cause is due to the coupling between the speaker and the microphone of a real device (i.e., acoustic echo), or it is obtained due to a mismatch in an electrical circuit (i.e., line echo), reflecting back to the source a part of the signal, one of the main solutions to cancel it is based on the system identification framework. In the system identification framework, the principle is to identify a system as close as possible to the system that produces the echo signal. Therefore, even though the real system contributes to the echo or maybe the acoustics of the environment affects in one way or another the echo (i.e., in acoustic echo cancellation), an estimate as accurate as possible should be obtained so that in the end, the echo is eliminated. Definitely, the system that must be identified can change in time, and because of this, the algorithm used for system identification is an adaptive algorithm that is able to track unpredictable behaviors.

The literature provides many adaptive algorithms, and even if their adaptive character should cope successfully with any situation, each of them is more efficient in certain scenarios with more or less computational complexity. Some of them are purely theoretical concepts used to derive state-of-the-art benchmarks and represent paths to new research.

There are also systems that present a high complexity, or by their nature produce other effects such as reverberation, delays, or nonlinearities. Different techniques were studied in the past years, but there is no single solution agreed upon, while it depends on the application and the effects that should be eliminated or reduced. Even the effects themselves can be of several types, which makes it difficult to find a single solution. However, a widespread approach is based on a multilinear system that combines either parallel or cascaded linear or non-linear adaptive filters [16].

1.2 Scope of the doctoral thesis

The purpose of the thesis is to develop new algorithms or improved variants of the existing ones for the identification of unknown systems through adaptive filters which overcome the existing solutions and aim real-time applications. Moreover, the thesis develops optimization methods that improve the overall performance of the adaptive algorithms. In the end, solutions for multi-path signals (e.g., the reverberation effect) that can occur in practical applications are developed, aiming to reduce the computational complexity of the rudimentary algorithms.

1.3 Content of the doctoral thesis

The thesis contains four main chapters and a final one that comprehends a summary of the thesis, the list of contributions, and future perspectives for research and development. The first chapter of the four main chapters consists of a short introduction to the theory of the adaptive algorithms which were used as starting point for the research of this thesis. Further, the other three main chapters represent the main contributions of the thesis and introduce the theoretical development along with experimental results and conclusions. The results are depicted in each contribution chapter to facilitate a better structure and view of the context. Further, the content of the chapters is summarized.

In chapter 2, we introduce some of the most popular adaptive algorithms. In Chapter 3, we propose two new algorithms, which follow an optimization criterion that is more practical for real-world applications. For each algorithm, we have developed an extension for a particular use case, i.e., when the input signal is considered white Gaussian noise.

Next, in Chapter 4, by using the data-reuse technique we aimed to improve the performance of one of the algorithms developed in Chapter 3 and maintain its linear computational complexity. Then, we have analyzed the data-reuse from a different perspective and developed a method to generate the fixed step-sizes of the NLMS algorithm and APA, so that we have obtained two new variable step-size algorithms.

Further, through Chapter 5 we introduce multiple structures without and with memory in the context of a multiple-input/single-output system. By developing these structures, we have obtained a way of splitting a long length filter into a cascade of shorter length filters, which form a single-input/single-output (SISO) system. Then, we have presented two approaches to using the input data to facilitate two contexts for the identification of linear and multilinear systems.

Finally, because the conclusions were presented in each chapter, in Chapter 6 we only summarize the main results. Also, we outline the main contributions and some perspectives for future research and development.

Chapter 2

Introduction to Adaptive Algorithms

This chapter represents a walkthrough of some of the most popular adaptive algorithms in the literature, which supported the research and development of the novel ideas and algorithms in this thesis.

2.1 System Model

In the system identification framework, the configuration represented in Fig. 2.1 is used. In this setup, $x(n)$ represents the input signal at the discrete-time index n, while $y(n)$ is the output of the system that must be determined (i.e., $\mathbf{h}$).

By filtering the input signal through both systems, we can write $\hat{y}(n) = \hat{\mathbf{h}}^T(n-1)\mathbf{x}(n)$ and $y(n) = \mathbf{h}^T\mathbf{x}(n)$, where $\mathbf{x}(n)$ is a vector containing the L most recent input samples.

In the end, the output of the adaptive system is subtracted from the desired signal, so that the error signal $e(n)$ results as $e(n) = d(n) - \hat{y}(n)$.

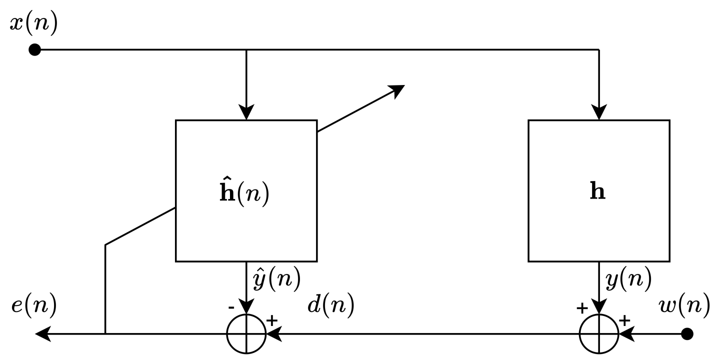

Fig. 2.1 System model.

2.2 Least-Mean-Square (LMS) Algorithm

The Wiener filter facilitated the development of one of the most popular adaptive algorithms, the LMS algorithm. The popularity of the LMS algorithm is due to its ease of understanding, implementing, and use of it. In the beginning, the Wiener filter aimed to determine an optimal filter, $\hat{\mathbf{h}}_{\mathbf{W}}$, that minimizes the error signal in each iteration.

In this context, the optimal filter that minimizes the error signal is identified by minimizing a cost function, the mean-square error (MSE), defined as $J[\hat{\mathbf{h}}(n)] = E[e^2(n)]$, where $E[\cdot]$ represents the statistical expectation. After several computations the optimal Wiener filter is obtained as $\hat{\mathbf{h}}_{\mathbf{W}} = \mathbf{R}_{\mathbf{x}}^{-1}\mathbf{p}_{\mathbf{x}d}$.

The computation of the cross-correlation vector $\mathbf{p}_{\mathbf{x}d}$ and the correlation matrix $\mathbf{R}_{\mathbf{x}}$, imply knowledge of statistical estimates, making the steepest-decent algorithm unsuitable for practical use. If we consider the instantaneous values of $\mathbf{p}_{\mathbf{x}d}$ and $\mathbf{R}_{\mathbf{x}}$ as

$$\mathbf{p}_{\mathbf{x}d}(n) = d(n)\mathbf{x}(n),\ \mathbf{R}_{\mathbf{x}}(n) = \mathbf{x}(n)\mathbf{x}^T(n), \tag{2.16}$$

the following equation results

$$\hat{\mathbf{h}}(n) = \hat{\mathbf{h}}(n-1) + 2\mu\mathbf{x}(n)e(n), \tag{2.17}$$

which defines the update equation of the LMS algorithm.

2.3 Normalized LMS Algorithm

In order to provide a higher degree of freedom, a good choice is to have a time dependent step-size. A first possibility is to normalize the step-size μ by the instantaneous estimate of the input signal and the equation (2.17) can be rewritten as

$$\hat{\mathbf{h}}(n) = \hat{\mathbf{h}}(n-1) + \frac{\mu}{\mathbf{x}^T(n)\mathbf{x}(n)}\mathbf{x}(n)e(n). \tag{2.20}$$

Equation (2.20) defines the update equation of the NLMS algorithm.

2.4 Affine Projection Algorithm (APA)

The APA comes as an extension of the NLMS algorithm. While the LMS and NLMS algorithms are running on a single input vector, the APA is handling multiple input vectors through the input matrix $\mathbf{X}(n)$. The update equation of the APA is

$$\hat{\mathbf{h}}(n) = \hat{\mathbf{h}}(n-1) + \mu\mathbf{X}(n)[\mathbf{X}^T(n)\mathbf{X}(n) + \delta\mathbf{I_P}]^{-1}\mathbf{e}(n), \tag{2.30}$$

where $\mathbf{I_P}$ is the identity matrix of size $P \times P$.

2.5 Recursive Least-Squares (RLS) Algorithm

The RLS algorithm represents an important reference in adaptive filtering theory due to its high performance and robustness. This time, the statistical averages are replaced by the weighted sum of square temporal values. Thus, the cost function, instead of the MSE is replaced by the least-squares (LS) cost function, defined as

$$J[\hat{\mathbf{h}}(n)] = \sum_{k=1}^{n} \lambda^{n-k} \varepsilon^2(k), \tag{2.31}$$

for which, $\varepsilon(k)$ denotes the aposteriori error signal, with $\varepsilon(k) = d(k) - \hat{\mathbf{h}}(n)\mathbf{x}(k)$.

The development of the cost function results in the following recursive equation for the optimal filter

$$\hat{\mathbf{h}}(n) = \hat{\mathbf{h}}(n-1) + \mathbf{k}(n)e(n), \tag{2.45}$$

where $\mathbf{k}(n)$ denotes the Kalman gain defined as

$$\mathbf{k}(n) = \frac{\mathbf{R_x}^{-1}(n-1)\mathbf{x}(n)}{\lambda + \mathbf{x}^T(n)\mathbf{R_x}^{-1}(n-1)\mathbf{x}(n)}. \tag{2.46}$$

2.6 Peformance Metrics

When measuring the performance of an adaptive algorithm, we refer to three main characteristics: the initial convergence rate, the misadjustment of the coefficients, and the tracking capability. The convergence rate describes how fast the coefficients of the unknown system are determined or how fast the steady-state is reached, while the misadjustment indicates how close the estimated solution is to the real coefficients. However, in some cases the unknown system can vary or completely change (especially in acoustic echo cancellation scenarios) and through the tracking capability we analyze how fast the adaptive algorithm is able to determine the coefficients of the new system. All these characteristics can be analyzed through the normalized misalignment (NM), defined as

$$\text{NM}(n) = 10\log_{10}\frac{||\mathbf{h} - \hat{\mathbf{h}}(n)||_2^2}{||\mathbf{h}||_2^2} \text{ (dB)}. \tag{2.47}$$

When the estimated coefficients (i.e., $\hat{\mathbf{h}}(n)$) are close to the real coefficients (i.e., $\mathbf{h}$), the nominator of the NM tends to zero and the ratio as well. By this, we can remark that the logarithm values go to minus infinity.

Chapter 3

Optimized LMS Algorithms

This chapter introduces the first contribution to this thesis, representing two LMS-based algorithms with variable step-size. The literature is rich based on this subject, proposing different solutions, in general, based on heuristic considerations. Beggining with [15], a long series of algorithms appeared under the name of "proportionate algorithms" [29], [20], [18]. However, the development principle that stands behind the LMS algorithm (i.e., the minimization of the MSE cost function) is not so practical in the context of echo cancellation, while in real life applications we would like to obtain a better estimate of the echo signal, which means a smaller misalignment between the estimated impulse response, $\hat{\mathbf{h}}(n)$, and the unknown impulse response, $\mathbf{h}(n)$. The work of this chapter has been published in [6] and [9].

3.1 Design Principle

In this framework, we consider the unknown system, $\mathbf{h}(n)$, is changing dynamically and is described by a first-order Markov model [14][17], characterized by the noise $\delta_{\mathbf{h}}(n)$.

The first step in this development is to replace the fixed step-size in the update equation of the LMS algorithm with a control matrix, $\mathbf{Y}(n)$. We can remark that $\mathbf{Y}(n)$ matrix could be, a scaled unit matrix, leading to a variable step-size LMS algorithm, a diagonal matrix, leading to an algorithm with a differential step-size, or an arbitrary matrix. We define the autocorrelation matrix of the coefficients error as $\mathbf{R_c}(n) \triangleq \mathrm{E}[\mathbf{c}(n)\mathbf{c}^T(n)]$, where $\mathbf{c}(n) = \hat{\mathbf{h}}(n) - \mathbf{h}(n)$, denotes the aposteriori misalignment. Furthermore, we denote $\Gamma(n) \triangleq \mathbf{R_c}(n) + \sigma_{\mathbf{h}}^2\mathbf{I}_L$, where $\sigma_{\mathbf{h}}^2 = \mathrm{E}[\delta_{\mathbf{h}}^2(n)]$ represents the variance of the system noise, $\delta_{\mathbf{h}}(n)$.

3.2 The Optimized LMS (LMSO) Algorithm

In this development, we consider the first of the three cases presented in the previous subchapter, where $\mathbf{Y}(n)$ is a scaled unit matrix, so that $\mathbf{Y}(n) = \mu(n)\mathbf{I}_L$.

3.2.1 Autocorrelation Matrix of the Coefficients Error

Starting with the substituion of the aposteriori misalignment in the autocorrelation matrix of the coefficients error and developing it, the following result is obtained

$$\mathbf{R_c}(n) = \mathbf{\Gamma}(n-1) - \mu(n)[\mathbf{\Gamma}(n-1)\mathbf{R_x} + \mathbf{R_x}\mathbf{\Gamma}(n-1)] \\ + \mu^2(n)\{2\mathbf{R_x}\mathbf{\Gamma}(n-1)\mathbf{R_x} + \mathbf{R_x}\mathrm{tr}[\mathbf{\Gamma}(n-1)\mathbf{R_x}]\} + \mu^2(n)\sigma_w^2\mathbf{R_x}. \tag{3.13}$$

3.2.2 Optimum Step-Size Derivation

The optimum step-size is obtained by computing the minimum of MSD (i.e., $m(n) \triangleq \mathrm{E}[\|\mathbf{c}(n)\|_2^2] = \mathrm{tr}[\mathbf{R_c}(n)]$) and we obtain

$$\mu(n) = \frac{\mathrm{tr}[\mathbf{\Gamma}(n-1)\mathbf{R_x}]}{\mathrm{tr}\{\mathbf{\Gamma}(n-1)\mathbf{R_x}[L\sigma_x^2\mathbf{I}_L + 2\mathbf{R_x}]\} + L\sigma_x^2\sigma_w^2}. \tag{3.31}$$

When the input signal is white Gaussian noise, the optimum step-size can be written as

$$\mu_o(n) = \frac{m_o(n-1) + L\sigma_{\mathbf{h}}^2}{\sigma_x^2[m_o(n-1) + L\sigma_{\mathbf{h}}^2](L+2) + L\sigma_w^2}. \tag{3.34}$$

We can conclude that we have obtained two algorithms, the first one that is based on the general equation (3.31) called least-mean-square optimized "general" (LMSO-G) algorithm, and the second one that is based on the simplified equation (3.34) called least-mean-square optimized "white" (LMSO-W) algorithm.

3.2.3 Experimental Results

The simulations were performed in the context of acoustic echo cancellation. We can remark in Figure 3.6 that the $\sigma_{\mathbf{h}}^2$ parameter plays an important role in governing the performance of the LMSO-G and LMSO-W algorithms. With respect to the tracking capability, a small value of $\sigma_{\mathbf{h}}^2$ inhibits the re-convergence when the echo path changes, while a higher value enables the algorithms to determine the new system in a reasonable time.

3.3 The Optimized Differential Step-Size LMS (ODSS-LMS) Algorithm

In this development, we consider the second case of the three presented in Section 3.1 with the step-size $\mathbf{Y}(n) = \mathrm{matdiag}[\mathbf{y}(n)]$, where $\mathrm{matdiag}(\cdot)$ denotes a diagonal matrix, having on the main diagonal the elements of the vector $\mathbf{y}(n)$, leading to a differential or proportionate step-size.

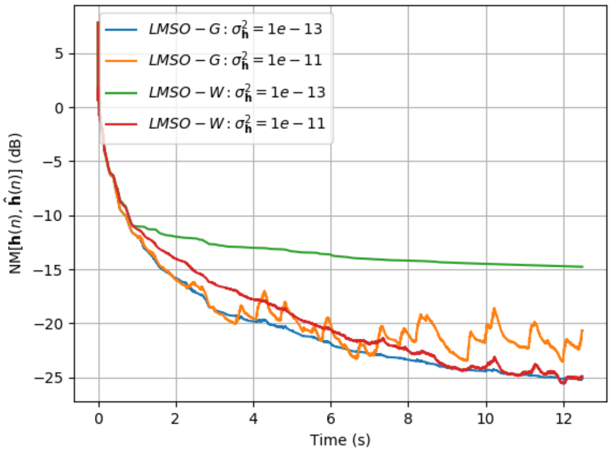

Fig. 3.6 Normalized misalignment of the LMSO-W and LMSO-G algorithms for different constant values of $\sigma_{\mathbf{h}}^2$. The input signal is a speech sequence and SNR $=$ 30dB.

3.3.1 Autocorrelation Matrix of the Coefficients Error

Proceeding in the same manner as in the previous section, the autocorrelation matrix of coefficients error results

$$\begin{aligned}\mathbf{R_c}(n) &= \mathbf{\Gamma}(n-1) - \mathbf{\Gamma}(n-1)\mathbf{R_x}\mathbf{Y}(n) - \mathbf{Y}(n)\mathbf{R_x}\mathbf{\Gamma}(n-1) \\ &\quad + \mathbf{Y}(n)\{2\mathbf{R_x}\mathbf{\Gamma}(n-1)\mathbf{R_x} + \mathbf{R_x}\mathrm{tr}[\mathbf{\Gamma}(n-1)\mathbf{R_x}]\}\mathbf{Y}(n) + \mathbf{Y}(n)\mathbf{R_x}\mathbf{Y}(n)\sigma_w^2. \qquad (3.41)\end{aligned}$$

3.3.2 Optimum Step-Size Vector Derivation

Onwards, we obtain the optimum step-size by computing the gradient of MSD with respect to $\mathbf{y}(n)$, having the solution:

$$\begin{aligned}\mathbf{y_o}(n) &= \left\{\sigma_x^2\mathbf{I}_L\{\mathrm{tr}[\mathbf{\Gamma}(n-1)\mathbf{R_x}] + \sigma_w^2\} + 2\mathrm{matdiag}[\mathbf{R_x}\mathbf{\Gamma}(n-1)\mathbf{R_x}]\right\}^{-1} \\ &\quad \times \mathrm{diag}[\mathbf{R_x}\mathbf{\Gamma}(n-1)]. \qquad (3.52)\end{aligned}$$

Again, if we consider that the input signal is white Gaussian noise and $\mathbf{R_x} = \sigma_x^2\mathbf{I}_L$, we obtain the following result for step-size

$$\mathbf{y_o}(n) = \left\{\sigma_x^2\mathbf{I}_L[m(n-1) + \sigma_{\mathbf{h}}^2(L+2)] + 2\sigma_x^2\mathrm{matdiag}[\mathbf{R_c}(n-1)] + \sigma_w^2\mathbf{I}_L\right\}^{-1}\gamma(n-1), \qquad (3.55)$$

with $\gamma(n) \triangleq \mathrm{diag}[\mathbf{\Gamma}(n)]$.

At this point, we can conclude that we have obtained two algorithms, the first one conducted by equation (3.52) called optimized differentiate least-mean-square "general" (ODSS-LMS-G) algorithm and the second one based on equation (3.55) called optimized differentiate least-mean-square "white" (ODSS-LMS-W).

3.3.3 Experimental Results

Fig. 3.14 highlights that both ODSS-LMS algorithms outperform the JO-NLMS [27] and IPNLMS [20] algorithms in terms of normalized misalignment level. We can remark the good performance of the ODSS-LMS-W algorithm which achieves a normalized misalignment level of -30dB in less than 1 second.

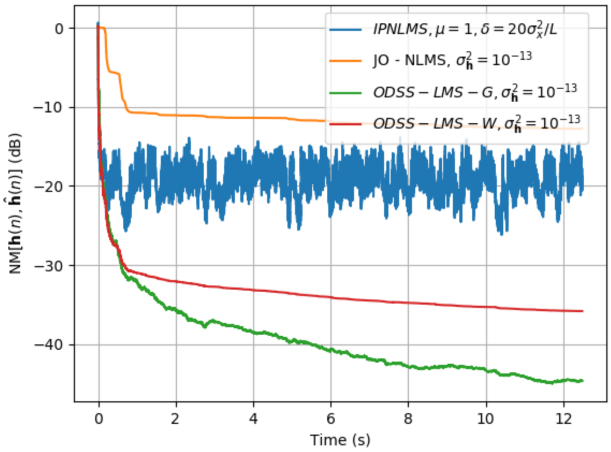

Fig. 3.14 Normalized misalignment of the IPNLMS, JO-NLMS, ODSS-LMS-G, and ODSS-LMS-W algorithms. The input signal is a speech sequence. The unknown system is the network echo path.

3.4 Practical Considerations

Across the development of both LMSO and ODSS-LMS algorithms, we have introduced parameters that must be estimated when we deal with real-time applications. The first one is the power of the input signal, σ_x^2, which can be easily estimated as $\sigma_x^2 = \frac{1}{L}\mathbf{x}^T(n)\mathbf{x}(n)$, or through a 1-pole exponential filter. Then, the autocorrelation matrix of the input signal can be estimated through a weighting window $\tilde{\mathbf{R}}_{\mathbf{x}}(n) = \lambda\tilde{\mathbf{R}}_{\mathbf{x}}(n-1) + (1-\lambda)\mathbf{x}(n)\mathbf{x}^T(n)$, where λ denotes the forgetting factor. If the power of the σ_w^2 noise is not known, it should be estimated as well, and for this, we can refer to [13], [25].

3.5 Computational Complexity Analysis

The overall computational complexity of the developed algorithms is summarized in Table 3.1 and Table 3.2.

3.6 Summary and Conclusions

Using the LMS algorithm as starting point and considering the minimization of the MSD as cost function, we have developed the LMSO-G and ODSS-LMS-G algorithms. Based

Table 3.1 Computational complexity comparison between LMS and LMSO algorithms

Algorithm / Operation	LMS	LMSO-G	LMSO-W
Multiplications	$2L+1$	$4L^3+7L^2+2L+4$	$2L+7$
Additions	$2L$	$2L^3+2L^2+3L^2(L-1)+5L$	$2L+3$
Divisions	0	1	1

Table 3.2 Computational complexity comparison between LMS and ODSS-LMS algorithms

Algorithm / Operation	LMS	ODSS-LMS-G	ODSS-LMS-W
Multiplications	$2L+1$	$3L^3+2L^2+16L$	$10L+3$
Additions	$2L$	$3L^3-2L^2+9L$	$10L+1$
Divisions	0	L	L

on a particular use case (i.e., considering that the input signal is white Gaussian noise), we have obtained the simplified versions, the LMSO-W and ODSS-LMS-W algorithms. The general versions, LMSO-G and ODSS-LMS-G, present a computational complexity of $O(L^3)$, while the simplified verions, LMSO-W and ODSS-LMS-W, are characterized by a linear computational complexity (i.e., $O(L)$).

As the simulations state, the LMSO-G and ODSS-LMS-G algorithms are robust when handling correlated input signals and present good performance in terms of convergence rate, normalized misalignment, and tracking capability. However, due to the high computational complexity, they are more suitable for benchmarking instead of practical use. Even if the simplified versions were developed considering white Gaussian signal as input, the simulation highlighted their good performances even when the input signal is correlated. Of course, we have remarked a drawback in performance for the LMSO-W and ODSS-LMS-W algorithms compared to their counterparts LMSO-G and ODSS-LMS-G algorithms, but we have obtained a big reduction in computational complexity instead. Thus, we can outline that the LMSO-W and ODSS-LMS-W algorithms are good candidates for real-time applications.

Also, the ODSS-LMS type algorithms have the benefit of the differentiate step-size, making them a good choice when the echo path is sparser. A new parameter was introduced, $\sigma_{\mathbf{h}}^2$, and the simulations presented its importance. According to the simulations, the $\sigma_{\mathbf{h}}^2$ parameter should be tuned so that to achieve a good compromise between low normalized misalignment level and tracking capability.

Chapter 4

Data-Reuse based Algorithms

4.1 Baseline Concept

Initially, this method was introduced in [10]. The data-reuse method implies the update of the coefficients at each time index, considering M data samples from the past, where M denotes the data-reuse order.

4.2 Data-Reuse LMSO (LMSO-DR) Algorithm

In the previous chapter, we introduced the LMSO-G and LMSO-W algorithms. We have noticed a drastic reduction in terms of the computational complexity for the LMSO-W algorithm (i.e, $O(L)$ compared to $O(L^3)$ for the LMSO-G algorithm), which impacted the overall performance. By using a more robust data-reuse method as in [12], we would like to achieve the performance of the LMSO-G algorithm by keeping the linear computational complexity.

4.2.1 Algorithm Development

Considering the LMSO-W algorithm, we can apply the data-reuse method and develop the data-reuse LMSO-W (LMSO-W-DR) algorithm.

We would like to obtain an equation that updates the filter's coefficients in one step, so that to avoid the M iterations. Therefore, considering [12] as a reference, we can start substituing the equations from the first iteration in the equations of the second iteration, and following this procedure until the Mth data-reuse order, two equivalent equations result

$$\mathbf{e}(n) = \mathbf{d}(n) - \mathbf{X}^T(n)\hat{\mathbf{h}}(n-1), \tag{4.15}$$

$$\hat{\mathbf{h}}(n) = \hat{\mathbf{h}}(n-1) + \mathbf{X}(n)\mathbf{Z}(n)\mathbf{e}(n), \tag{4.16}$$

where $\mathbf{Z}(n)$ is the step-size matrix defined as $\mathbf{Z}(n) = \mathbf{K}^{-1}(n)\mathbf{S}(n)$, with $\mathbf{S}(n)$, a diagonal $M \times M$ matrix, built with the step-size of the LMSO-W algorithm. It can be noticed that $\mathbf{K}(n)$ term opens a connection with the APA [22].

4.2.2 Experimental Results

The system to be identified is the acoustic echo path. In Fig. 4.2 the input signal is an AR(1) process with $\beta = 0.9$ and the echo path changes in the middle of the simulation. We analyze the impact of the data-reuse order on the convergence rate. As expected, a higher data-reuse order, i.e., $M = 4$, improves the convergence rate and in this case, the LMSO-W-DR algorithm is even faster than the LMSO-G algorithm. It can be remarked that using the smallest data-reuse order, i.e, $M = 2$, the convergence rate is faster than that of the LMSO-G algorithm.

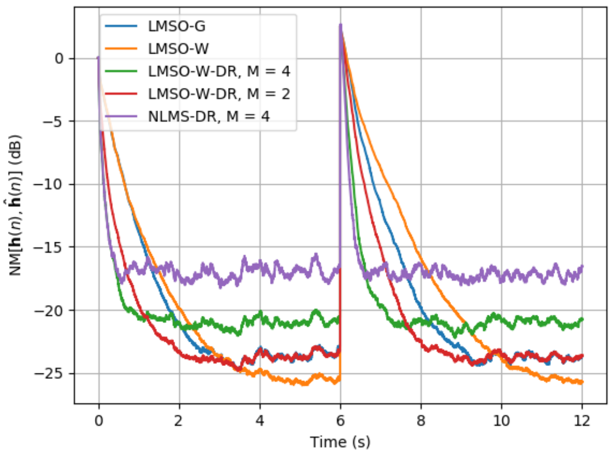

Fig. 4.2 Normalized misalignment of the LMSO-W-DR, LMSO-W, LMSO-G, and NLMS-DR algorithms. For the LMSO type algorithms the constant value is set $\sigma_{\mathbf{h}}^2 = 1e-11$, while for the NLMS-DR algorithm $\mu = 1$; the echo path changes at time 5s. The input signal is an AR(1) process ($\beta = 0.9$), $L = 512$, and SNR = 20dB.

4.3 Variable Step-Size NLMS (VSS-NLMS) Algorithm based on Data-Reuse

Following the baseline concept introduced in Section 4.1 and considering the NLMS algorithm presented in Section 2.3 as workhorse, we develop a variable step-size NLMS (VSS-NLMS) algorithm [3] for which the step-size is governed by a simple set of rules.

4.3.1 Algorithm Development

First of all, let us develop equation (2.20) for each data-reuse iteration as we did for the LMSO-W-DR algorithm, but taking into account fixed data (e.g., $\mathbf{x}(n)$, $d(n)$) [12], [19],

[26], [21], and analyzing the evolution of the step-size μ. By developing the equations of the NLMS algorithm for each iteration and replacing the equations from the first iteration in the equation from the second iteration, proceeding in the same manner until the kth order. Therefore, the update equation in the kth iteration is

$$\hat{\mathbf{h}}_k(n) = \hat{\mathbf{h}}(n-1) + \left[1-(1-\mu)^{k+1}\right] \frac{\mathbf{x}(n)e(n)}{\mathbf{x}^T(n)\mathbf{x}(n)}, \tag{4.30}$$

with the kth a posteriori error $e_k(n) = (1-\mu)^k e(n)$. In this context, the update equation of the VSS-NLMS algorithm becomes

$$\hat{\mathbf{h}}(n) = \hat{\mathbf{h}}(n-1) + \frac{\mu(n)}{\mathbf{x}^T(n)\mathbf{x}(n)}\mathbf{x}(n)e(n). \tag{4.35}$$

Then, we can generate a set of step-sizes, each of them used for a certain number of iterations. In order to not stall the algorithm, we can use mechanism that detects changes in the unknown system, such as [28] and resets the use of the generated step-sizes

4.3.2 Experimental Results

For these simulations we have used the acoustic echo path.

We simulate the VSS-NLMS algorithm in the scenario represented in Fig. 4.15, where the input signal is speech sequence. In this scenario, the SNR changes after 50 seconds from 20 dB to 10 dB. Initially, the variance σ_w^2 is considered to be known for the NPVSS-NLMS algorithm, but not when SNR change occurs. Even if the NPVSS-NLMS algorithm overcomes the VSS-NLMS algorithm in terms of convergence rate and normalized misalignment level, the VSS-NLMS algorithm is definitely more robust to power level changes.

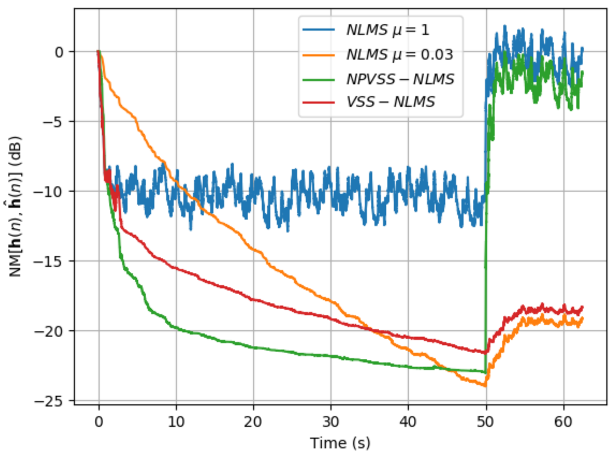

Fig. 4.15 Normalized misalignment of the NLMS algorithm with $\mu = 1$ and $\mu = 0.03$, normalized misalignment of the NPVSS-NLMS algorithm, and normalized misalignment of the VSS-NLMS algorithm using $l = 0.5$ and the lower bound $\mu = 0.03$. The input signal is speech, $L = 512$, and the SNR decreases from 20 dB to 10 dB after 50 seconds.

4.4 Variable Step-Size APA (VSS-APA)

As the APA is an extension of the NLMS algorithm, implementing a projection order P, it is natural at this point to continue the development of a variable step-size APA (VSS-APA).

4.4.1 Algorithm Development

Following the procedure from Section 4.3, definitely we can improve the overall performance, maintaining the same computational complexity.

Based on the mathematical induction until the kth data-reuse iteration we obtain

$$\hat{\mathbf{h}}_k(n) = \hat{\mathbf{h}}(n-1) + [1-(1-\mu)^k]\mathbf{X}(n)[\mathbf{X}^T(n)\mathbf{X}(n)]^{-1}\mathbf{e}(n) \tag{4.50}$$

and $\mathbf{e}_k(n) = (1-\mu)^{k-1}\mathbf{e}(n)$.

At this point, we can design the VSS-APA using the same principle used for the VSS-NLMS algorithm, by generating a set of step-sizes, each of them used for a certain number of iterations. A small step-size μ could stall the algorithm to not reacting to any changes in the unknown system that is identified. For this reason, we can use a detector [25] that resets the use of the generated step-sizes.

4.4.2 Experimental Results

The unknown system to be determined is represented by the acoustic echo path.

Forwards, the input signal is a speech sequence. In Fig. 4.22, the echo path changes in the middle of the simulation. Even if the convergence rate of APA with $\mu = 1$ is similar to that of the VSS-APA, APA presents variations of more than 5 dB in some scenarios. Also, the VSS-APA continues to achieve a much lower normalized misalignment level than APA and maintains the good performance after echo path change.

4.5 Summary and Conclusions

In this chapter, starting from the data-reuse approach we have developed three adaptive algorithms: LMSO-W-DR, VSS-NLMS, and VSS-APA. The LMSO-W-DR algorithm was developed by optimizing the data-reuse approach and improving the performance of it compared to its forerunner, i.e., LMSO-W algorithm, presented in Section 4.2. Our goal was to obtain an algorithm that provides a performance close to that of the LMSO-G algorithm and the simulations outlined that we have successfully reached this in terms of convergence rate, tracking, and good estimates. The performance gain of the LMSO-W-DR algorithm is achieved with a linear computational complexity, that is

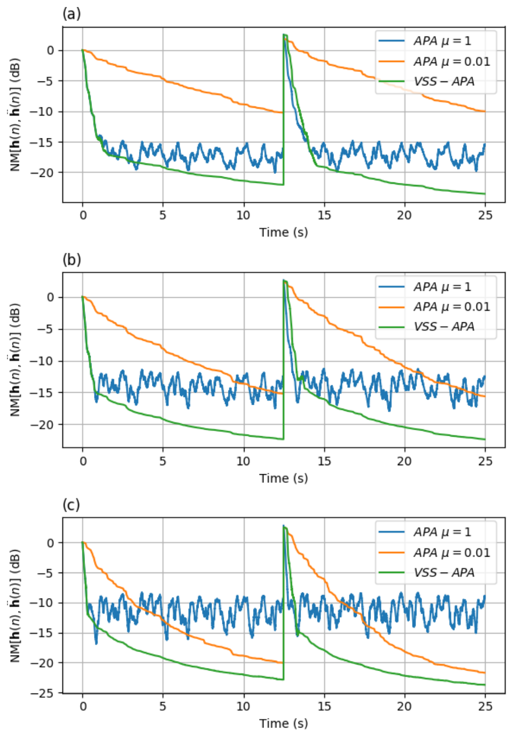

Fig. 4.22 Normalized misalignment of the APA with $\mu = 1$ and $\mu = 0.01$, and normalized misalignment of the VSS-APA using the step-size $\mu = 0.01$ and $l = 0.1$, for different values of the projection order (a) $P = 2$; (b) $P = 4$; and (c) $P = 8$. The input signal is a speech sequence, $L = 512$, and SNR = 30 dB.

$O(ML)$, with $M \ll L$. Also, the LMSO-W-DR opens a connection with the APA which can be exploited in future works.

Further, we have used the data-reuse approach to analyze the convergence rate of the NLMS algorithm and APA, and we have developed a method of step-sizes generation that represents the core part of the obtained algorithms: VSS-NLMS and VSS-APA. The simulations outlined the good performance compared to their forerunners, NLMS algorithm and APA, considering all perspectives. A remarkable advantage of VSS-NLMS algorithm and VSS-APA is the robustness during SNR variations, which in a real-time conversation is a usual use case. While the VSS-NLMS algorithm and VSS-APA introduce the use of a step-size reset mechanism, their computational complexity is still the same as for the NLMS algorithm and APA, introducing only a few more operations. Furthermore, while APA requires a higher projection order P to improve the performance, the VSS-APA obtains much better performance with a smaller projection order, which reduces the computational complexity.

The methods developed and presented in this chapter can be used in future algorithms and to facilitate the performance of any kind of algorithm that implies fixed parameters. The simulations support that LMSO-W-DR, VSS-NLMS, and VSS-APA are good candidates for real-time applications, that require a low computational complexity as well.

Chapter 5

Cascaded Adaptive Filters

System identification or tracking capability are not the only problems among the applications that involve echo cancellation. In real-world applications, the echo path can reach thousands of coefficients. Therefore, the performance of the algorithm can be affected in terms of convergence rate, estimates precision, or tracking, but equally important, the computational complexity increases with the length of the echo path. The literature provides solutions that imply an adaptive filter with a FIR structure, targetting sparse in nature echo paths as well. Further, the environment (i.e., echo path) can be characterized by different properties which can lead to the generation of multiple reflections, that is another challenge and in real-world applications, it is called the reverberation effect.

In this chapter, approaching a multilinear perspective, we develop solutions to these complex problems, aiming to minimize the computational complexity, obtain good performance, and handle the reverberation effect. In order to assess the reverberation effect from a mathematical point of view, we describe the impulse response using the Kronecker product decomposition. The development of this chapter is published in [2], [7], [8].

5.1 Multilinear Structures without Memory

A MISO system can be tailored through the following input-output relationship

$$y(n) = \sum_{l_1=1}^{L_1} \sum_{l_2=1}^{L_2} \cdots \sum_{l_N=1}^{L_N} x_{l_1 l_2 \ldots l_N}(n) h_{1,l_1} h_{2,l_2} \ldots h_{N,l_N}, \tag{5.1}$$

described by the coefficients h_{i,l_i}. Further, let us rewrite equation (5.1) as

$$y(n) = \sum_{l_1=1}^{L_1} h_{1,l_1} \cdots \sum_{l_{N-1}=1}^{L_{N-1}} h_{N-1,l_{N-1}} s_{l_{N-1}}(n), \tag{5.3}$$

where

$$s_{l_{N-1}}(n) = \sum_{l_N=1}^{L_N} x_{l_1 l_2 \ldots l_N}(n) h_{N,l_N} \tag{5.4}$$

denotes the output of the Nth MISO structure of L_N inputs. Thus, Fig. 5.1(a) depicts the $s_{l_{N-1}}$ MISO structure with the L_N inputs, while Fig. 5.1(b) presents its symbolic representation which is a memoryless weighted adder.

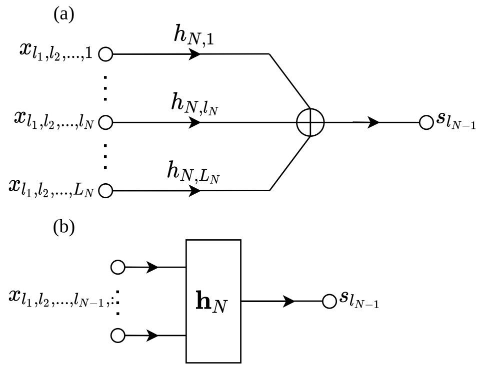

Fig. 5.1 (a) Representation of the MISO structure $s_{l_{N-1}}$; (b) The symbolic representation of $s_{l_{N-1}}$ (memoryless weighted adder).

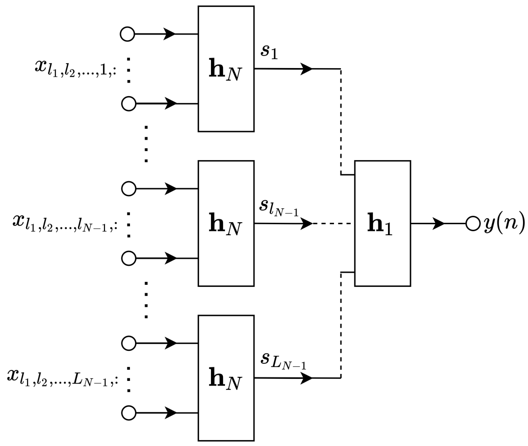

Fig. 5.2 The MISO system of N weighted adder levels.

Based on (5.3) and considering the symbolic representation from Fig. 5.1(b), we can sketch the MISO system through the graphical representation from Fig. 5.2, which encompasses N levels of weighted adders.

5.2 Multilinear Structures with Memory

The transformation of the structure $s_{l_{N-1}}$ into a structure with memory can be achieved by inserting a delay line, so that we can write

$$s_{l_{N-1}}(n) = \sum_{l_N=1}^{L_N} x_{l_{N-1}}(n - l_N + 1) h_{N,l_N}, \tag{5.5}$$

with $x_{l_1 l_2 \ldots l_N}(n) = x_{l_{N-1}}(n - l_N + 1),\ l_N = 1, 2, \cdots, L_N$.

Consequently, the MISO structure $s_{l_{N-1}}$ becomes a SISO structure and its output is generated by a transversal FIR filter $\mathbf{h}_N$. Proceeding in the same manner on each layer of the global system, we obtain in the end a cascade of transversal FIR filters, as it is shown in Fig. 5.6.

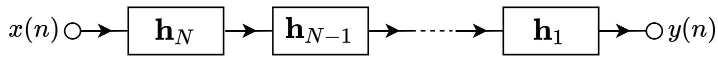

Fig. 5.6 SISO system in cascaded configuration.

5.3 System Model based on Kronecker Product Decomposition

Considering the system identification framework as before, let us introduce the output of the MISO system as

$$y(n) = \boldsymbol{\mathcal{X}}(n) \times_1 \mathbf{h}_1^T \times_2 \mathbf{h}_2^T \times_i \ldots \times_N \mathbf{h}_N^T, \tag{5.9}$$

where $\mathbf{h}_i$ represents the ith transversal filter from the cascaded depicted in Fig. 5.6, $\times_i$ denotes the multiplication operation by the dimension, $i = 1, 2, \ldots, N$, and N represents the multilinear degree.

We can rewrite (5.9) as $y(n) = \text{vec}^T[\boldsymbol{\mathcal{X}}(n)]\text{vec}(\boldsymbol{\mathcal{H}})$, where $\boldsymbol{\mathcal{H}}$ is a rank-1 tensor of dimension $L_1 \times L_2 \times \cdots \times L_N$ [11][24], or $y(n) = \widetilde{\mathbf{x}}^T(n)\mathbf{h}$. The input vector $\widetilde{\mathbf{x}}(n)$ and the coefficients vector $\mathbf{h}$ have $L_1 L_2 \cdots L_N$ elements. We can conclude that the multilinear perspective is translated into a linear problem by the way of processing the input data.

5.4 Z-Transform Analysis

The output of the global system from Fig. 5.6 can be developed to

$$Y(z) = S_{l_1}(z)H_1(z) = X_1(z)H_2(z^{L_1})H_1(z) = X_1(z)H(z), \tag{5.24}$$

where the transfer function $H(z) = H_2(z^{L_1})H_1(z)$. In this transfer function, we can outline that $H_2(z^{L_1})$ results from the interpolation with zeroes by the L_1 factor of the $H_2(z)$ function and is characterized by L_2 non-zero coefficients from a total length of $L_1(L_2 - 1) + 1$ coefficients.

5.5 Cascaded Multilinear NLMS Algorithm

Since the NLMS algorithm is very popular among the adaptive algorithms due to its simplicity and low computational complexity, let us use it as a workhorse and prove the cascaded filters approach. In this section, we develop the cascaded multilinear form NLMS (NLMS-CMF) algorithm by minimizing the MSE cost functions for the cascaded system.

5.5.1 Algorithm Development

In this development we obtain the equations of the cascaded multilinear form NLMS (NLMS-CMF) algorithm

$$
\begin{aligned}
\hat{\mathbf{h}}_1(n) &= \hat{\mathbf{h}}_1(n-1) + \frac{\mu_{\hat{\mathbf{h}}_1}\widetilde{\mathbf{x}}_{\hat{\mathbf{h}}_2\hat{\mathbf{h}}_3\cdots\hat{\mathbf{h}}_N}(n)e_{\hat{\mathbf{h}}_2\hat{\mathbf{h}}_3\cdots\hat{\mathbf{h}}_N}(n)}{\widetilde{\mathbf{x}}^T_{\hat{\mathbf{h}}_2\hat{\mathbf{h}}_3\cdots\hat{\mathbf{h}}_N}(n)\widetilde{\mathbf{x}}_{\hat{\mathbf{h}}_2\hat{\mathbf{h}}_3\cdots\hat{\mathbf{h}}_N}(n) + \delta_{\hat{\mathbf{h}}_1}}, \\
\hat{\mathbf{h}}_2(n) &= \hat{\mathbf{h}}_2(n-1) + \frac{\mu_{\hat{\mathbf{h}}_2}\widetilde{\mathbf{x}}_{\hat{\mathbf{h}}_1\hat{\mathbf{h}}_3\cdots\hat{\mathbf{h}}_N}(n)e_{\hat{\mathbf{h}}_1\hat{\mathbf{h}}_3\cdots\hat{\mathbf{h}}_N}(n)}{\widetilde{\mathbf{x}}^T_{\hat{\mathbf{h}}_1\hat{\mathbf{h}}_3\cdots\hat{\mathbf{h}}_N}(n)\widetilde{\mathbf{x}}_{\hat{\mathbf{h}}_1\hat{\mathbf{h}}_3\cdots\hat{\mathbf{h}}_N}(n) + \delta_{\hat{\mathbf{h}}_2}}, \\
&\vdots \\
\hat{\mathbf{h}}_N(n) &= \hat{\mathbf{h}}_N(n-1) + \frac{\mu_{\hat{\mathbf{h}}_N}\widetilde{\mathbf{x}}_{\hat{\mathbf{h}}_1\hat{\mathbf{h}}_2\cdots\hat{\mathbf{h}}_{N-1}}(n)e_{\hat{\mathbf{h}}_1\hat{\mathbf{h}}_2\cdots\hat{\mathbf{h}}_{N-1}}(n)}{\widetilde{\mathbf{x}}^T_{\hat{\mathbf{h}}_1\hat{\mathbf{h}}_2\cdots\hat{\mathbf{h}}_{N-1}}(n)\widetilde{\mathbf{x}}_{\hat{\mathbf{h}}_1\hat{\mathbf{h}}_2\cdots\hat{\mathbf{h}}_{N-1}}(n) + \delta_{\hat{\mathbf{h}}_{N-1}}}, \qquad (5.34)
\end{aligned}
$$

where $\delta_{\hat{\mathbf{h}}_i}$ denotes the regularization parameter for each $\hat{\mathbf{h}}_i$ filter, $i = 1, 2, \ldots, N$.

5.5.2 Experimental Results

Let us prove the developed structures by running the simulations on two cascaded filters ($\hat{\mathbf{h}}_1, \hat{\mathbf{h}}_2$), i.e., $N = 2$, which is a cascaded bilinear form NLMS (NLMS-CBF) algorithm. In these simulations, we consider a system identification scenario, where the unknown system is characterized by the first echo path from G168 Recommendation [1] padded with zeroes, L = 512.

In Fig. 5.9 we analyze the tracking capability and the echo path changes at time 7 seconds. We can remark that for $\mu = \mu_{\hat{\mathbf{h}}_1} = \mu_{\hat{\mathbf{h}}_2} = 1$ the NLMS algorithm presents a slightly faster convergence rate than the NLMS-CBF algorithm when the echo path changes. However, their normalized misalignment level is the same. On the other hand, it seems that the step-sizes $\mu = \mu_1 = \mu_2 = 0.1$ are too small when the input signal is correlated. Therefore, an appropriate step-size must be selected.

5.6 Cascaded Multilinear RLS Algorithm

The RLS algorithm is popular due to its good performance in terms of convergence rate, normalized misalignment level, and tracking capability, being easy to parametrize and use in practical applications. However, its performance comes with the cost of computational complexity which is $O(L^2)$. Therefore, we aim to provide a solution that uses the benefit of the cascaded filters approach to reduce the computational complexity and maintain the good performance of the RLS algorithm, so that to obtain a cascaded multilinear form RLS (RLS-CMF) algorithm. In this development, we start by minimizing the LS cost functions for the cascaded system.

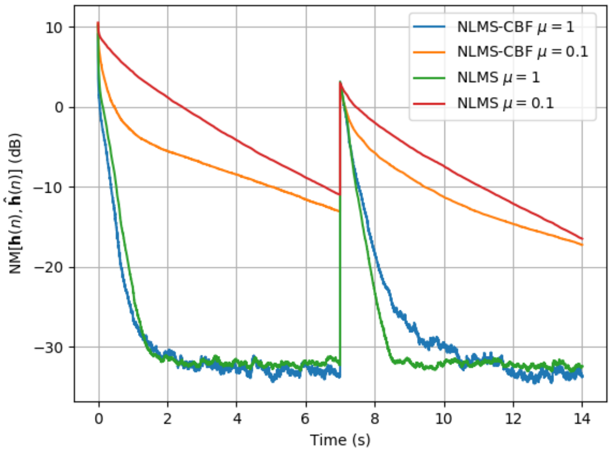

Fig. 5.9 Normalized misalignment of the NLMS-CBF and NLMS algorithms. The input signal is an AR(1) process, $\beta = 0.8$, $L = 512$, $L_1 = L_2 = L/2$, and SNR = 20 dB.

5.6.1 Algorithm Development

Following the same procedure as in Section 2.5, we can identify the update equations of the cascaded multilinear RLS algorithm as

$$\begin{aligned}
\hat{\mathbf{h}}_1(n) &= \hat{\mathbf{h}}_1(n-1) + \mathbf{s}_{\hat{\mathbf{h}}_2\hat{\mathbf{h}}_3\cdots\hat{\mathbf{h}}_N}(n) e_{\hat{\mathbf{h}}_2\hat{\mathbf{h}}_3\cdots\hat{\mathbf{h}}_N}(n), \\
\hat{\mathbf{h}}_2(n) &= \hat{\mathbf{h}}_2(n-1) + \mathbf{s}_{\hat{\mathbf{h}}_1\hat{\mathbf{h}}_3\cdots\hat{\mathbf{h}}_N}(n) e_{\hat{\mathbf{h}}_1\hat{\mathbf{h}}_3\cdots\hat{\mathbf{h}}_N}(n), \\
&\vdots \\
\hat{\mathbf{h}}_N(n) &= \hat{\mathbf{h}}_N(n-1) + \mathbf{s}_{\hat{\mathbf{h}}_1\hat{\mathbf{h}}_2\cdots\hat{\mathbf{h}}_{N-1}}(n) e_{\hat{\mathbf{h}}_1\hat{\mathbf{h}}_2\cdots\hat{\mathbf{h}}_{N-1}}(n),
\end{aligned} \tag{5.38}$$

where $\mathbf{s}_{\hat{\mathbf{h}}_2\hat{\mathbf{h}}_3\cdots\hat{\mathbf{h}}_N}(n), \mathbf{s}_{\hat{\mathbf{h}}_1\hat{\mathbf{h}}_3\cdots\hat{\mathbf{h}}_N}(n), \vdots, \mathbf{s}_{\hat{\mathbf{h}}_1\hat{\mathbf{h}}_2\cdots\hat{\mathbf{h}}_{N-1}}(n)$ denote the update terms.

5.6.2 Experimental Results

In this section we aim to compare the performance of the cascaded multilinear RLS algorithm with that of the RLS algorithm in two different contexts. In the first one we consider that the unknown system is characterized by the first echo path from the G168 Recommendation [1], while in the second context the echo path is obtained as Kronecker product between multiple impulse responses [23].

For the first context, let us choose three different degrees of multilinear forms: $N = 2$, the bilinear form (BF), $N = 3$, the trilinear form (TF), $N = 4$, the quadratic form (QF). Therefore, we denote the cascaded multilinear RLS algorithm for the chosen degrees as cascaded BF RLS (RLS-CBF) algorithm, cascaded TF RLS (RLS-CTF) algorithm, and cascaded QF RLS (RLS-CQF) algorithm.

Next, in Fig. 5.12 the input signal is an AR(1) process with $\beta = 0.85$ and the echo path changes in the middle of the simulation. When the echo path changes, the effect of the multilinear degree is much visible in terms of the normalized misalignment level.

Also, it seems that the RLS-CBF algorithm reaches the normalized misalignment level of the RLS algorithm at some point (e.g., after 5 seconds of simulation).

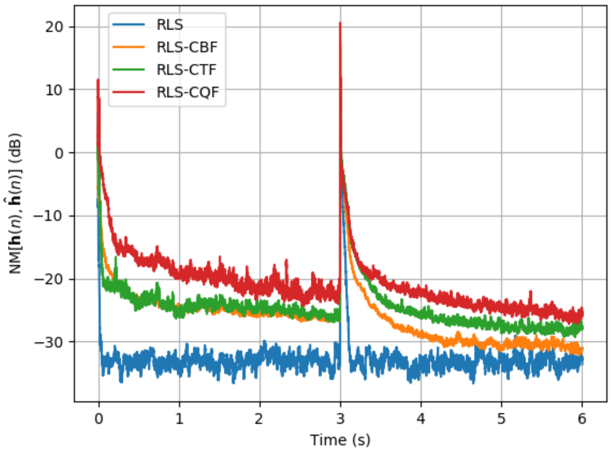

Fig. 5.12 Normalized misalignment of the RLS-CBF, RLS-CTF, RLS-CQF, and RLS algorithms. The input signal is an AR(1) process with $\beta = 0.85$ and SNR = 20 dB.

Further, we continue the experiments in the second context, where we choose two different multilinear degrees, $N = 2$ (bilinear) and $N = 3$ (trilinear). We denote the cascaded multilinear RLS algorithm based on the Kronecker product decomposition as RLS-CKD algorithm.

Let us begin with the simulations of the bilinear form, where the system to be identified is obtained through the Kronecker product between the first impulse response, i.e., $\mathbf{h}_1$, from the G168 Recommnedation [1] (with $L_1 = 64$) and an exponential impulse response, $\mathbf{h}_2$ (with $L_2 = 8$). The global impulse response is $\mathbf{h} = \mathbf{h}_2 \otimes \mathbf{h}_1$, with $L = L_1 L_2 = 512$ coefficients.

In Fig. 5.17 the input signal is a speech sequence and the echo path changes in the middle of the simulation. We can remark that the RLS algorithm achieves a normalized misalignment level of -20 dB after 7 seconds which is not desirable in a real-time application. On the other hand, the RLS-CKD algorithm achieves a normalized misalignment level of -40 dB in 2 seconds. This time the influence of the M parameter is visible from the beginning and impacts the converge rate of the RLS-CKD algorithm. However, we can outline the ease of the RLS-CKD algorithm to determine the new system when the impulse response changes.

Further, we analyze the performance of the RLS-CKD algorithm in the trilinear context. The impulse response to be determined is obtained as the Kronecker product between the first echo path from the G168 Recommendation [1], i.e., $\mathbf{h}_1$ (with $L_1 = 64$), a randomly generated impulse response $\mathbf{h}_2$ (with $L_2 = 8$), and an exponential impulse response $\mathbf{h}_3$ (with $L_3 = 4$). In brief, the global system to be identified is $\mathbf{h} = \mathbf{h}_3 \otimes \mathbf{h}_2 \otimes \mathbf{h}_1$, of $L = L_1 L_2 L_3 = 2048$ coefficients.

In Fig. 5.20 the input signal is an AR(1) process with $\beta = 0.9$ and the echo path is changing after 4 seconds. We can remark that the RLS-CKD algorithm outperforms the

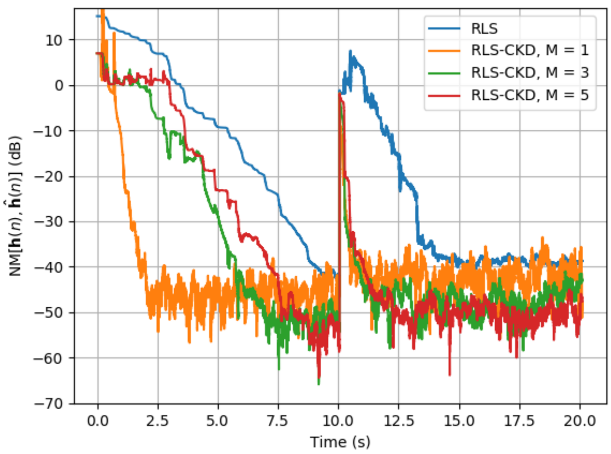

Fig. 5.17 Normalized misalignment of the RLS-CKD ($L_1 = 64$, $L_2 = 8$) and classical RLS ($L = 512$) algorithms. The input signal is a speech sequence, SNR = 30dB, and the impulse response changes after 10 seconds of simulation.

RLS algorithm in terms of converge rate, normalized misalignment level, and tracking capability. This time, the effect of the M parameter on the convergence rate of the RLS-CKD algorithm is visible from the beginning.

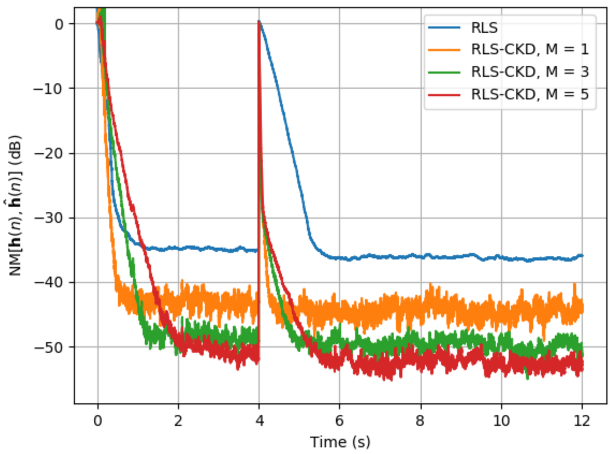

Fig. 5.20 Normalized misalignment of the RLS-CKD ($L_1 = 64$, $L_2 = 8$, $L_3 = 4$) and classical RLS ($L = 2048$) algorithms. The input signal is an AR(1) process with $\beta = 0.9$, SNR = 20dB, and the impulse response changes after 4 seconds of simulation.

5.7 Computational Complexity Analysis

The way of splitting a global system into a cascade of smaller systems (in terms of impulse response length), represents a huge advantage for the computational complexity reduction. This advantage is more important for those algorithms that present a quadratic or cubic computational complexity, making them appropriate for applications as indicated in Table 5.1.

Table 5.1 Algorithms Computational Complexity Reduction

Algorithm	Complexity	Operations Example
RLS	$O(L^2)$	$2048^2 = 4194304$
RLS-CBF	$\sum_{i=1}^{2} O(L_i^2)$	$2 \times 1024^2 = 2097152$
RLS-CTF	$\sum_{i=1}^{3} O(L_i^2)$	$3 \times 682^2 = 1395372$
RLS-CQF	$\sum_{i=1}^{4} O(L_i^2)$	$4 \times 512^2 = 1048576$

However, if the impulse response is decomposable through the Kronecker product, the echo path can be identified through a cascade of shorter length filters and the benefits in terms of computational complexity are outlined in Table 5.2.

Table 5.2 Algorithms Computational Complexity Reduction through KPD Approach

Algorithm	Complexity	Operations Example
RLS	$O(L^2)$	$2048^2 = 4194304$
RLS-CKD (bilinear)	$\sum_{i=1}^{2} O(L_i^2)$	$64^2 + 32^2 = 5120$
RLS-CKD (trilinear)	$\sum_{i=1}^{3} O(L_i^2)$	$64^2 + 8^2 + 4^2 = 4176$
RLS-CKD (quadratic)	$\sum_{i=1}^{4} O(L_i^2)$	$32^2 + 8^2 + 4^2 + 2^2 = 1108$

5.8 Summary and Conclusions

In this chapter, we have introduced a set of structures without and with memory, and developed a way for splitting a long length filter in a cascade of shorter length filters. We have proved the theory through experiments and outlined the performance through the NLMS-CBF, RLS-CBF, RLS-CTF, RLS-CQF algorithms. Also, we have presented an optimized way to build the input data, which was approached in the context of the echo paths described through the KPD. We have assesed a particular effect that can be met in acoustic environments, the reverberation effect, by describing the echo path through the KPD. We have outlined that the RLS-CKD algorithm is a good candidate to handle this kind of effect and achieves good performance in terms of converge rate, normalized misalignment level, and tracking capability. The simulations outlined that the proposed algorithm outperforms their counterparts versions (i.e., NLMS or RLS algorithms). Also, we have presented the benefits in terms of computational complexity for a rudimentary echo path and for an echo path that can be described through KPD.

Chapter 6

Conclusions

6.1 Obtained results

The system identification problem is one of the most interesting in the signal processing field, which raises many challenges nowadays. This thesis provides comprehensive research in the adaptive filtering field and introduces new algorithms for system identification, aiming at both linear and multilinear systems. However, for a better understanding of the context and to facilitate the connection with the theory, the obtained results were presented along with the conclusions at the end of each chapter. Therefore, there is no need to reiterate them here. However, the main contributions are listed in the next section, outlining the main achievements.

6.2 Original contributions

The original contributions are briefly listed below, specifying the original publication where the contribution was published. The mention $[i]$, where i is a number, denotes the listing position in the next section.

Chapter 3

- Development of an optimized LMS algorithm with a variable step-size, the LMSO algorithm [1];
- Development of an optimized LMS algorithm with a differential step-size, the ODSS-LMS algorithm [2];
- Development of the simplified versions for the LMSO and ODSS-LMS algorithms; the LMSO-W and ODSS-LMS-W algorithms, suitable for real-time applications [1], [2];
- Convergence and stability analysis of the proposed algorithms [1], [2];
- Computational complexity analysis [1], [2];

- Performance analysis of the proposed algorithms, compared to their counterparts or other recent solutions [1], [2].

Chapter 4

- Improvement of the previously proposed LMSO-W algorithm, by using the data-reuse technique, obtainining the LMSO-W-DR algorithm [4];
- Theory development to present the connection with APA [4];
- Performance analysis of the LMSO-W-DR [4];
- Development of the variable step-size NLMS (VSS-NLMS) algorithm [7];
- Development of the variable step-size APA (VSS-APA) [8];
- Convergence modes analysis for the VSS-NLMS algorithm and VSS-APA [7], [8];
- Experimental studies to outline the improvements by using the proposed approaches [7], [8].

Chapter 5

- Introduced structures without and with memory (MISO or SISO), that ca be described by a multilinear input-output form [3], [5];
- Starting from the weighted adder-filter structure, development of a way to split a long length filter into a cascade of shorter length filters, aiming both linear and multilinear system identification [3], [5];
- An optimized way to build the input data tensor for the identification of multilinear systems [6];
- Development of the cascaded multilinear NLMS and RLS algorithms [3];
- Theory validation through the cascaded bilinear form NLMS (NLMS-BF) algorithm [3];
- Performance analysis based on the multilinear order [5];
- Experimental study to showcase the identification of a system that is characterized by the reverberation effect using the proposed RLS-CKD algorithm [6].

6.3 List of original publications

In this section the list of original publications is presented with the corresponding citations from the References.

[1] A.-G. Rusu, S. Ciochină, and C. Paleologu, "On the Step-Size optimization of the LMS Algorithm", *42nd International Conference on Telecommunications and Signal Processing (TSP)*, pp. 168–173, 2019 [6] [ISI Proceedings];

[2] A.-G. Rusu, S. Ciochină, C. Paleologu, and J. Benesty, “An Optimized Differential Step-Size LMS Algorithm”, Algorithms, 12(8):147, 2019 [9] [ISSN: 1999-4893, **CiteScore - Q2 (Numerical Analysis)**];

[3] A.-G. Rusu and S. Ciochină, “Cascaded adaptive filters in a bilinear approach for system identification”, *International Symposium on Electronics and Telecommunications (ISETC)*, pp. 1–4, Nov. 2020 [2] [ISI Proceedings];

[4] A.-G. Rusu, L.-M. Dogariu, S. Ciochină, and C. Paleologu, “A Data-Reuse Approach for an Optimized LMS Algorithm”, *International Conference on Speech Technology and Human-Computer Dialogue (SpeD)*, pp. 31–35, 2021 [5] [ISI Proceedings];

[5] A.-G. Rusu, S. Ciochină, and C. Paleologu, “Cascaded Adaptive Filters in a Multilinear Approach for System Identification”, *International Symposium for Design and Technology in Electronic Packaging (SIITME)*, pp. 1–4, Oct. 2021 [7] [ISI Proceedings];

[6] A.-G. Rusu, S. Ciochină, C. Paleologu, and J. Benesty, “Cascaded RLS Adaptive Filters Based on a Kronecker Product Decomposition”, Electronics, 11(3):409, 2022 [8] [ISSN: 2079-9292, **Q3 - IF - 2.690**];

[7] A.-G. Rusu, C. Paleologu, J. Benesty, and S. Ciochină, “A Variable Step Size Normalized Least-Mean-Square Algorithm Based on Data Reuse”, Algorithms, 15(4):111, 2022 [3] [ISSN: 1999-4893, **CiteScore - Q2 (Numerical Analysis)**];

[8] A.-G. Rusu, L.-M. Dogariu, R.-L. Costea, C. Paleologu, J. Benesty, and S. Ciochină, “A Variable Step-Size Affine Projection Algorithm Based on Data Reuse”, *45th International Conference on Telecommunications and Signal Processing (TSP)*, 2022 [4] [ISI Proceedings].

6.4 Perspectives for further developments

Future works will focus on improving the work of this thesis in the context of linear and multilinear system identification. Definitely, one of the directions is the development of a state-of-the-art method to estimate the system’s noise and convert the proposed LMSO and ODSS-LMS algorithms into nonparametric algorithms. Therefore, the LMSO and ODSS-LMS algorithms could become general solutions for the identification of any kind of impulse response. Another path of research is the analysis of the connection between the LMSO-DR algorithm and APA, exploring new ways to handle the input data.

We aim to extend the way of splitting a long length filter into a cascade of shorter length filters, to analyze different types of non-linearities that can occur in real-world applications.

References

[1] (2002). *Digital Network Echo Cancellers, ITU-T Recommendations G.168*.

[2] A.-G. Rusu and S. Ciochină (Nov. 2020). Cascaded adaptive filters in a bilinear approach for system identification. In *International Symposium on Electronics and Telecommunications (ISETC)*, pages 1–4.

[3] A.-G. Rusu, C. Paleologu, J. Benesty, and S. Ciochină (2022). A Variable Step Size Normalized Least-Mean-Square Algorithm Based on Data Reuse. *Algorithms*, 15(4).

[4] A.-G. Rusu, L.-M. Dogariu, R.-L. Costea, C. Paleologu, J. Benesty, and S. Ciochină (2022). A Variable Step-Size Affine Projection Algorithm Based on Data Reuse. In *45th International Conference on Telecommunications and Signal Processing (TSP)*.

[5] A.-G. Rusu, L.-M. Dogariu, S. Ciochină, and C. Paleologu (2021). A Data-Reuse Approach for an Optimized LMS Algorithm. In *International Conference on Speech Technology and Human-Computer Dialogue (SpeD)*, pages 31–36.

[6] A.-G. Rusu, S. Ciochină, and C. Paleologu (2019). On the Step-Size optimization of the LMS Algorithm. In *42nd International Conference on Telecommunications and Signal Processing (TSP)*, pages 168–173.

[7] A.-G. Rusu, S. Ciochină, and C. Paleologu (Oct. 2021). Cascaded Adaptive Filters in a Multilinear Approach for System Identification. In *International Symposium for Design and Technology in Electronic Packaging (SIITME)*, pages 1–4.

[8] A.-G. Rusu, S. Ciochină, C. Paleologu, and J. Benesty (2022). Cascaded RLS Adaptive Filters Based on a Kronecker Product Decomposition. *Electronics*, 11(3).

[9] A.-G. Rusu, S. Ciochină, C. Paleologu, Constantin, and J. Benesty (2019). An Optimized Differential Step-Size LMS Algorithm. *Algorithms*, 12(8).

[10] B.A. Schnaufer and W.K. Jenkins (1993). New data-reusing LMS algorithms for improved convergence. In *Proceedings of 27th Asilomar Conference on Signals, Systems and Computers*, volume 2, pages 1584–1588.

[11] C. Elisei-Iliescu, C. Paleologu, J. Benesty, C. Stanciu, C. Anghel, and S. Ciochină (May 2019). Recursive least-squares algorithms for the identification of low-rank systems. *IEEE/ACM Trans. Audio, Speech, Language Processing*, 27:903–918.

[12] C. Paleologu and J. Benesty (Aug. 2012). A practical data-reuse adaptive algorithm for acoustic echo cancellation. *Proc. 20th European Signal Processing Conference*, pages 31–34.

[13] C. Paleologu, S. Ciochină, and J. Benesty (2008). Variable step-size NLMS algorithm for under-modeling acoustic echo cancellation. *IEEE Signal Processing Lett.*, 15:5–8.

[14] C. Paleologu, S. Ciochină, J. Benesty, and S. L. Grant (2015). An overview on optimized NLMS algorithms for acoustic echo cancellation. *EURASIP Journal Advances Signal Processing*.

[15] D. L. Duttweiler (Sept. 2000). Proportionate normalized least-mean-squares adaptation in echo cancelers. *IEEE Trans. Speech, Audio Processing*, 8:508–518.

[16] E.-W. Bai, D. Li (Nov. 2004). Convergence of the iterative Hammerstein system identification algorithm. *IEEE Transactions on Automatic Control*, 49(11):1929–1940.

[17] G. Enzner, H. Buchner, A. Favrot, and F. Kuech (2014). Acoustic echo control. In *Academic Press Library in Signal Processing: Image, Video Processing and Analysis, Hardware, Audio, Acoustic and Speech Processing*, volume 4, chapter 30, pages 807–877. Academic Press.

[18] H. Deng and M. Doroslovački (May. 2006). Proportionate adaptive algorithms for network echo cancellation. *IEEE Trans. Signal Processing*, 54:1794–1803.

[19] J. A. Apolinário, Jr., M. L. R. de Campos, and P. S. R. Diniz (Nov. 2000). Convergence analysis of the binormalized data-reusing LMS algorithm. *IEEE Trans. Signal Processing*, 48:3235–3242.

[20] J. Benesty and S. L. Gay (2002). An improved PNLMS algorithm. *Proc. IEEE ICASSP*, pages II–1881–II–1884.

[21] J. Benesty and T. Gänsler (2003). On data-reuse adaptive algorithms. In *Proc. IWAENC*, pages 31–34.

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

[23] L.-M. Dogariu, C. L. Stanciu, C. Elisei-Iliescu, C. Paleologu, J. Benesty, and S. Ciochină (Apr. 2021). Tensor-based adaptive filtering algorithms. *Symmetry*, 13.

[24] L. M. Dogariu, C. Paleologu, J. Benesty, and S. Ciochină (Jan. 2020). An efficient Kalman filter for the identification of low-rank systems. *Signal Processing*, 166.

[25] M. A. Iqbal and S. L. Grant (2008). Novel variable step size NLMS algorithms for echo cancellation. In *Proc. IEEE ICASSP*, pages 241–244.

[26] P. S. R. Diniz and S. Werner (Jan. 2003). Set-membership binormalized data-reusing LMS algorithms. *IEEE Trans. Signal Processing*, 51(1):124–134.

[27] S. Ciochină, C. Paleologu, and J. Benesty (2016). An optimized NLMS algorithm for system identification. *Signal Processing*, 118:115–121.

[28] S.-H. Leung and C. F. So (Aug. 2005). Gradient-based variable forgetting factor RLS algorithm in time-varying environments. *IEEE Trans. Signal Processing*, 53:3141–3150.

[29] S. L. Gay (1998). An efficient, fast converging adaptive filter for network echo cancellation. *Proc. IEEE Asilomar Conference on Signals, Systems, and Computers*, pages 394–398.