THIAGO YUJI AOYAGI

ON FILTER ORDER IN FIR ADAPTIVE FILTERS

São Paulo
2022

THIAGO YUJI AOYAGI

ON FILTER ORDER IN FIR ADAPTIVE FILTERS

Versão Corrigida

Dissertação apresentada à Escola Politécnica da Universidade de São Paulo para obtenção do Título de Mestre em Ciências.

São Paulo
2022

THIAGO YUJI AOYAGI

ON FILTER ORDER IN FIR ADAPTIVE FILTERS

Versão Corrigida

Dissertação apresentada à Escola Politécnica da Universidade de São Paulo para obtenção do Título de Mestre em Ciências.

Área de Concentração:
Sistemas Eletrônicos

Orientador:
Cássio Guimarães Lopes

Coorientador:
Vítor Heloiz Nascimento

São Paulo
2022

Este exemplar foi revisado e corrigido em relação à versão original, sob responsabilidade única do autor e com a anuência de seu orientador.

São Paulo, 07 de Fevereiro de 2022

Assinatura do autor: ______________________

Assinatura do orientador: ______________________

Catalogação-na-publicação

Aoyagi, Thiago Yuji
On filter order in FIR adaptive filters / T. Y. Aoyagi -- versão corr. -- São Paulo, 2022.
106 p.

Dissertação (Mestrado) - Escola Politécnica da Universidade de São Paulo. Departamento de Engenharia de Sistemas Eletrônicos.

1.Processamento Digital de Sinais 2.Processamento de Sinais Adaptativos I.Universidade de São Paulo. Escola Politécnica. Departamento de Engenharia de Sistemas Eletrônicos II.t.

“Our noblest impulse, the impulse to know and understand, makes it our duty to search. And even a false theory, if only it was found through genuine search, is for that reason superior to the complacent certainty of those who reject it because they presume to know — to know, although they themselves have not searched!”

-Arnold Schoenberg-

ACKNOWLEDGMENTS

I am very grateful to all the opportunities I had since I joined the Signal Processing Lab (LPS). I guess it all began with Vitor's undergraduate course, which drew my atention to the signal processing field. Then, Cássio entered into scene as I started master's, and kept me motivated and exerted major influence in this work. Each has taught me and guided me not only with technical advises, but also sharing their views and experiences in the academy. I ended up causing some mess until I managed to make our distinct ways work together, and I take it as part of the learning.

I would also like to thank my fellows from LPS Carlos and Paulo, who had been daily companion before the pandemic hit, and had made my days in the lab funnier with either casual or technical discussions.

RESUMO

Um problema encontrado na prática ao projetar um filtro adaptativo FIR (do inglês, *finite impulse response*) é escolher um comprimento adequado para o filtro. O comprimento ideal para o filtro depende da aplicação, por isso é comum determiná-lo por métodos simples e práticos, como por tentativa e erro. Com um número pequeno de coeficientes, o filtro tem menor complexidade e pode se beneficiar de uma maior taxa de convergência, mas seu desempenho em regime é afetado por submodelamento. Por outro lado, com muitos coeficientes, asseguramos um baixo ou inexistente efeito de submodelamento, ao custo de limitar a máxima taxa de convergência estável do filtro, aumentar a complexidade computacional e também reduzir a capacidade do filtro em acompanhar variações no tempo. Neste trabalho, analisamos como o comprimento do filtro afeta o desempenho de algoritmos adaptativos, em particular dos algoritmos LMS e ϵ-NLMS. Para ambientes estacionários, analisamos o desempenho tanto em transiente quanto em regime, e propomos um projeto para o comprimento do filtro que garante alta convergência e baixo submodelamento, assumindo que a resposta impulsiva do meio segue uma envoltória de decaimento exponencial. Mostramos como um filtro com o comprimento proposto é particularmente interessante para ser usado em uma combinação de filtros, operando como o filtro rápido. Para ambientes não-estacionários, focamos no estudo do desempenho em regime, e mostramos com simulações que um filtro curto pode superar o desempenho de um filtro mais longo tanto em convergência quanto ao rastrear as variações temporais do sistema.

Palavras-Chave – Filtros adaptativos, comprimento do filtro, submodelamento, escolha do comprimento do filtro, combinação de filtros

ABSTRACT

A practical problem faced when designing an FIR (Finite Impulse Response) adaptive filter is to set an appropriate filter length. The best choice for the length is application-dependent, and is common practice to determine it by some rough approximation, such as by trial-and-error. By setting a small number of coefficients, the filter has a reduced complexity and may benefit from an increased convergence rate, but its steady-state performance is degraded by undermodeling. By setting a large number of coefficients, we ensure the filter suffers negligible or no undermodeling effects, but we limit the maximum stable convergence rate, increase the computational complexity and also decrease the filter ability to respond in nonstationary scenarios. In this work, we analyze how the filter length affects the performance of adaptive algorithms, in particular, for the LMS and the ϵ-NLMS algorithms. For stationary scenarios, we analyze both transient and steady-state performance, and propose a method for selecting the filter length that ensures fast convergence rate and low undermodeling effects, assuming that the system impulse response follows an exponential decay envelope. We show that a filter with the proposed length is particularly interesting to operate as the fast filter within a combination of filters. For nonstationary scenarios, we focus our study on the steady-state performance, and show through simulations that a short filter may outperform a longer one in both convergence and tracking performance.

Keywords – Adaptive filtering, filter length, undermodeling, filter length selection, combination of filters

LIST OF FIGURES

CONTENTS

1 INTRODUCTION

A practical problem when designing an FIR (Finite Impulse Response) adaptive filter is how to choose the filter order — the number of tap weights, also called the filter length. The adaptive filter must have enough taps to be able to model the impulse response of the target system (or of the inverse system, depending on the application) so that the overall estimation error is satisfactorily low. As the impulse response is unknown by definition in adaptive filtering, the filter order is often adjusted by trial-and-error.

Such a loose choice for the filter length, however, may yield an insufficient or excessive number of taps, affecting negatively the filter performance [1–4]. If we set an insufficient length for the filter, the estimation will be biased, which is also known as undermodeling [5, 6]. In recursive estimation, as is the case of adaptive filtering, this reflects mainly on a degradation of the steady-state performance. If we set a large length, even though we may benefit from less bias since many real systems have very long or infinite impulse responses, we may encounter implementation problems due to the processing limitations of the computational device, and the increased variance due to the estimation of a larger number of parameters.

In spite of the increased estimation error of an undermodeled filter, there are some advantages that are worth considering: by reducing the number of taps, we reduce the algorithm complexity and may reduce the mean square error since the increased bias may be offset by reduced variance. In some cases, we can increase the convergence speed, as some analyses of undermodeled algorithms show [7–10].

Therefore, the problem of selecting the filter length faces the compromise between choosing a large length that ensures a low steady-state error level, but that limits the convergence rate, and choosing a small length to take advantage of lower complexity and of a possible increase of convergence rate, at the cost of an increased steady-state error. This dilemma is somewhat similar to the well-known trade-off between convergence rate and steady-state error found, for example, when tuning the step size of the LMS algorithm, or when tuning the forgetting factor of the RLS algorithm. When the environment is stationary, a large length usually guarantees any desired steady-state performance level since we can adjust other filter parameters. With the LMS algorithm, for example, while a large length ensures low estimation bias, we can reduce undefinitely the step size so as to reduce the estimation variance. In time-variant environments, on the other hand, it might not be possible to improve the steady-state performance by reducing the step size because small step sizes might make the filter unable to track the system behavior.

In the literature, the problem of choosing the length of a fixed length adaptive filter has not drawn much attention as, for example, the question of tuning the step size or designing time-varying step sizes for LMS-like algorithms. Some works [11] investigate the choice of the filter length for specific applications. The classical analyses of adaptive algorithms [1–3, 12] consider filters with the same length as the optimal weight vector (the actual impulse response), which hinders the treatment of the filter length as a free variable. On the other hand, there are many works that propose techniques to adaptively select the filter length, yielding variable-length algorithms [8, 13–20] that can outperform fixed length algorithms in both transient and steady-state performance. Some of these algorithms are also capable of adapting to changes in the length of the impulse response. Nevertheless, they are intrinsically more complex algorithms, and most of them encounter implementation difficulties. For example, [13] and [14] encounter difficulties to set the instants to increment the filter taps; [8] has increased computational complexity to compute the length at each iteration; and [15] and [16] are too sensitive to parameter choice.

Another techniques in the literature widely used to select the model order are the information criterion methods, such as the AIC (Akaike Information Criterion) and BIC (Bayesian Information Criterion) [21,22]. Assuming a probabilistic model for the observed data, these methods aim to select the model order that maximizes the relative Kullback-Leibler information. In [23], the AIC was employed to select the filter order of adaptive filters. However, as based on maximum likelihood estimation, information criterion methods do not consider the excess mean-square error intrinsic to stochastic algorithms, as the ones used in adaptive filtering.

1.1 Objectives

In this work, we focus on analyzing the effects of the filter length on the performance of a fixed length adaptive filter, in particular, for the LMS and the ϵ-NLMS algorithms. We derive the performance analyses considering the filter length independent from the length of the impulse response, as done in analyses of undermodeled filters [8–10]. We derive these analyses for time-invariant systems (stationary environments) and for time-variant systems (often refered to as *nonstationary environments* in the literature).

Based on such analyses, for the stationary case, we propose a filter length design method for the LMS/ϵ-NLMS algorithm by pondering the transient and steady-state performance. Specifically, the proposed method allows a small degradation of the steady-state performance due to undermodeling in order to take advantage of the faster convergence rate and the lower complexity of shorter filters. The proposed method assumes that the underlying impulse response has an exponential decay envelope. Although this seems very restrictive, such models for impulse response are found in many practical applications of adaptive filtering, such as acoustics and communications [8, 18, 24–27].

A filter designed with the proposed length, which enjoys fast convergence rate, is an ideal candidate to operate within a combination of filters. Combinations of filters take independent adaptive filters with distinct properties and combine them so that the overall output can perform at least as well as the best component filter [28–32]. A common example is combining one filter with fast convergence rate, though with high steady-state error, as the proposed filter, with a slow filter that has a low level of steady-state error. We propose the fast filter in the combination to be designed with the proposed length, so that the combination benefits from both complexity reduction and fast convergence rate.

1.2 Contributions

This work contributes with the following original developments:

1. Extension of the optimal step size for the fastest convergence rate based on the initial difference [33], by including undermodeling effects (Subsection 3.2.1);

2. A method to select the filter length for the LMS and ϵ-NLMS algorithms based on the pondering between the transient and steady-state performance (Section 3.3);

3. Combination of filters with an undermodeled fast filter (Section 3.4);

4. Extension of the cycle length design for the weight feedback scheme [34], by considering the whole mean-square error, instead of only the minimum error, and by including undermodeled filters (Subsection 3.4.1);

5. Tracking performance of the undermodeled LMS and ϵ-NLMS algorithms (Section 4.2);

6. Derivation of the optimal step size (minimizing steady-state mean-square error) for the nonstationary and undermodeled case, for white input (Section 4.3).

1.3 Publications

This work resulted in the publication of the following paper:

- T. Y. Aoyagi, A. C. Ferreira, C. G. Lopes, and V. H. Nascimento, “Exploring undermodeling in combinations of adaptive filters”, in *2021 IEEE Statistical Signal Processing Workshop*, 2021, pp. 66-70.

1.4 Structure of this work

In Chapter 2, we introduce the state-of-the-art of adaptive filtering that is relevant to this text. We introduce the system identification framework and the assumptions about the environment under which we derive our analyses; we present the linear estimation problem that underlies the operation of adaptive filters; we present the adaptive algorithms that are relevant to this text, such as the LMS, the ϵ-NLMS and the combinations of filters; and we present how we evaluate the performance of adaptive algorithms.

In Chapter 3, we treat the development for the stationary case. We begin with the analysis of both transient and steady-state performance for the LMS and ϵ-NLMS algorithms, considering the length of the filter independent from the length of the impulse response (as we will see, the case of undermodeled filters is general enough to comprise any filter length case); we derive the step size that maximizes convergence rate; we propose a method to choose the filter length so that the filter benefits from fast convegence rate and low undermodeling; we show how a combination of filters can benefit from a fast filter designed with the proposed length; and finally we design of the cycle length of the weight feedback scheme, including the effects of undermodeling and of the excess mean-square error.

In Chapter 4, we analyze the tracking performance of undermodeled LMS and ϵ-NLMS algorithms in nonstationary environment, employing the energy conservation relation; we then derive the step size that minimizes the steady-state error, considering white input signal; in simulations, we show that an undermodeled filter can benefit from both faster convergence rate and lower steady-state error.

Finally, we conclude this text in Chapter 5 discussing the ideas in this text that can be further developed in future works.

1.5 Notation

We introduce here the notation we use throughout this work, which is based on [35].

Random variables are denoted by boldface, as $\boldsymbol{x}$, and deterministic variables are denote by plainface, as x. Deterministic variables include observations of random variables, mostly used when describing update rules of an algorithm or relations based on observed or measured quantities. The notation of boldface, on the other hand, are employed in the context of analyzing statistical behavior.

Scalars and vectors are denoted by lowercase letters, while matrices are denoted by uppercase. Uppercase letters also denote the natural integers for the size of arrays (vectors and matrices). When iteration-dependent, scalars are indexed in parenthesis as $x(i)$, and vectors and matrices are indexed in subscript as, respectively, x_i and X_i. Note that, when iteration-dependent, we can distinguish scalars and vectors by the way they are indexed. All vectors are column vectors, except the regressor vector u_i (defined around (2.1)), which is taken as row vector for convenience of notation. The k-th element of a vector x is denoted by $x(k)$, and the $\{k, \ell\}$-th element of a matrix X is denoted as $X(k, \ell)$.

In this work, we deal with arrays of different lengths, so it is important to identify their size in the notation. We identify a vector length by a subscript, for example, x_M is a vector of length M. For a matrix of size $M \times N$, we write $X_{M,N}$, and for an $M \times M$ square matrix, we write X_M.

The variance of a zero-mean random variable $\boldsymbol{x}$ or of a zero-mean stationary random process $\boldsymbol{x}(i)$ is denoted by $\sigma_x^2 = E\boldsymbol{x}^2$. The autocorrelation matrix of a random vector $\boldsymbol{x}_M$ is denoted as $R_{x,M} = E\boldsymbol{x}_M\boldsymbol{x}_M^{\mathrm{T}}$. The cross-correlation between $\boldsymbol{x}_M$ and $\boldsymbol{y}_N$ is denoted as $R_{xy,M,N} = E\boldsymbol{x}_M\boldsymbol{y}_N^{\mathrm{T}}$.

We list in the following the notation of some mathematical operations we use:

- $||x|| = \sqrt{x^{\mathrm{T}}x}$ is the Euclidean norm of a vector x;
- $\lceil x \rceil$ is the ceiling function of x;
- $\sum x_M$, when indexes are not explicit, sums over all elements of the array (vector or matrix);

- $X_{M,N} \odot Y_{M,N}$ is the Hadamard matrix product between $X_{M,N}$ and $Y_{M,N}$, that is, their element-wise product.

2 ADAPTIVE FILTERING AND ADAPTIVE ALGORITHMS

An adaptive filter is a digital filter whose parameters are recursively adapted in order to perform a desired task. It is more powerful than a simple digital filter of fixed weights to extract desired information from observed signals since, as an adaptive and time-varying device, it is capable of tracking the behavior of time-variant environments and hence it finds application in a large variety of fields, such as communication, radar, biomedical engineering, control, acoustics, among others [1–4, 35].

The adaptive filter works by extracting the correlation between two signals. One signal is taken as the filter input, while the other, the desired signal, usually sensed in real-time from the same environment of the input signal, is taken to supervise the filtering operation in order to realize the desired task. The error between the desired signal and the filter output, which measures the filtering performance, is then used in an adaptive algorithm, the core mechanism in an adaptive filter, to adapt iteratively the filter weights.

Let us illustrate with some examples. In a system identification problem [36, 37], we wish the adaptive filter to replicate the behavior of a target system. By observing the input and the output of the system, we feed the filter with the same input and we want to minimize the error between the filter output and the observed output (the desired signal).

Another classical application of adaptive filtering is acoustic echo cancellation [24, 25, 38]. An active echo canceller is required whenever a loudspeaker and a microphone are placed such that the microphone picks up the signal radiated by the loudspeaker and its reflections at the borders of the enclosure, causing a usually annoying effect that depends on the application. In a hands-free telephony, for example, this annoying effect would be the interlocutor to hear a delayed and attenuated version of his own speech. An adaptive filter can be used to prevent this by reproducing a synthesized version of the echo, and by subtracting it from the signal received by the microphone, the echo would be cancelled in a proper operation. In this problem, the loudspeaker signal is input to the adaptive filter, while the microphone signal, composed by the echo signal superposed to the local speaker voice, is taken as the desired signal.

In this chapter, we introduce the fundamental definitions and relations in the field of adaptive filtering that are relevant to this text. In Section 2.1, we introduce the system identification framework, which describes the class of adaptive filtering problems we deal with. In Section 2.2, we introduce some mathematical assumptions for the environment and signals to make the analysis of adaptive filtering tractable. In Section 2.3, we present the estimation problem that underlies the operation of an adaptive filter and, then, in Section 2.4, we define the adaptive algorithms with which we work throughout this text and show how they are related to this estimation problem. In Section 2.5, we present combinations of filters, which are schemes used to improve the adaptive filtering performance by combining multiple adaptive filters of different characteristics. Finally, in Section 2.6, we show how we evaluate the performance of an adaptive filter.

2.1 The system identification framework

The first step to implement an adaptive filter is to determine what are the input and the desired signals in the application. Traditionally, there are four classes of applications of adaptive filtering that determines its configuration: system identification, inverse identification, interference cancelling and prediction [1–3].

Throughout this work, we consider the system identification framework, in which the desired signal is the output of the unknown system, possibly corrupted by some measurement noise, and the input of the adaptive filter is the same excitation input of the system. This setup is usually adopted in the literature for algorithm analysis due to the facility it provides to evaluate the algorithm performance [35, 39, 40]. It comprises applications such as channel estimation [41,42], time delay estimation [43–45] and adaptive model control [1, 46].

Let us describe this formally. Consider an unknown system fed by a discrete-time input signal $\boldsymbol{u}(i)$, where $i \in \mathbb{Z}$. In a general system identification setup, the output is expressed as [2, 4]

$$\boldsymbol{d}(i) = \mathcal{H}\left(\boldsymbol{u}_{P,i}\right) + \boldsymbol{v}(i), \tag{2.1}$$

where $\mathcal{H}(\cdot)$ is an unknown function, $\boldsymbol{u}_{P,i} = \begin{bmatrix}\boldsymbol{u}(i) & \boldsymbol{u}(i-1) & \dots & \boldsymbol{u}(i-P+1)\end{bmatrix} \in \mathbb{R}^{1\times P}$ is the input regressor vector, and $\boldsymbol{v}(i)$ is an additive measurement noise, uncorrelated with $\boldsymbol{u}(i)$. The term $\mathcal{H}\left(\boldsymbol{u}_{P,i}\right)$ indicates that the output depends only on the lastest P samples of $\boldsymbol{u}(i)$.

We wish the output of the adaptive filter to be as close as possible to the system output, and so we arrange the filter as depicted in Figure 1. Considering the filter with M adaptive weights, and denoting the filter weight vector as $w_{M,i} \in \mathbb{R}^{M\times 1}$, the filter output is given by

$$\boldsymbol{y}(i) = \boldsymbol{u}_{M,i} w_{M,i-1}, \tag{2.2}$$

where $\boldsymbol{u}_{M,i} = \begin{bmatrix}\boldsymbol{u}(i) & \boldsymbol{u}(i-1) & \ldots & \boldsymbol{u}(i-M+1)\end{bmatrix} \in \mathbb{R}^{1\times M}$. The error between the desired signal and the filter output is

$$\boldsymbol{e}(i) \;=\; \boldsymbol{d}(i) - \boldsymbol{y}(i) \;=\; \boldsymbol{d}(i) - \boldsymbol{u}_{M,i} w_{M,i-1}. \tag{2.3}$$

Most analyses in the literature consider that the length of the filter matches the system order $M = P$. In this work, however, we are interested in the general case in which the filter might undermodel the system ($M < P$) or overmodel the system ($M > P$).

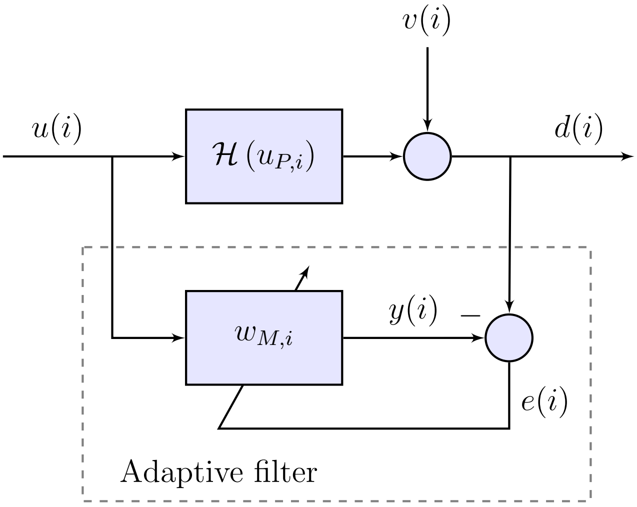


Figure 1: System identification framework for adaptive filters.

2.2 Modeling the environment

The model above describes a quite general scenario in which an adaptive filter is able to operate. However, to continue the analysis in this and in the following chapters, we must make some simplifying assumptions about the input signal and the system output signal to make theoretical derivations tractable.

2.2.1 Input signal

We assume that the input signal $\boldsymbol{u}(i)$ is a zero-mean wide-sense stationary (WSS) process [2, Ch.2] [3, Ch.2]. This implies that its mean function is constant $E\boldsymbol{u}(i) = 0$ and that its autocorrelation function depends only on the difference k between two time instants $r_u(k) = E\boldsymbol{u}(i)\boldsymbol{u}(i-k)$. Another implication is that the autocorrelation matrix, say of size P, $R_{u,P} = E\boldsymbol{u}_{P,i}^{\mathrm{T}}\boldsymbol{u}_{P,i}$, is a Toeplitz matrix and time-invariant. All the entries in the main diagonal of $R_{u,P}$ have the value of the input variance $\sigma_u^2 = E\boldsymbol{u}^2(i)$.

In this text, when simulating correlated input signals, we take $\boldsymbol{u}(i)$ as the first-order autoregressive (AR(1)) process [2, Ch.2] [3, Ch.2]

$$\boldsymbol{u}(i) = \rho\boldsymbol{u}(i-1) + \sqrt{\sigma_u^2(1-\rho^2)}\boldsymbol{n}(i), \tag{2.4}$$

where $|\rho| < 1$ and $\boldsymbol{n}(i)$ is a zero-mean i.i.d. sequence of unit variance. Under this model, it is very practical to generate the input signal and to quantify its autocorrelation. If $\rho = 0$, $\boldsymbol{u}(i)$ is white ($\boldsymbol{u}(i) = \boldsymbol{n}(i)$). Otherwise, $\boldsymbol{u}(i)$ is correlated, and the autocorrelation is stronger as $|\rho|$ is closer to one.

2.2.2 System output

We assume that the system output (2.1) is a linear function of the input as [4, 35, 40]

$$\boldsymbol{d}(i) = \boldsymbol{u}_{P,i}\boldsymbol{h}_{P,i} + \boldsymbol{v}(i), \tag{2.5}$$

where $\boldsymbol{h}_{P,i} = \begin{bmatrix}\boldsymbol{h}(0) & \boldsymbol{h}(1) & \cdots & \boldsymbol{h}(P-1)\end{bmatrix}^{\mathrm{T}} \in \mathbb{R}^{P\times 1}$ is the impulse response of the system. We also assume that the noise $\boldsymbol{v}(i)$ is a zero-mean i.i.d. process of variance σ_v^2.

We model the system impulse response by the Markov model [35, Ch.7] [47]

$$\begin{cases}\boldsymbol{h}_{P,i} &= h_P + \boldsymbol{\theta}_{P,i}\\ \boldsymbol{\theta}_{P,i} &= c\boldsymbol{\theta}_{P,i-1} + \boldsymbol{q}_{P,i}\end{cases}, \tag{2.6}$$

where $h_P \in \mathbb{R}^{P\times 1}$ is the deterministic counterpart, $\boldsymbol{\theta}_{P,i} \in \mathbb{R}^{P\times 1}$ is the internal state, c is a real scalar where $|c| < 1$, and $\boldsymbol{q}_{P,i} \in \mathbb{R}^{P\times 1}$ is a zero-mean i.i.d. sequence with mean vector and covariance matrix, respectively,

$$E\boldsymbol{q}_{P,i} = 0_P, \quad E\boldsymbol{q}_{P,i}\boldsymbol{q}_{P,i}^{\mathrm{T}} = Q_P. \tag{2.7}$$

Note that each entry $\boldsymbol{\theta}_{P,i}(k)$, for $k = 0, 1, \ldots, P-1$, is a first-order Markov process, and each entry of $\boldsymbol{h}_{P,i}(k)$ is one of these Markov processes added to a deterministic value $h_P(k)$.

We assume that the Markov process begins at some past instant such that, when the adaptive filter begins to run, at $i = 0$, the mean and mean-square behavior of the model (2.6) have already reached steady-state. As $\boldsymbol{q}_{P,i}$ is zero-mean, we are effectively assuming that $E\boldsymbol{\theta}_{P,i} = 0_P$ and $E\boldsymbol{h}_{P,i} = h_P$ for any instant $i \geq 0$.

Along this text, we often use a particular case of the model (2.6) in which the component $\boldsymbol{\theta}_{P,i}$ is identically zero (case in which the covariance matrix Q_P is identically zero), for any i, resulting the deterministic and time-invariant impulse response $\boldsymbol{h}_{P,i} = h_P$. We call this case as the *stationary* scenario since the system output is a WSS process under this model [48, Ch.10]. Otherwise, we call the general case of the model (2.6) as the *nonstationary* scenario (time-variant impulse response).

Most nonstationary models for adaptive filtering in the literature [2–4, 35] consider zero-mean impulse response $h_P = 0_P$ (that is, $\boldsymbol{h}_{P,i} = \boldsymbol{\theta}_{P,i}$), which does not allow a fair evaluation of the transient of an adaptive algorithm since the filter weights are usually initialized at the origin. Some of the classical models further consider $c = 1$ [4,35], yielding a model known as the random-walk model. Although this model provides some algebraic simplifications for algorithm analysis, it is not necessarily meaningful in practice, since the covariance matrix of $\boldsymbol{h}_{P,i}$ grows unboundedly as $i \to \infty$. This unstable behavior makes the random-walk model unsuitable for analyzing undermodeled algorithms because, as we will see in Sections 2.3 and 4.1, undermodeling effects depend on the magnitude and distribution of the weights along the impulse response, and that is why we exclude the case $|c| = 1$ in this text.

2.3 Linear optimal estimation

The objective in adaptive filtering is to minimize the error $\boldsymbol{e}(i)$ between the filter output $\boldsymbol{y}(i)$, which is a linear function of the input signal (see (2.2)), and the observed output $\boldsymbol{d}(i)$. Equivalently, we can also say that we want the linear model $\boldsymbol{y}(i)$ to estimate $\boldsymbol{d}(i)$ from the input regressor $\boldsymbol{u}_{M,i}$ [35, Ch.2]. Implicitly, we estimate the parameters $w_{M,i}$ of this linear model.

In this section, we address the theoretical issues of this estimation problem, such as the cost function and the optimal solution. The general derivations below assume that the input and the output signals are jointly WSS[1] [1, 2, 49], and so the resulting solution is time-invariant, so we drop the iteration index in the notation for convenience (in fact, this estimation problem can actually be written as a simple linear estimation problem, as shown in [35], where the input and the output are regarded as random variables rather than random processes).

Mathematically speaking, we can pose the problem above as the task of finding the filter weights w_M that minimizes the mean-square error (MSE) function

$$J(w_M) = E\boldsymbol{e}^2 = E(\boldsymbol{d} - \boldsymbol{y})^2 = E(\boldsymbol{d} - \boldsymbol{u}_M w_M)^2. \tag{2.8}$$

Define the input autocorrelation matrix $R_{u,M} \triangleq E\boldsymbol{u}_M^{\mathrm{T}}\boldsymbol{u}_M$, the cross-correlation vector $R_{du,M} \triangleq E\boldsymbol{d}\boldsymbol{u}_M^{\mathrm{T}}$ and the variance $\sigma_d^2 = E\boldsymbol{d}^2$. It can be shown that the optimal solution that minimizes (2.8) is [35]

$$w_M^o = R_{u,M}^{-1} R_{du,M}, \tag{2.9}$$

and the cost associated to it, the minimum MSE attainable, is

$$J_{\min} = J(w_M^o) = \sigma_d^2 - R_{du,M}^{\mathrm{T}} R_{u,M}^{-1} R_{du,M}. \tag{2.10}$$

Now, consider that the system is modeled by the linear relation (2.5), with stationary impulse response $\boldsymbol{h}_{P,i} = h_p$. Assume, at a first moment, that the filter is undermodeled, i.e., that $M < P$. Define $N = P - M$ and the following partitions

$$h_P = \begin{bmatrix} h_M \\ \bar{h}_N \end{bmatrix}, \quad \boldsymbol{u}_P = \begin{bmatrix} \boldsymbol{u}_M & \bar{\boldsymbol{u}}_N \end{bmatrix}, \tag{2.11}$$

where $h_M = \begin{bmatrix} h(0) & \dots & h(M-1) \end{bmatrix}^{\mathrm{T}}$, $\bar{h}_N = \begin{bmatrix} h(M) & \dots & h(P-1) \end{bmatrix}^{\mathrm{T}}$ and $\bar{\boldsymbol{u}}_N = \begin{bmatrix} \boldsymbol{u}(i-M) & \dots & \boldsymbol{u}(i-P+1) \end{bmatrix}$. Consequently, the autocorrelation matrix $R_{u,P}$ can be partioned as

$$R_{u,P} = \begin{bmatrix} R_{u,M} & R_{u,M,N} \\ R_{u,M,N}^{\mathrm{T}} & R_{u,N} \end{bmatrix}. \tag{2.12}$$

Thus, the moments in (2.10) that depend on $\boldsymbol{d}$ become

$$\sigma_d^2 = E(\boldsymbol{u}_P h_P + \boldsymbol{v})^{\mathrm{T}}(\boldsymbol{u}_P h_P + \boldsymbol{v}) = \sigma_v^2 + h_P^{\mathrm{T}} R_{u,P} h_P, \tag{2.13}$$

[1] Two jointly WSS processes are both WSS, and their cross-correlation function depends only on their time difference.

and

$$\begin{aligned} R_{du,M} &= E\boldsymbol{u}_M^{\mathrm{T}}(\boldsymbol{u}_P h_P + \boldsymbol{v}) = E\boldsymbol{u}_M^{\mathrm{T}}\boldsymbol{u}_P h_P \\ &= E\boldsymbol{u}_M^{\mathrm{T}}\begin{bmatrix}\boldsymbol{u}_M & \bar{\boldsymbol{u}}_N\end{bmatrix}\begin{bmatrix}h_M \\ \bar{h}_N\end{bmatrix} \\ &= R_{u,M}h_M + R_{u,M,N}\bar{h}_N. \end{aligned} \tag{2.14}$$

Then, the optimal filter weights and the respective minimum cost are given, respectively, by [50]

$$w_M^o = h_M + R_{u,M}^{-1}R_{u,M,N}\bar{h}_N \tag{2.15}$$

and

$$J_{\min} = \sigma_v^2 + \bar{h}_N^{\mathrm{T}}\left[R_{u,N} - R_{u,M,N}^{\mathrm{T}}R_{u,M}^{-1}R_{u,M,N}\right]\bar{h}_N. \tag{2.16}$$

The term $R_{u,M}^{-1}R_{u,M,N}\bar{h}_N$ in (2.15) is the estimation bias, which is nonzero only when the input signal is correlated (for white input, $R_{u,M,N} = 0_{M,N}$) and when the filter is undermodeled. The term $R_{u,N} - R_{u,M,N}^{\mathrm{T}}R_{u,M}^{-1}R_{u,M,N}$ in (2.16) is the Schur complement of $R_{u,M}$ in $R_{u,P}$. It can be shown [51] that, if $R_{u,M}$ is positive definite, then its Schur complement in $R_{u,P}$ is positive semi-definite if and only if $R_{u,P}$ is positive semi-definite. Autocorrelation matrices, such as $R_{u,M}$ and $R_{u,P}$, are positive semi-definite [3], but for practical signals they are usually positive definite. Therefore, we can say that in practice the Schur complement of $R_{u,M}$ is positive definite, which implies that the term $\bar{h}_N^{\mathrm{T}}\left[R_{u,N} - R_{u,M,N}^{\mathrm{T}}R_{u,M}^{-1}R_{u,M,N}\right]\bar{h}_N$ is nonnegative, and that the absolute minimum cost is σ_v^2, attained only if $\bar{h}_N = 0_N$.

When $M = P$, the terms $\bar{h}_N$, $\bar{\boldsymbol{u}}_N$, $R_{u,M,N}$ and $R_{u,N}$ are empty arrays, and the derivations above follow analogously, but these terms do not appear. The optimal solution and the minimum error become, respectively,

$$w_P^o = h_P \quad \text{and} \quad J_{\min} = \sigma_v^2. \tag{2.17}$$

When the filter overmodels the impulse response, that is, when $M > P$, we can artificially fill the vector h_P with zeros[2] up to the length M, obtaining a case analogous to the case for $M = P$, where the last $M - P$ terms in h_M are zero. Thus, we see that the case of undermodeled filter has the most complete derivations, and the results for the other cases can be obtained straightforwardly.

[2] When we say that the impulse response has length P, we say that it only admits nonzero coefficients from $h(0)$ to $h(P-1)$. Outside this range, the coefficients are all implicitly zero.

2.4 Adaptive algorithms

The optimal solution (2.9) depends on the exact knowledge of the statistics of the environment, namely, the autocorrelation matrix $R_{u,M}$ and the cross-correlation vector $R_{du,M}$. As these quantities are usually unknown in practice, this solution cannot be applied directly. That is where adaptive filters come into scene. Adaptive filters are provided with an adaptive algorithm that estimates these statistical quantities from the environment in real time, enabling a practical approximation of the linear optimal solution.

The optimal solution (2.9) can be obtained recursively by the *steepest descent algorithm* [1, 2, 35]

$$w_{M,i} = w_{M,i-1} - \mu \nabla J(w_{M,i-1}), \tag{2.18}$$

where the adaptation step size μ is a scalar usually much smaller than one, and the gradient $\nabla J(w_M)$ of the cost function (2.8) is [35]

$$\nabla J(w_M) = R_{u,M} w_M - R_{du,M}. \tag{2.19}$$

As the cost function (2.8) is convex[3] [1, 2, 35], the steepest descent algorithm always converges to the global minimum (2.9).

The steepest descent algorithm avoids the computation of the inverse $R_{u,M}^{-1}$, but still depends on the unknown statistical quantities. If we approximate them by the instantaneous measures

$$R_{u,M} \approx u_{M,i}^{\mathrm{T}} u_{M,i}, \quad R_{du,M} \approx d(i) u_{M,i}^{\mathrm{T}}, \tag{2.20}$$

then the algorithm (2.18) becomes

$$\begin{aligned} w_{M,i} &= w_{M,i-1} - \mu \left(u_{M,i}^{\mathrm{T}} u_{M,i} w_{M,i-1} - d(i) u_{M,i}^{\mathrm{T}} \right) \\ &= w_{M,i-1} + \mu u_{M,i}^{\mathrm{T}} \left(d(i) - u_{M,i} w_{M,i-1} \right), \end{aligned} \tag{2.21}$$

where the term in parenthesis is the filter output error (2.3). Thus,

$$w_{M,i} = w_{M,i-1} + \mu u_{M,i}^{\mathrm{T}} e(i), \tag{2.22}$$

which is the *least-mean-square* (LMS) algorithm, probably the most popular adaptive algorithm used in practice due to its low computational cost, robustness and easy implementation. We assume that the algorithm starts operating at the instant $i = 0$ given an initial condition $w_{M,-1}$, usually taken as $w_{M,-1} = 0_M$.

[3] Provided that $R_{u,M}$ is positive definite, as is in most of the practical cases.

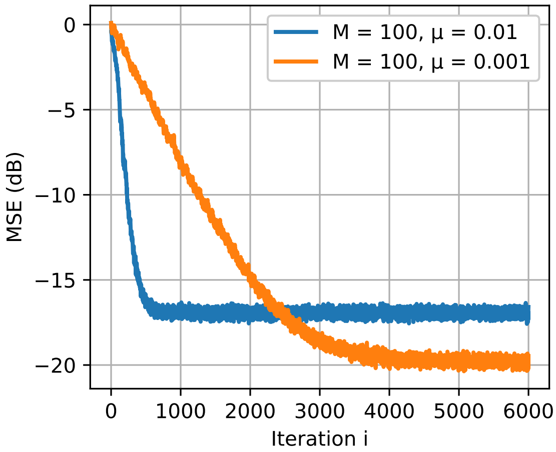


Figure 2: Illustrating the MSE performance of LMS filters. Curves obtained from the ensemble average of 1000 realizations, with white input of variance $\sigma_u^2 = 1$, noise variance $\sigma_v^2 = 0.01$ and impulse response of length $P = 100$.

Figure 2 illustrates the performance curves of two LMS filters, both with the same length but distinct step sizes. The cost (2.8) (MSE) is approximated by taking the ensemble average of several independent realizations of the learning curve. These curves exemplify the well-known dilemma when designing an LMS algorithm: within a certain range of step size values, when increasing the step size, the convergence rate of the algorithm increases but at the same time the error level in steady-state also increases, whereas an ideally desired performance would be high convergence rate and low steady-state error.

Other adaptive algorithms can be obtained from recursive algorithm structures other than (2.18), or by using other cost functions or by using other strategies to improve some specific attribute. For example, to reduce the complexity of LMS, we can quantize the error (and eventually the step size) in the term $\mu u_{M,i}^{\mathrm{T}} e(i)$, so it can be implemented more efficiently by means of shift registers [52, 53]. Another particular algorithm that will be of interest in this text is the normalized LMS (ϵ-NLMS) algorithm

$$w_{M,i} = w_{M,i-1} + \frac{\mu}{\epsilon + ||u_{M,i}||^2} u_{M,i}^{\mathrm{T}} e(i). \tag{2.23}$$

where the regularization factor ϵ is a small scalar. It can be shown that this algorithm also approximates the solution (2.9), but it is derived using Newton's recursion [35]. The ϵ-NLMS algorithm is more robust than the LMS in case the power of the input signal $u(i)$ fluctuates considerably, such as for a speech signals, since the step size is normalized by $||u_{M,i}||^2$.

2.5 Combinations of adaptive filters

The algorithms presented in the previous subsection are adequate for single adaptive filters, also called as standalone filters. We can improve the performance of adaptive filtering by combining distinct standalone filters into a scheme called combination of filters.

Combinations of adaptive filters have been used to improve the performance of adaptive algorithms by taking advantage of the best characteristics of multiple independent filters. If properly designed, the overall combination works at least as well as the best component filter at every iteration, making combinations a good strategy to deal with transient/steady-state trade-offs and to improve tracking performance [28,29,31,34,54,55].

Concretely, the combination outputs a weighted sum of the outputs of the component filters, adaptively adjusted by a supervisor. In a convex combination of two filters, the overall output combines the component filter outputs $y_1(i)$ and $y_2(i)$ as

$$y_c(i) = \lambda(i)y_1(i) + [1 - \lambda(i)]\, y_2(i), \tag{2.24}$$

where the mixing parameter is $0 \leq \lambda(i) \leq 1$. The role of the supervisor is to adjust $\lambda(i)$ so that $y_c(i) = y_1(i)$ when this is the filter with the best performance at a certain instant, and $y_c(i) = y_2(i)$ when the second component filter is the best. When the performance of the component filters are comparable, $y_c(i)$ is a weighted combination of both.

A simple approach to adjust $\lambda(i)$ is proposed in [28] to minimize the instantaneous squared error $e_c^2(i)$, where

$$e_c(i) = d(i) - y_c(i) \tag{2.25}$$

is the error of the overall combination. A more robust adaptation of $\lambda(i)$ is proposed in [29], where it is computed via the auxiliary variable $a(i)$ by

$$\lambda(i) = \frac{1}{1 + \mathrm{e}^{-a(i-1)}} \tag{2.26}$$

and $a(i)$ is updated as

$$a(i) = a(i-1) + \mu_a e_c(i)\, [e_2(i) - e_1(i)]\, \lambda(i)\, [1 - \lambda(i)]\,, \tag{2.27}$$

where μ_a is a small step size, and $e_k(i) = d(i) - y_k(i)$, $k = 1, 2$, are the errors of each component filter. The update of $a(i)$, however, stops whenever $\lambda(i)$ is too close to the limit values of zero or one. This can be avoided by restricting the values of $a(i)$ within a symmetric interval $[-a_+, a_+]$ so that $\lambda(i)$ keeps a minimum distance from zero and one, and thus $a(i)$ never stops adapting. A common practice is setting $a_+ = 4$ [56]. This restriction, however, creates the problem that the mixing parameter never reaches zero or one, and this affects the combination performance. The restriction of values of $a(i)$, therefore, should work along with the thresholding of the values of $\lambda(i)$ as

$$\lambda_u(i) = \begin{cases} 1, & a(i) \geq a_+ - \delta \\ \lambda(i), & a_+ - \delta < a(i) < -a_+ + \delta \\ 0, & -a(i) \leq a_+ + \delta \end{cases}, \tag{2.28}$$

where δ is the threshold, and then the combination overall output is computed as

$$y_c(i) = \lambda_u(i) y_1(i) + [1 - \lambda_u(i)]\, y_2(i). \tag{2.29}$$

The order of magnitude of a well designed step size μ_a varies significantly according to the environment, which makes it a hard-designing parameter, and makes it suffer from lack of robustness for time-varying environments. Aiming at these inconveniences, [54] proposes the power normalization of the step size μ_a, where $a(i)$ is adapted as

$$a(i) = a(i-1) + \frac{\mu_a}{p(i)} e_c(i) \left[e_2(i) - e_1(i)\right] \lambda(i) \left[1 - \lambda(i)\right], \tag{2.30}$$

where the step size μ_a is normalized by $p(i)$, an estimate of the power of $e_2(i) - e_1(i)$, which is updated every iteration as

$$p(i) = bp(i-1) + (1-b)\left[e_2(i) - e_1(i)\right]^2, \tag{2.31}$$

where $0 < b < 1$. This procedure for estimating $\lambda(i)$ was shown to track closely the optimum value of $\lambda(i)$ [29,31,57] (in the sense of minimizing the combination MSE $E\boldsymbol{e}_c^2(i)$).

When used with LMS/ϵ-NLMS algorithms, combinations help with the dilemma between fast convergence rate and low steady-state error, incorporating both the convergence rate of a fast filter and the steady-state of a slow one, as illustrated in Figure 3, where the two LMS filters with different step sizes of Figure 2 are combined.

A strategy to further improve combinations is the weight feedback scheme [34], which anticipates the convergence of the slower filter in the combination. Consider that the component filters may have different lengths $M_1 \leq M_2$, and that their weights are $w_{1,M_1,i}$ and $w_{2,M_2,i}$. Consider the partition $w_{2,M_2,i} = \left[w_{2,M_1,i}^{\mathrm{T}} \;\; \bar{w}_{2,M_2-M_1,i}^{\mathrm{T}}\right]^{\mathrm{T}}$. In a combination

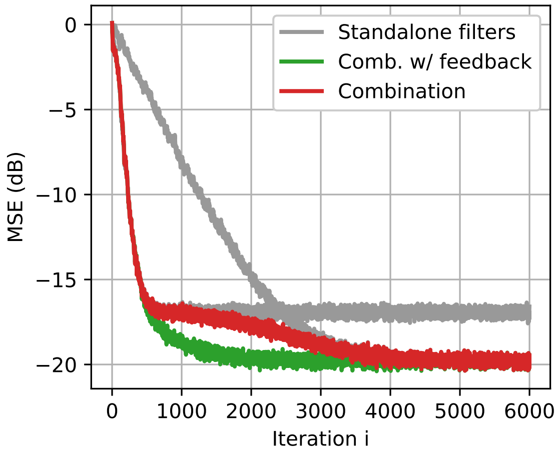


Figure 3: Illustrating the MSE performance of a combination of filters by combining the filters from Figure 2. Combination parameters are $\alpha_+ = 4$, $\delta = 1$, $b = 0.95$, $\mu_a = 1$ and the feedback period is $L = 300$.

with weight feedback, the equivalent weights of the overall combination, computed as

$$\begin{aligned} w_{c,M_2,i} &= \lambda(i) \begin{bmatrix} w_{1,M_1,i} \\ 0_{M_2-M_1} \end{bmatrix} + [1-\lambda(i)]\, w_{2,M_2,i} \\ &= \begin{bmatrix} \lambda(i) w_{1,M_1,i} + [1-\lambda(i)]\, w_{2,M_1,i} \\ [1-\lambda(i)]\, \bar{w}_{2,M_2-M_2,i} \end{bmatrix} = \begin{bmatrix} w_{c,M_1,i} \\ \bar{w}_{c,M_2-M_1,i} \end{bmatrix}, \end{aligned} \tag{2.32}$$

is fed back to each component filter once in a period of L iterations. Concretely, at each iteration i before the weight is updated by the adaptive algorithm, we update filter k, for $k = 1, 2$, as

$$w_{k,M_k,i} = \begin{cases} w_{c,M_k,i}, & \text{for } i = \ell L \\ w_{k,M_k,i}, & \text{otherwise} \end{cases}, \tag{2.33}$$

where $\ell \in \mathbb{Z}$. The advantage of such a scheme over a standard combination is illustrated in Figure 3.

2.6 Performance measures

As the algorithms discussed are practical approximations to minimize the cost function (2.8), the MSE arises as the natural measure for evaluating the performance of an adaptive algorithm. However, in the literature of adaptive filtering [1,3,35], it is usual to evaluate instead measures that are related to the MSE, but that are more meaningful depending on the application. These alternative measures are also more sensitive to distinguish errors that are closer to the minimum possible error.

So far the filter weight vector $w_{M,i}$ has been treated as a deterministic variable since it is computed by deterministic operations from observed variables, such as in (2.22) and (2.23). When measuring performance, however, we aim at measuring its statistical behavior, and so we regard it as a random variable $\boldsymbol{w}_{M,i}$.

The minimum MSE $J_{\min}$ for the underlying estimation problem, provided that the system output is modeled as (2.5), is given by (2.16). Although this minimum value depends on the filter length, the absolute minimum is attained when $M \geq P$, yielding $J_{\min} = \sigma_v^2$ (as given in (2.17)).

A common practice is to evaluate the filter performance by the error that exceeds this minimum estimation error, which is called excess MSE (EMSE). We define the EMSE as the error that exceeds the absolute minimum error σ_v^2, so that

$$\mathrm{MSE}(i) \triangleq E\boldsymbol{e}^2(i) = \sigma_v^2 + \mathrm{EMSE}(i). \tag{2.34}$$

Some works on undermodeled adaptive filters [9, 10] consider the EMSE relative to the minimum error for specific length M, just as in (2.16), but this hinders the EMSE to fairly compare the performance of filters with different lengths.

It can be shown that the EMSE, under the assumptions in Section 2.2, is nonnegative. Consider that $M \leq P$. Using (2.5), the MSE can be expanded as

$$E\boldsymbol{e}^2(i) = E\left(\boldsymbol{d}(i) - \boldsymbol{u}_{M,i}\boldsymbol{w}_{M,i-1}\right)^2 = E\left(\boldsymbol{v}(i) + \boldsymbol{u}_{P,i}\boldsymbol{h}_{P,i} - \boldsymbol{u}_{M,i}\boldsymbol{w}_{M,i-1}\right)^2. \tag{2.35}$$

As $\boldsymbol{v}(i)$ is zero-mean and assumed to be independent from the other quantities, we have

$$E\boldsymbol{e}^2(i) = \sigma_v^2 + E\left(\boldsymbol{u}_{P,i}\boldsymbol{h}_{P,i} - \boldsymbol{u}_{M,i}\boldsymbol{w}_{M,i-1}\right)^2. \tag{2.36}$$

Comparing (2.36) with (2.34), we note that

$$\mathrm{EMSE}(i) = E\left(\boldsymbol{u}_{P,i}\boldsymbol{h}_{P,i} - \boldsymbol{u}_{M,i}\boldsymbol{w}_{M,i-1}\right)^2, \tag{2.37}$$

concluding, therefore, that the EMSE is a nonnegative quantity. Note that the EMSE measures directly the difference between the filter output and the system output without noise, so it is conceptually interesting in applications such as interference cancelling and active noise control. We can simplify the EMSE expression (2.37) by defining the *a priori* *estimation error*[4]

$$\boldsymbol{e}_a(i) \triangleq \boldsymbol{u}_{P,i}\boldsymbol{h}_{P,i} - \boldsymbol{u}_{M,i}\boldsymbol{w}_{M,i-1} = \boldsymbol{u}_{P,i}\left(\boldsymbol{h}_{P,i} - \begin{bmatrix}\boldsymbol{w}_{M,i-1}\\ 0_N\end{bmatrix}\right), \tag{2.38}$$

then the excess MSE can be written as

$$\text{EMSE}(i) = E\boldsymbol{e}_a^2(i). \tag{2.39}$$

For the stationary case (time-invariant impulse response $\boldsymbol{h}_{P,i} = h_P$) and with stationary and white input ($R_{u,P} = \sigma_u^2 I_P$), the EMSE can be decomposed in a convenient way. Define the *weight error vector*

$$\tilde{\boldsymbol{w}}_{P,i}^M \triangleq \boldsymbol{h}_{P,i} - \begin{bmatrix}\boldsymbol{w}_{M,i}\\ 0_N\end{bmatrix}. \tag{2.40}$$

Note that this is not the term that appears in (2.38). In (2.40), the filter weights and the impulse response are considered in the same time instant. However, for time-invariant systems, $\boldsymbol{h}_{P,i} = h_P$ is independent of the time index, and the a priori estimation error (2.38) can be rewritten simply as

$$\boldsymbol{e}_a(i) = \boldsymbol{u}_{P,i}\tilde{\boldsymbol{w}}_{P,i-1}^M. \tag{2.41}$$

Assuming also that $\boldsymbol{u}_{P,i}$ is statistically independent[5] from $\boldsymbol{w}_{P,i}$ and from h_P, then (2.39) becomes

$$\text{EMSE}(i) = E\left[\left(\tilde{\boldsymbol{w}}_{P,i-1}^M\right)^{\mathrm{T}}\left(E\boldsymbol{u}_{P,i}^{\mathrm{T}}\boldsymbol{u}_{P,i}\right)\tilde{\boldsymbol{w}}_{P,i-1}^M\right] = \sigma_u^2 E||\tilde{\boldsymbol{w}}_{P,i-1}^M||^2, \tag{2.42}$$

that is, for stationary environment and white input, we can directly measure the EMSE from the mean-square deviation

$$\text{MSD}(i) \triangleq E||\tilde{\boldsymbol{w}}_{P,i}^M||^2. \tag{2.43}$$

As measuring the error directly from the model weights, the MSD is particularly meaningful for applications such as system identification and time delay estimation.

In the analysis of the undermodeled filters, it will also be convenient to define partial

[4] We call it estimation error because it measures the difference between the filter estimate $\boldsymbol{y}(i)$ and the optimal estimate $\boldsymbol{u}_{P,i}h_P$, obtained in the case of $M = P$. We characterize it as *a priori* because we consider the filter weights before the update of the current iteration.

[5] In fact, this is not true because $\boldsymbol{w}_{P,i}$ depends on $\boldsymbol{u}_{M,i-1}$, which is correlated with $\boldsymbol{u}_{M,i}$. However, for analysis purposes, the statistical independence between $\boldsymbol{u}_{M,i-1}$ and $\boldsymbol{u}_{M,i}$ is usually assumed in the literature [1, 4, 35], and it is one of the well-known independence assumptions.

error measures, comprising only the weights in the range of the filter. So, define the partial weight error vector

$$\tilde{\boldsymbol{w}}_{M,i} \triangleq \boldsymbol{h}_{M,i} - \boldsymbol{w}_{M,i}, \tag{2.44}$$

and the partial a priori estimation error

$$\boldsymbol{e}_{a,M}(i) \triangleq \boldsymbol{u}_{M,i}\boldsymbol{h}_{M,i} - \boldsymbol{u}_{M,i}\boldsymbol{w}_{M,i-1} = \boldsymbol{u}_{M,i}\left(\boldsymbol{h}_{M,i} - \boldsymbol{w}_{M,i-1}\right). \tag{2.45}$$

From (2.40) and (2.44), note that

$$\tilde{\boldsymbol{w}}_{P,i}^{M} = \begin{bmatrix} \boldsymbol{h}_{M,i} \\ \bar{\boldsymbol{h}}_{N,i} \end{bmatrix} - \begin{bmatrix} \boldsymbol{w}_{M,i} \\ 0_N \end{bmatrix} = \begin{bmatrix} \tilde{\boldsymbol{w}}_{M,i} \\ \bar{\boldsymbol{h}}_{N,i} \end{bmatrix}. \tag{2.46}$$

Taking the expectation of the squared norm of (2.46), the MSD can be obtained as

$$\mathrm{MSD}(i) = E||\tilde{\boldsymbol{w}}_{P,i}^{M}||^2 = E||\tilde{\boldsymbol{w}}_{M,i}||^2 + E||\bar{\boldsymbol{h}}_{N,i}||^2. \tag{2.47}$$

All the performance measures MSE, EMSE and MSD are mean functions of some random process. In order to assess these quantities from simulations, we take the ensemble average of a large number of independent realizations of the corresponding random processes.

3 ANALYSIS AND DESIGN OF UNDERMODELED FILTERS IN STATIONARY SCENARIOS

In the literature, the analyses of adaptive algorithms are commonly carried out assuming that the adaptive filter has the same tap-length as the optimal weight vector (or the actual impulse response) [1–4, 12, 58, 59]. Although this framework allows the understanding of how different parameters, such as step size, environment noise and number of taps, influence the algorithm performance, it does not allow to evaluate the filter length as a free variable: the filter length is bounded to the length of the impulse response. We are not able to evaluate the influence of the filter length on the algorithm performance for a fixed impulse response scenario.

When considering the filter length designed independently from the system impulse response, two new situations arise: undermodeling and overmodeling. As discussed in Section 2.3, the overmodeling case can be comprised into the case in which the filter has the same length as the impulse response. The extension of the performance analysis for undermodeling, however, is nontrivial. Although undermodeling has a vast literature in other fields, such as in control theory [5, 6], its study in adaptive filtering is relatively recent. Some brief but general analyses of undermodeling can be found in [3, 4], and more involved analyses for LMS and ϵ-NLMS algorithms can be found in [8–10]. We can also find some analysis of undermodeling in adaptive IIR (Infinite Impulse Response) filters in [60–62].

In this chapter, we evaluate the performance of the LMS and the ϵ-NLMS algorithms as function of the filter length, specifically, for the undermodeling case. We consider in this chapter the case of stationary environments (time-invariant impulse response), and address the nonstationary case (time-variant impulse response) in the next chapter.

When comparing the performance of distinct algorithms, it is customary in the literature to fix the transient performance of the algorithms and compare their steady-state error (or vice-versa) [30]. However, in the case of this study, there are two degrees of freedom in the filter design: even fixing the filter length, the transient and steady-state performance is highly dependent on the choice of the step size. And it is worth recalling that filters of different lengths have different stability bounds for the step size. Thus, in order to offset this difference when we compare filters of different lengths, we adjust the step size according to the best step size (the step size of maximum convergence rate, which we will call "optimal" step size) of a given filter length.

Using the analysis results, we then propose a novel method for selecting the length of an adaptive filter based on the pondering between the transient and steady-state performance. The proposed method selects a length in the imminence of relevant undermodeling, and, as we will show in simulations, a filter with such a length attains fast convergence rate with small degradation due to undermodeling in steady-state. Also, by taking a length value in the imminence of relevant undermodeling, the method yields a filter with less computational complexity when compared to other filters with similar steady-state performance.

The proposed method can also be used within a combination of filters. Since the filter with the proposed length benefits from fast convergence rate, it fits into the role of a fast filter in a combination, working along with a slow filter to ensure good steady-state performance. We also propose a simple modification on the weight feedback period from [30] to comprise undermodeled filters.

This chapter is organized as follows: in Section 3.1, we firstly present some notes and assumptions about the impulse response; in Section 3.2, we derive the mean-square behavior of the LMS and ϵ-NLMS algorithms for the white input case, and then analyze their performance as function of the filter length; in Section 3.3, we describe the design of the filter length and present simulation results comparing a filter with the proposed length to filters with other lengths; finally, in Section 3.4, we show the benefits of designing the fast filter in a combination with the proposed method.

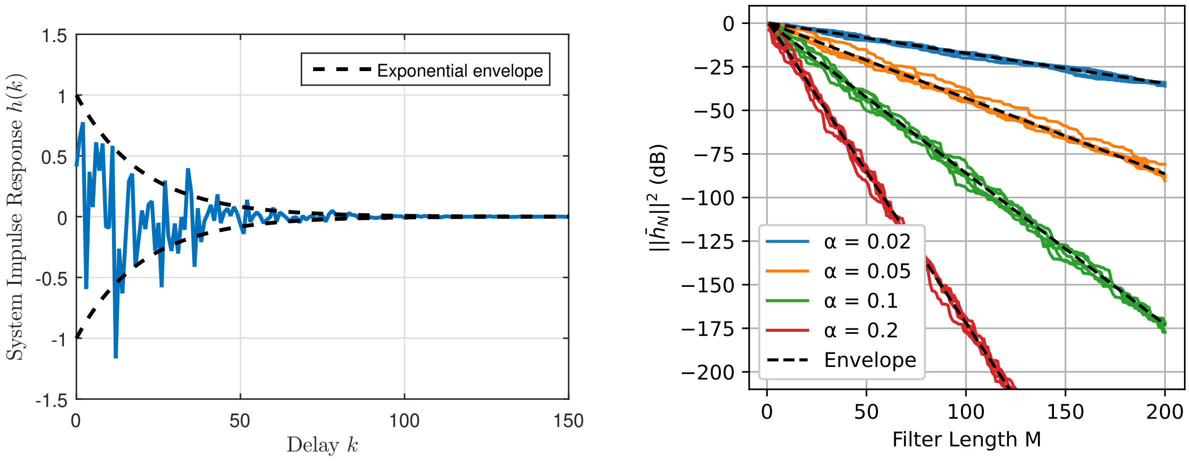
3.1 Exponential decay impulse response

In this and in the following chapter, we derive some expressions that rely on the system impulse response partition terms $||h_M||^2$ and $||\bar{h}_N||^2$. With no contraints, these terms have as many degrees of freedom as the number of elements in h_M and $\bar{h}_N$, respectively. In order to treat these variables conveniently, in some parts of the text we make a simplifying assumption about the system impulse response so as to make these terms parametric functions.

In some parts of this text, we will assume that the system impulse response is infinite and has an *exponential decay envelope*, so that it can be written as

$$h(k) = \beta \mathrm{e}^{-\alpha k} h_b(k), \text{ for } k = 0, 1, \ldots, \infty, \tag{3.1}$$

where scalar $\alpha > 0$ is the exponential decay factor, $\beta > 0$ is the scale factor, and $h_b(k)$ is the base sequence of the impulse response. In order to preserve the overall exponential shape of $h(k)$, the moving average of $h_b(k)$ must be approximately constant, such as in a sinusoid or in a realization of a stationary process. The order of magnitude of $h_b(k)$ is not crucial, since it can be offset by the constant β, although it is more meaningful for β to represent the order of magnitude of $h(k)$ as a whole. Figure 4a illustrates a impulse response under such an assumption.



(a)

(b)

Figure 4: (a) Example of an impulse response with exponential decay envelope, where $\alpha = 0.05$, $||h_P||^2 = 1$ and $h_b(k)$ is a Gaussian i.i.d. sequence; (b) $||\bar{h}_N||^2$ as a function of M for system impulse responses with different decay α, where $h_b(k)$ are Gaussian i.i.d. sequences, $||h_P||^2 = 1$ and $P = 500$, that is sufficiently long for the simulation cases.

If we approximate h_P by its exponential envelope, we can write

$$||\bar{h}_N||^2 \approx \sum_{k=M}^{\infty} \left(\beta \mathrm{e}^{-\alpha k}\right)^2 = \beta^2 \frac{\mathrm{e}^{-2\alpha M}}{1 - \mathrm{e}^{-2\alpha}} = \mathrm{e}^{-2\alpha M} ||h_P||^2. \tag{3.2}$$

Now, $||\bar{h}_N||^2$ is a function of M parameterized by α and $||h_P||^2$. From (2.11), $||h_P||^2 = ||h_M||^2 + ||\bar{h}_N||^2$, and then

$$||h_M||^2 \approx (1 - \mathrm{e}^{-2\alpha M}) ||h_P||^2, \tag{3.3}$$

which will conveniently simplify some expressions further in this text. Figure 4b illustrates how the exponential envelope approximates the impulse response. We depict $||\bar{h}_N||^2 \times M$ for several impulse responses h_P, for fixed $||h_P||^2$ and different values of α, where $h_b(k)$ is simulated as Gaussian i.i.d. sequences.

Assumption (3.1) will not be necessary, in particular, in the derivations of Sections 3.2 and 4.2, and we use it only when analyzing simulation results since the decay α has experimentally proved to influence relevantly the algorithm performance. In Section 3.3, it will be a necessary assumption to significantly simplify the derivations for the proposed design method.

This exponential decay assumption has practical fundament. When a system can be modeled as a finite-order linear system, its transfer function is given by a rational function [63]. The impulse response of such a transfer function is a sum of exponentials, and α can be taken as the decay rate of the dominant pole. The exponential envelope impulse response appears in practical scenarios such as acoustic paths [19, 24, 25] and radio communication systems [26, 27], and is also assumed in some adaptive filtering studies [8, 18].

3.2 Performance analysis in stationary scenarios

In this section, we analyze the performance of undermodeled filters for stationary environments considering that the input signal is white, that is, $R_{u,P} = \sigma_u^2 I_P$. We make the white input assumption because it significantly simplifies the transient analysis. We derive the mean-square convergence model using a general framework called data-normalized algorithm, and we obtain expressions for the transient and steady-state performances for the LMS and ϵ-NLMS algorithms. We then compare the results of these analytical results to simulated data, and analyze the behavior of the filter performance in terms of the length M under various environment conditions.

We conduct the mean-square analysis using the data-normalized algorithm [35, Ch.9]

$$\boldsymbol{w}_{M,i} = \boldsymbol{w}_{M,i-1} + \mu \frac{\boldsymbol{u}_{M,i}^{\mathrm{T}}}{g(\boldsymbol{u}_{M,i})} \boldsymbol{e}(i), \tag{3.4}$$

where μ is the adaptation step size, $\boldsymbol{e}(i)$ is the instantaneous filter output error (2.3), and $g(\cdot)$ is a positive-valued scalar function that determines the specific algorithm. We are interested, in particular, in the LMS algorithm, obtained when $g(\boldsymbol{u}_{M,i}) = 1$, and in the ϵ-NLMS algorithm, obtained when $g(\boldsymbol{u}_{M,i}) = \epsilon + ||\boldsymbol{u}_{M,i}||^2$. After deriving the transient and steady-state performance in terms of this general algorithm, we compute the particular results for the LMS and ϵ-NLMS algorithms.

Writing the system output $\boldsymbol{d}(i)$ as (2.5) and considering the partition of h_P in (2.11), the output error (2.3) becomes

$$\begin{aligned}\boldsymbol{e}(i) &= \boldsymbol{d}(i) - \boldsymbol{u}_{M,i}\boldsymbol{w}_{M,i-1} \\ &= \boldsymbol{u}_{M,i}h_M + \bar{\boldsymbol{u}}_{N,i}\bar{h}_N + \boldsymbol{v}(i) - \boldsymbol{u}_{M,i}\boldsymbol{w}_{M,i-1} \\ &= \boldsymbol{u}_{M,i}\tilde{\boldsymbol{w}}_{M,i-1} + \boldsymbol{v}(i) + \bar{\boldsymbol{u}}_{N,i}\bar{h}_N. \end{aligned} \tag{3.5}$$

Substracting both sides of (3.4) from h_M, and using (3.5), we have

$$h_M - \boldsymbol{w}_{M,i} = h_M - \boldsymbol{w}_{M,i-1} - \mu\frac{\boldsymbol{u}_{M,i}^{\mathrm{T}}}{g(\boldsymbol{u}_{M,i})}\left(\boldsymbol{u}_{M,i}\tilde{\boldsymbol{w}}_{M,i-1} + \boldsymbol{v}(i) + \bar{\boldsymbol{u}}_{N,i}\bar{h}_N\right), \tag{3.6}$$

$$\tilde{\boldsymbol{w}}_{M,i} = \left(I_M - \mu\frac{\boldsymbol{u}_{M,i}^{\mathrm{T}}\boldsymbol{u}_{M,i}}{g(\boldsymbol{u}_{M,i})}\right)\tilde{\boldsymbol{w}}_{M,i-1} - \frac{\mu\boldsymbol{u}_{M,i}^{\mathrm{T}}}{g(\boldsymbol{u}_{M,i})}\left(\boldsymbol{v}(i) + \bar{\boldsymbol{u}}_{N,i}\bar{h}_N\right). \tag{3.7}$$

Now, we employ the well-known independence assumptions [1,4,35]: we assume that $\boldsymbol{u}_{M,i}$ is statistically independent from $\boldsymbol{u}_{M,j}$, for any $i \neq j$[1], implying that $\boldsymbol{u}_{M,i}$ is an i.i.d. sequence. We also assume that $\bar{\boldsymbol{u}}_{N,i}$ is independent from $\boldsymbol{u}_{M,j}$, for any $i \neq j$. As the filter weights are updated with the input regressor and the system output signal, $\tilde{\boldsymbol{w}}_{M,i-1}$ depends on the past samples of $\boldsymbol{u}(i)$, some of them contained in $\boldsymbol{u}_{M,i}$ and $\bar{\boldsymbol{u}}_{N,i}$. However, under these assumptions, we can consider $\tilde{\boldsymbol{w}}_{M,i-1}$ statistically independent of $\boldsymbol{u}_{M,i}$ and $\bar{\boldsymbol{u}}_{N,i}$, and as we considered that $\boldsymbol{v}(i)$ is an i.i.d. process, $\tilde{\boldsymbol{w}}_{M,i-1}$ is also independent of $\boldsymbol{v}(i)$. In the case of white input, $\bar{\boldsymbol{u}}_{N,i}$ is uncorrelated with $\boldsymbol{u}_{M,i}$. Taking expectation of the squared norm in both sides of (3.7), considering all the above assumptions, we have

$$\begin{aligned}E||\tilde{\boldsymbol{w}}_{M,i}||^2 = E\tilde{\boldsymbol{w}}_{M,i-1}^{\mathrm{T}}\left(I_M - 2\mu T_{1,M} + \mu^2 T_{3,M}\right)^2\tilde{\boldsymbol{w}}_{M,i-1} \\ + \mu^2\sigma_v^2\mathrm{Tr}T_{2,M} + \mu^2\sigma_u^2||\bar{h}_N||^2\mathrm{Tr}T_{2,M}, \end{aligned} \tag{3.8}$$

where

$$T_{1,M} \triangleq E\frac{\boldsymbol{u}_{M,i}^{\mathrm{T}}\boldsymbol{u}_{M,i}}{g(\boldsymbol{u}_{M,i})}, \quad T_{2,M} \triangleq E\frac{\boldsymbol{u}_{M,i}^{\mathrm{T}}\boldsymbol{u}_{M,i}}{g^2(\boldsymbol{u}_{M,i})} \quad \text{and} \quad T_{3,M} \triangleq E\frac{\boldsymbol{u}_{M,i}^{\mathrm{T}}\boldsymbol{u}_{M,i}\boldsymbol{u}_{M,i}^{\mathrm{T}}\boldsymbol{u}_{M,i}}{g^2(\boldsymbol{u}_{M,i})} \tag{3.9}$$

are the algorithm-dependent terms. For white stationary input, these moment matrices are multiples of the identity, and we can define scalars t_k so that

$$T_{k,M} = t_k I_M, \text{ for } k = 1, 2, 3, \tag{3.10}$$

[1] Strictly speaking, this assumption is not valid, but is widely considered in the literature for algorithm analysis because it yields simplified and reliable results.

and rewrite (3.8) as

$$E||\tilde{\boldsymbol{w}}_{M,i}||^2 = \gamma E||\tilde{\boldsymbol{w}}_{M,i-1}||^2 + \mu^2 M t_2 \left(\sigma_v^2 + \sigma_u^2||\bar{h}_N||^2\right), \tag{3.11}$$

where

$$\gamma = 1 - 2\mu t_1 + \mu^2 t_3. \tag{3.12}$$

3.2.1 Transient and steady-state performances

We evaluate the transient performance by the initial difference of the recursion (3.11), which, as we will see, considers noise and undermodeling effects [33], unlike approaches based only on evaluating γ (3.12), as customary in the literature [35,40]. Define the initial difference

$$\kappa \triangleq E||\tilde{\boldsymbol{w}}_{P,0}^{M}||^2 - E||\tilde{\boldsymbol{w}}_{P,-1}^{M}||^2 = E||\tilde{\boldsymbol{w}}_{M,0}||^2 - E||\tilde{\boldsymbol{w}}_{M,-1}||^2. \tag{3.13}$$

Considering null initial condition $w_{M,-1} = 0_M$, then $E||\tilde{\boldsymbol{w}}_{M,-1}||^2 = ||h_M||^2$. And using (3.11) and (3.12) in (3.13), we have

$$\begin{aligned}\kappa &= (\gamma - 1)||h_M||^2 + \mu^2 M t_2 \left(\sigma_v^2 + \sigma_u^2||\bar{h}_N||^2\right)\\ &= (-2\mu t_1 + \mu^2 t_3)||h_M||^2 + \mu^2 M t_2 \left(\sigma_v^2 + \sigma_u^2||\bar{h}_N||^2\right).\end{aligned} \tag{3.14}$$

Note that κ is a quadratic function of μ. Factoring (3.14), we obtain

$$\boxed{\kappa = t_1||h_M||^2\nu\mu\left(\mu - \frac{2}{\nu}\right),} \tag{3.15}$$

where

$$\boxed{\nu = \frac{t_3}{t_1} + \frac{Mt_2}{t_1}\left(\frac{\sigma_v^2 + \sigma_u^2||\bar{h}_N||^2}{||h_M||^2}\right).} \tag{3.16}$$

As κ is a quadratic function of μ (always opening upwards), the step size that minimizes it and the corresponding minimum value are straightforwardly obtained:

$$\boxed{\mu^o = \frac{1}{\nu}, \qquad \kappa^o = -\frac{t_1||h_M||^2}{\nu}.} \tag{3.17}$$

The initial difference, for stable convergence of the algorithm, must be negative. So, the step size that minimizes (3.15) actually maximizes the absolute value of κ (for stable convergence), and the initial difference with the largest modulus yields the fastest convergence rate, and thus we refer to μ^o as the *optimal step size.*

At steady-state, $i \to \infty$ and (3.11) becomes

$$\begin{aligned} E||\tilde{\boldsymbol{w}}_{M,\infty}||^2 &= \gamma E||\tilde{\boldsymbol{w}}_{M,\infty}||^2 + \mu^2 M t_2 \left(\sigma_v^2 + \sigma_u^2||\bar{h}_N||^2\right) \\ &= \frac{\mu M t_2 \left(\sigma_v^2 + \sigma_u^2||\bar{h}_N||^2\right)}{2t_1 - \mu t_3}. \end{aligned} \tag{3.18}$$

Then, using (3.18) in (2.47), the steady-state MSD is given by

$$\boxed{\mathrm{MSD}(\infty) = \frac{\mu M t_2 \left(\sigma_v^2 + \sigma_u^2||\bar{h}_N||^2\right)}{2t_1 - \mu t_3} + ||\bar{h}_N||^2.} \tag{3.19}$$

3.2.2 Moments for LMS and ϵ-NLMS algorithms

Now, we derive the values of t_k for the LMS and the ϵ-NLMS algorithms. For the LMS algorithm ($g(\boldsymbol{u}_{M,i}) = 1$), $T_{1,M}$ and $T_{2,M}$ are just the autocorrelation matrix $\sigma_u^2 I_M$, so that $t_1 = t_2 = \sigma_u^2$. We can simplify $T_{3,M}$ if we assume that the input is Gaussian, so that the fourth-order moments can be written analytically. In Appendix 3.A, we show that

$$E\boldsymbol{u}_{M,i}^{\mathrm{T}}\boldsymbol{u}_{M,i}\boldsymbol{u}_{M,i}^{\mathrm{T}}\boldsymbol{u}_{M,i} = R_{u,M}\mathrm{Tr}R_{u,M} + 2(R_{u,M})^2, \tag{3.20}$$

and, along with the fact that the input is white, $T_{3,M}$ becomes

$$T_{3,M} = \sigma_u^2 I_M(\sigma_u^2 M) + 2\sigma_u^4 I_M = \sigma_u^4(M+2)I_M, \tag{3.21}$$

which implies that $t_3 = \sigma_u^4(M+2)$.

Substituting the values t_k in (3.16), we have

$$\begin{aligned} \nu &= \frac{\sigma_u^4(M+2)}{\sigma_u^2} + M\frac{\sigma_u^2}{\sigma_u^2}\left(\frac{\sigma_v^2 + \sigma_u^2||\bar{h}_N||^2}{||h_M||^2}\right) \\ &= \sigma_u^2\left[2 + M + M\left(\frac{\sigma_v^2 + \sigma_u^2||\bar{h}_N||^2}{\sigma_u^2||h_M||^2}\right)\right] \\ &= \sigma_u^2\left[2 + M\left(\frac{\sigma_v^2 + \sigma_u^2||h_P||^2}{\sigma_u^2||h_M||^2}\right)\right]. \end{aligned} \tag{3.22}$$

Note that the term multiplying M is always larger than 1. For large M, we can approximate

$$\nu \approx \sigma_u^2 M\left(\frac{\sigma_v^2 + \sigma_u^2||h_P||^2}{\sigma_u^2||h_M||^2}\right) = \sigma_u^2 M\frac{||h_P||^2}{||h_M||^2}\left(1 + \frac{1}{\mathrm{SNR}}\right), \tag{3.23}$$

where $\mathrm{SNR} = \sigma_u^2||h_P||^2/\sigma_v^2$ is the signal-to-noise ratio of the system.

For the steady-state expression, substituting t_k for LMS in (3.19), we obtain

$$\mathrm{MSD}(\infty) = \frac{\mu M \left(\sigma_v^2 + \sigma_u^2 ||\bar{h}_N||^2\right)}{2 - \mu\sigma_u^2(M+2)} + ||\bar{h}_N||^2. \tag{3.24}$$

For the ϵ-NLMS algorithm ($g(\boldsymbol{u}_{M,i}) = \epsilon + ||\boldsymbol{u}_{M,i}||^2$), to compute analytically the moments $T_{1,M}$, $T_{2,M}$ and $T_{3,M}$, we must assume that the input signal is white and Gaussian, so that $||\boldsymbol{u}_{M,i}||^2$ can be considered statistically independent from any function of $\boldsymbol{u}_{M,i}$ [59, 64]. We also assume that $\epsilon \approx 0$. Then, as derived in Appendix 3.B, we have

$$T_{1,M} = E\frac{\boldsymbol{u}_{M,i}^{\mathrm{T}}\boldsymbol{u}_{M,i}}{||\boldsymbol{u}_{M,i}||^2} = \frac{1}{M}I_M, \tag{3.25}$$

$$T_{2,M} = E\frac{\boldsymbol{u}_{M,i}^{\mathrm{T}}\boldsymbol{u}_{M,i}}{(||\boldsymbol{u}_{M,i}||^2)^2} = \frac{1}{M(M+2)\sigma_u^2}I_M \tag{3.26}$$

and

$$T_{3,M} = E\frac{\boldsymbol{u}_{M,i}^{\mathrm{T}}\boldsymbol{u}_{M,i}\boldsymbol{u}_{M,i}^{\mathrm{T}}\boldsymbol{u}_{M,i}}{(||\boldsymbol{u}_{M,i}||^2)^2} = \frac{1}{M}I_M, \tag{3.27}$$

from which $t_1 = t_3 = 1/M$ and $t_2 = \left(M(M+2)\sigma_u^2\right)^{-1}$. Substituting these values of t_k in (3.16), we have

$$\nu = 1 + \frac{M}{(M+2)\sigma_u^2}\left(\frac{\sigma_v^2 + \sigma_u^2||\bar{h}_N||^2}{||h_M||^2}\right), \tag{3.28}$$

which becomes, for large M,

$$\nu \approx \left(\frac{\sigma_v^2 + \sigma_u^2||h_P||^2}{\sigma_u^2||h_M||^2}\right) = \frac{||h_P||^2}{||h_M||^2}\left(1 + \frac{1}{\mathrm{SNR}}\right). \tag{3.29}$$

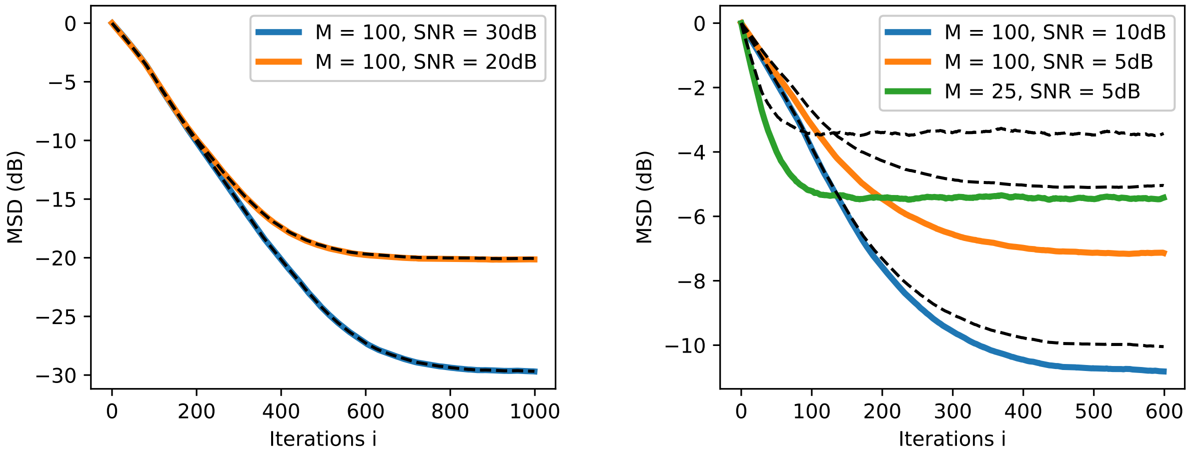
For the ϵ-NLMS algorithm, the steady-state expression (3.19) becomes

$$\mathrm{MSD}(\infty) = \frac{M}{\sigma_u^2(M+2)}\frac{\mu\left(\sigma_v^2 + \sigma_u^2||\bar{h}_N||^2\right)}{2-\mu} + ||\bar{h}_N||^2. \tag{3.30}$$

Note that the term $\left(1 + \mathrm{SNR}^{-1}\right)||h_P||^2/||h_M||^2$ appears for both algorithms in (3.23) and (3.29). When SNR → ∞ and $||h_M||^2 \to ||h_P||^2$, that is, when the noise level and the undermodeling effect become negligible, this term tends to one. This makes the optimal step size μ^o (3.17) (either for the LMS or for the ϵ-NLMS) become the same optimal step size derived by minimizing γ [35,40], which, for the LMS and the ϵ-NLMS algorithms, are respectively,

$$\mu^o = \frac{1}{\sigma_u^2 M} \quad \text{and} \quad \mu^o = 1. \tag{3.31}$$

Figure 5 compares the performance curves of filters with the optimal step size (3.31) and with optimal step size (3.17). When the SNR is high, the difference between using one or the other optimal step size is very subtle. On the other hand, as the SNR decreases, and as the undermodeling effect becomes critical, the steady-state performance when using (3.17) has nitid advantage.



(a)

(b)

Figure 5: Comparing performance curves when using the traditional optimal step size (3.31) (dash-line) and when using the optimal step size (3.17) (solid lines), for the LMS algorithm. The curves were simulated with $\sigma_u^2 = 1$ and exponential h_P with $\alpha = 0.05$ and $||h_P||^2 = 1$.

3.2.3 Comparing to other analyses in the literature

As already mentioned, there are analyses in the literature regarding the performance of undermodeled adaptive filters. Although their results are similar or identical to what we obtain in this text, it is worth pointing out the particularities of each work.

In [8], the mean-square convergence model is derived for the undermodeled LMS (refered to as deficient-length LMS) for white input via the squared norm of the weight error vector $||\tilde{\boldsymbol{w}}_{P,i}^M||^2$, similar to what we do. However, they do not partition the weight error vector as $\tilde{\boldsymbol{w}}_{M,i}$ and $\bar{h}_N$ and derive directly from $\tilde{\boldsymbol{w}}_{P,i}^M$.

In [9], a comprehensive analysis for the undermodeled LMS is derived for general input correlation, based on the the covariance matrix of the weight error vector. Their derivations are carried out with the partial weight error $\tilde{\boldsymbol{w}}_{M,i}$, not considering the error counterpart $\bar{h}_N$ on the excess MSE, that is, their excess error is computed relatively to the optimal error (2.15) of a specific length M. In [10], the same author derives a similar analysis for the undermodeled ϵ-NLMS algorithm, now using the squared norm of the weight error vector.

Our approach to analyze the mean-square convergence model is based on the squared norm of the weight error vector using the data-normalized algorithm framework, similiar to [35, Ch.9], from which the LMS and the ϵ-NLMS algorithms are particular cases. In this way, the analysis carried out in this section can be viewed as the extension of the analysis in [35, Ch.9] for the undermodeled algorithm case, for white input.

Although not explicit, the case of undermodeled filters can be comprised within the classical analyses of full-length filters [3,12,35,58,59]. In the classical analyses, the system impulse response has the same length as the filter, that is, $M = P$ and thus there is no explicit undermodeling. However, the undermodeling can be considered hidden in the noise. From the system output model (2.5), note that

$$\boldsymbol{d}(i) = \boldsymbol{u}_{P,i}h_P + \boldsymbol{v}(i) = \boldsymbol{u}_{M,i}h_M + \bar{\boldsymbol{u}}_{N,i}\bar{h}_N + \boldsymbol{v}(i), \tag{3.32}$$

and, if we define $\boldsymbol{v}_0(i) = \bar{\boldsymbol{u}}_{N,i}\bar{h}_N + \boldsymbol{v}(i)$, the model is equivalent to a system impulse response of length M and noise $\boldsymbol{v}_0(i)$.

When the input is white, $\boldsymbol{v}_0(i)$ is independent of $\boldsymbol{u}_{M,i}$, and if we consider the independence assumptions for $\bar{\boldsymbol{u}}_{N,i}$, that is, that $\bar{\boldsymbol{u}}_{N,i}$ is independent of $\bar{\boldsymbol{u}}_{N,j}$ for $i \neq j$, turning $\boldsymbol{v}_0(i)$ into an i.i.d. process, then the classical analysis would derive the same results as in this text. One important difference is that, with such analysis, the EMSE and MSD would be computed relatively to the optimum error (2.16) of the specific length, that is, relatively to

$$\sigma^2_{v_0} = \sigma^2_v + \sigma^2_u ||\bar{h}_N||^2, \tag{3.33}$$

which is (2.16) for white input.

When the input is correlated, $\boldsymbol{v}_0(i)$ is correlated to $\boldsymbol{u}_{M,i}$, and this strategy to analyze undermodeled filters with the framework of full-length filter analysis cannot be applied. In this case, the optimal solution to which the filter weight converges is biased, as shown in (2.15).

3.2.4 Simulations

Now we explore the theoretical results of transient and steady-state performance derived in this section. We analyze the behavior of κ (3.15) and MSD(∞) (3.19) as functions of the filter length under some different conditions, and compare the theoretical curves to simulated experimental data. We adopt $|\kappa|$ as figure of merit for the transient, instead of just κ, because it is directly proportional to the convergence speed.

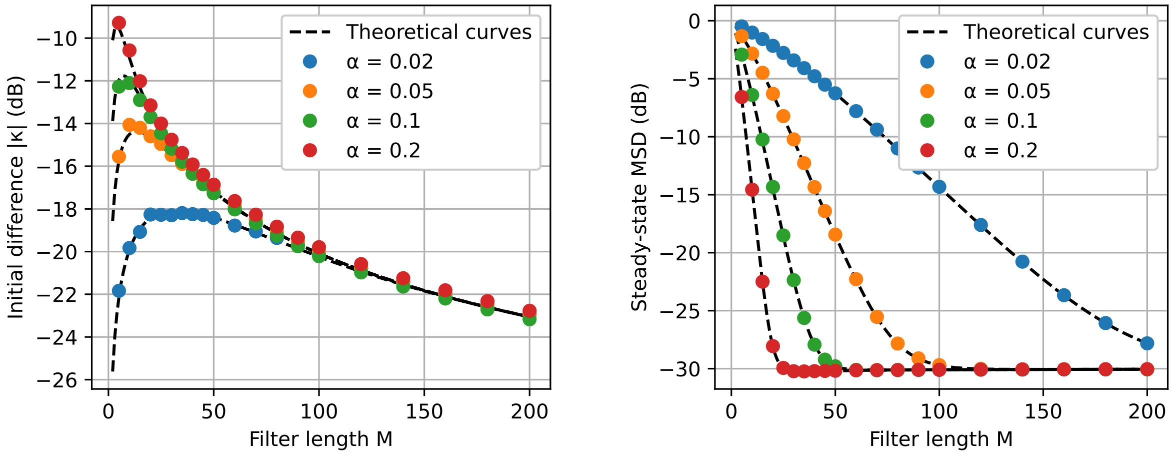
For the following analyses, we consider $\sigma_u^2 = 1$ and impulse responses with $||h_P||^2 = 1$, $P = 500$, which can be considered infinite in the simulated scenarios[2], and $h_b(k) = 1$, for $k = 0, 1, \cdots, P-1$. The other environment parameters (α and σ_v^2) and the step size value change along the tests. The experimental measures for transient and steady-state are computed from approximate MSD curves, obtained from the ensemble average of 1000 realizations of squared-deviation curves $||\tilde{w}_{P,i}^M||^2$. We consider ν with the simplified expressions (3.23) (LMS algorithm) and (3.29) (ϵ-NLMS algorithm) to compute both the theoretical value of $|\kappa|$ and the experimental values of the step size (3.17).

Figure 6 compares the behavior of $|\kappa|$ and $\text{MSD}(\infty)$ using the LMS algorithm for exponential decay rates of $\alpha = 0.02$, 0.05, 0.1 and 0.2. The noise variance is set to $\sigma_v^2 = 0.001$. Every filter runs with the optimal step size (3.17), which varies according to the filter length. Note that the experimental values are very close to the theory. As we decrease the filter length, the initial difference $|\kappa|$ increases until reaching a maximum value, and then declines very fast. This means that short filters (but not excessively short ones) converge faster than large filters. Note that $|\kappa|$ is higher for higher values of decay α when M is small, which means that when the energy of the impulse response is more concentrated at smaller delays, the filter can attain faster convergence rate. For large lengths, on the other hand, almost no difference in $|\kappa|$ is observed for different values of α. The steady-state MSD decreases as we increase M, until reaching a minimum error level. The steady-state MSD decreases faster for steeper decays in the impulse response, but the minimum error level attained for filters of large length is the same no matter the decay α. This happens because, for small length, undermodeling is predominant in the steady-state error and decreases exponentially as M increases. For sufficiently large M, when undermodeling becomes negligible, $\text{MSD}(\infty)$ is roughly constant in terms of M and depends more significantly on the other variables, such as noise variance and step size value.

Figure 7 shows analogous simulations for the ϵ-NLMS algorithm. Note that these results are very similar to the LMS case. For this reason, in the subsequent simulations we show the results only for the LMS algorithm.

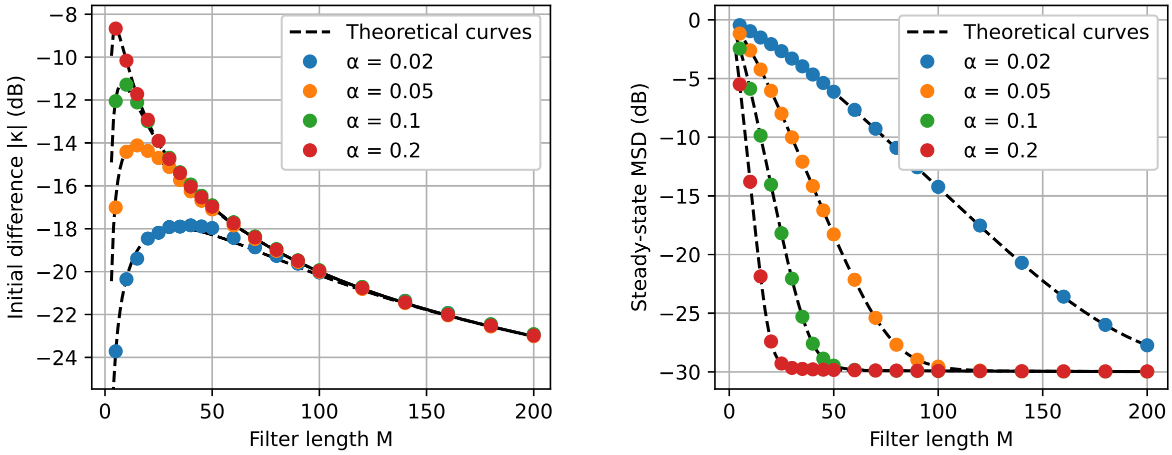
For the curves shown in Figure 8, we fix the impulse response decay rate to $\alpha = 0.05$ and vary the noise level from $\sigma_v^2 = 0.0001$ to $\sigma_v^2 = 0.5$. Note that the noise level affects mostly the steady-state MSD for large filter lengths. As σ_v^2 increases, the level of steady-state MSD for large M increases. The steady-state error of short length filters is also

[2] In the simulated scenario with the slowest decay in the impulse response, when $\alpha = 0.02$, clipping the infinite exponential at $P = 500$ causes a loss of 0.0000002% in the energy of the impulse response.



(a) (b)

Figure 6: Theoretical and experimental curves of (a) initial difference and (b) steady-state error for the LMS algorithm, comparing different decay rates α. Every filter runs with the optimal step size (3.17), and $\sigma_v^2 = 0.001$.



(a) (b)

Figure 7: Theoretical and experimental curves of (a) initial difference and (b) steady-state error for the ϵ-NLMS algorithm, comparing different decay rates α. Every filter runs with the optimal step size (3.17), and $\sigma_v^2 = 0.001$.

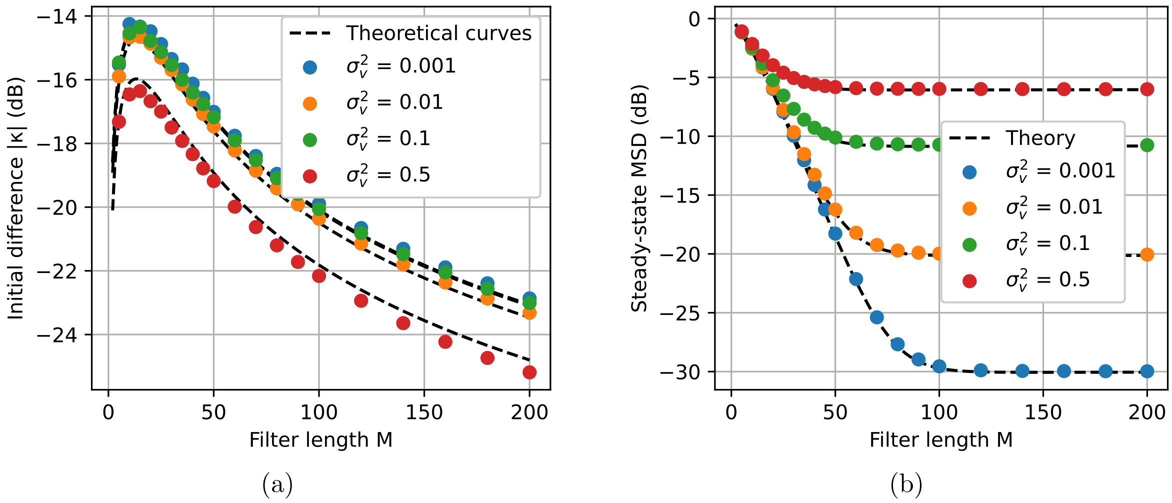


Figure 8: Theoretical and experimental curves of (a) initial difference and (b) steady-state error for the LMS algorithm, comparing different noise levels σ_v^2. Every filter runs with the optimal step size (3.17), and $\alpha = 0.05$.

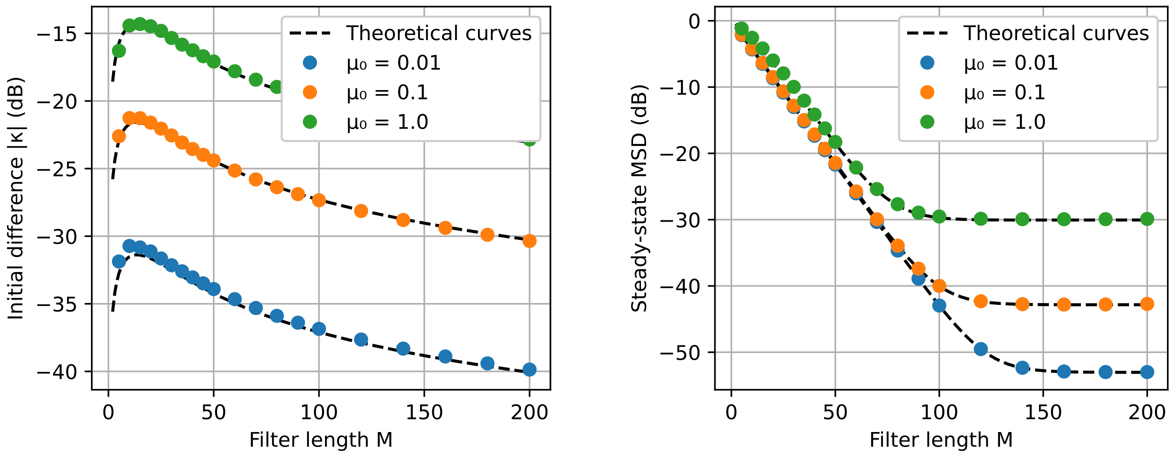
affected, but is less sensitive to variations in the noise power. The initial differences are barely affected by variations of σ_v^2. They subtly decrease as the noise power increases, and the reduction becomes relevant only for $\sigma_v^2 = 0.1$ or higher.

In Figure 9, we fix $\alpha = 0.05$ and $\sigma_v^2 = 0.001$ and now compare the transient and steady-state curves for different values of step size. Consider the decomposition of the step size as

$$\mu = \mu_0 \mu^o, \tag{3.34}$$

where $0 < \mu_0 \leq 1$ is the normalized step size, that is, the ratio of the actual step size μ to the optimal step size. We compare, specifically, curves for different μ_0 (notice that μ still varies for each curve). The behavior of the steady-state performance when varying μ_0 is somewhat similar to when we vary the noise level. As we decrease μ_0, short length filters have almost no effect in their steady-state values, but long filters have lower level of steady-state error. For the initial difference, however, there are significant differences. The curves for $|\kappa|$ shift down considerably as we decrease μ_0, what is expected since the step size controls the rate of convergence.

In all figures presented so far in this subsection, note that the step sizes μ are not constant along each curve. They decrease as M increases, according to the optimal step size μ^o (3.17). Figure 10 shows what happens to the curves for the LMS algorithm if we use a fixed μ instead, as we vary M. For short filters, the curves in Figure 10 are very similar to the curves in Figure 9. But now, as we increase M, the initial difference stays constant (or decreases very slowly), and the steady-state now reaches a point of minimum



(a) (b)

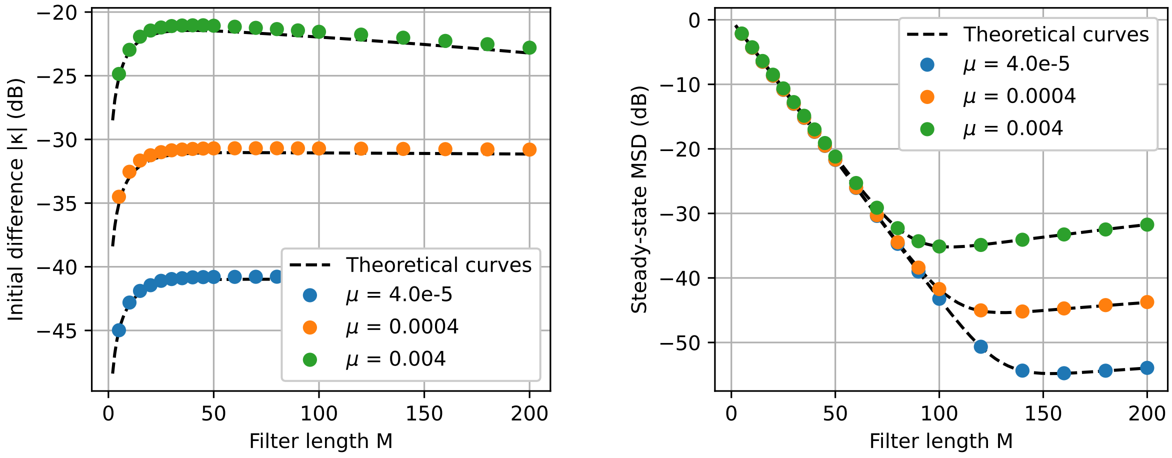
Figure 9: Theoretical and experimental curves of (a) initial difference and (b) steady-state error for the LMS algorithm, comparing different normalized step sizes μ_0. Every filter runs with step size $\mu = \mu_0\mu^o$, $\alpha = 0.05$ and $\sigma_v^2 = 0.001$.

MSD and then begins to grow slowly.

The simulations so far have considered that the impulse response is an (practically) infinite exponential. In the following two simulation cases, we show what happens when the impulse response is clipped, creating an abrupt change along the coefficients in h_P.

Figure 11 depicts the simulation case of Figure 6 where the only difference is that the impulse response is clipped at $k = 70$ ($P = 70$), that is, $h(k) = 0$ for $k \geq 70$. The entries of h_P are also rescaled so as to keep $||h_P||^2 = 1$. The curves of $|\kappa|$ have no apparent difference from the curves in Figure 6a. In Figure 11b, note that the curves of steady-state MSD that does not reach the minimum error level before $M = 70$ in Figure 6b now exhibit an abrupt decrease at this point. The curves that reach the minimum level before $M = 70$, which are the curves for higher values of decay α, are apparently not affected by the impulse response clipping.

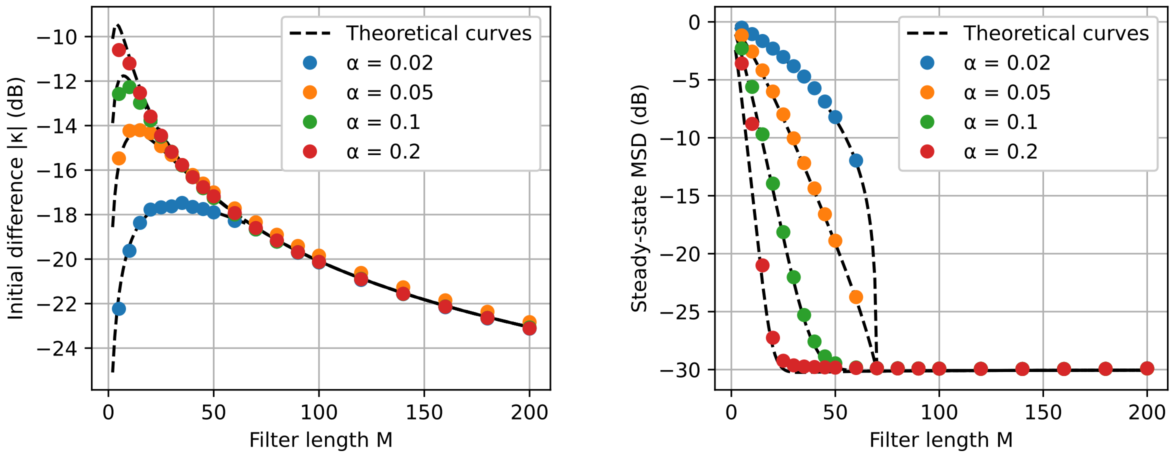
Figure 12 shows an example of these curves when the impulse response is not exponential. We simulate, instead, an impulse response of length $P = 150$, where $h(k)$ is a positive constant for $k = 0, 1, \cdots, P - 1$, and $h(k) = 0$ otherwise, so that $||h_P||^2 = 1$. Note that this is a clipped exponential h_P where $\alpha = 0$. We set $\sigma_v^2 = 0.001$ and compare curves for different step sizes μ_0. Now, the convergence rate increases until the length of the filter equals the length of the impulse response, and decreases for larger filters. The steady-state error is constant when we vary M so that $M \geq P$, but for roughly any length smaller than P, the steady-state error is incomparably higher.



(a)

(b)

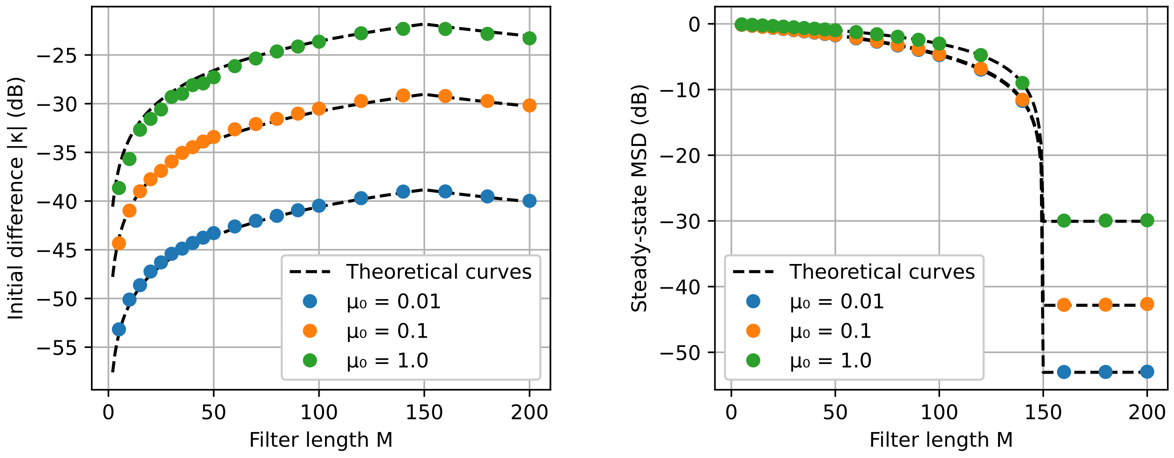
Figure 10: Theoretical and experimental curves of (a) initial difference and (b) steady-state error for the LMS algorithm, comparing different step sizes μ. Now step size μ is constant along each curve, $\alpha = 0.05$ and $\sigma_v^2 = 0.001$.



(a)

(b)

Figure 11: Comparing theoretical and experimental curves of (a) initial difference and (b) steady-state error for the LMS algorithm, comparing different decay rates α. $\sigma_v^2 = 0.001$ and now the impulse response is clipped at $P = 70$.



(a) (b)

Figure 12: Comparing theoretical and experimental curves of (a) initial difference and (b) steady-state error for the LMS algorithm, comparing different normalized step sizes μ_0. Every filter runs with the optimal step size (3.17), $\sigma_v^2 = 0.001$ and now $\alpha = 0$ (rectangular impulse response).

3.3 Design of the filter length

The curves from Subsection 3.2.4 show an interesting pattern in the behavior of the transient and steady-state performance of the LMS/ϵ-NLMS algorithms, when considering an exponential impulse response and white input: increasing the filter length (beyond the peak in the curve of κ, and while keeping the normalized step size μ_0 fixed) reduces the convergence rate and reduces the steady-state error (until reaching a minimum level). This behavior creates a dilemma for the choice of the filter length very similar to the dilemma faced when designing the step size for fixed length: do we reduce the steady-state error at the cost of a slower convergence, or increase convergence rate at the cost of a higher steady-state error?

In order to make a reasonable choice for the filter length, we define an indicator that balances these two attributes. We define the *overall performance* as the ratio between the absolute initial difference $|\kappa|$ and the steady-state MSD

$$\boxed{\eta(M, \mu_0) \triangleq \frac{|\kappa|}{\text{MSD}(\infty)},} \tag{3.35}$$

where the normalized step size μ_0 is so that

$$\boxed{\mu = \mu_0 \mu^o.} \tag{3.36}$$

Larger values of η imply better performance of the filter, in a sense that comprises both the transient and the steady-state performance.

If we take the logarithm of (3.35) (which does not change its extreme points), we obtain

$$\ln \eta(M, \mu_0) = \ln|\kappa| - \ln \mathrm{MSD}(\infty). \tag{3.37}$$

In Subsection 3.2.4, we observed these logarithmic curves for exponential impulse responses[3]: $\ln|\kappa|$ is a smooth hill-shaped function, and $\ln \mathrm{MSD}(\infty)$ is a bounded and decreasing function of M, tending to a constant value as $M \to \infty$. It is expected, and we will see later in this section, that η is a hill-shaped function of M for exponential impulse responses.

We want to find the value of M that maximizes (3.35) (and equivalently maximizes (3.37)), which we call *optimal length* M^o, and use it to design the filter length. Let us expand the terms in (3.35). Using (3.36) in (3.15) and (3.19), where $\mu^o = 1/\nu$, we have

$$\begin{aligned}\kappa &= t_1 ||h_M||^2 \nu \mu_0 \mu^o \left(\mu_0 \mu^o - \frac{2}{\nu}\right) \\ &= -\mu_0 \left(2 - \mu_0\right) \frac{t_1 ||h_M||^2}{\nu},\end{aligned} \tag{3.38}$$

and

$$\mathrm{MSD}(\infty) = \frac{\mu_0 M t_2 \left(\sigma_v^2 + \sigma_u^2 ||\bar{h}_N||^2\right)}{2t_1\nu - \mu_0 t_3} + ||\bar{h}_N||^2. \tag{3.39}$$

The expressions of ν for LMS (3.23) and ϵ-NLMS (3.29), considering the approximation for large M, can be written as

$$\nu \approx t_x \frac{||h_P||^2}{||h_M||^2} \left(1 + \mathrm{SNR}^{-1}\right), \tag{3.40}$$

where $t_x = \sigma_u^2 M$ for the LMS, and $t_x = 1$ for the ϵ-NLMS.

In (3.38), the term that depends on the algorithm is given by ν/t_1. For the LMS algorithm, $t_1 = \sigma_u^2$, and for the ϵ-NLMS algorithm, $t_1 = 1/M$. It turns out that for both algorithms, considering ν as (3.40), ν/t_1 is

$$\frac{\nu}{t_1} \approx M \frac{||h_P||^2}{||h_M||^2} \left(1 + \mathrm{SNR}^{-1}\right), \tag{3.41}$$

and then (3.38) becomes, for both LMS and ϵ-NLMS,

$$\kappa \approx -\frac{\mu_0 \left(2 - \mu_0\right)}{\left(1 + \mathrm{SNR}^{-1}\right) ||h_P||^2} \frac{||h_M||^4}{M}. \tag{3.42}$$

[3] In fact, the curves were in logarithm of base 10, which differs only by a scale factor.

In (3.39), the term that depends on the algorithm is given by

$$\begin{aligned}\frac{2t_1\nu - \mu_0 t_3}{\mu_0 M t_2} &\approx \frac{2t_1 t_x \frac{||h_P||^2}{||h_M||^2}\left(1+\text{SNR}^{-1}\right) - \mu_0 t_3}{\mu_0 M t_2}\\ &= \frac{||h_P||^2}{||h_M||^2}\left[\frac{2t_1 t_x}{\mu_0 M t_2}\left(1+\text{SNR}^{-1}\right) - \frac{\mu_0 t_3}{\mu_0 M t_2}\frac{||h_M||^2}{||h_P||^2}\right]\\ &= \frac{||h_P||^2}{||h_M||^2}\left[\frac{2t_1 t_x}{M t_2}\left(\frac{1+\text{SNR}^{-1}}{\mu_0}\right) - \frac{t_3}{M t_2} + \frac{t_3}{M t_2}\frac{||\bar{h}_N||^2}{||h_P||^2}\right].\end{aligned} \tag{3.43}$$

From the terms t_k derived in Subsection 3.2.2, one can verify that $t_3/t_2M = \sigma_u^2$ and $t_1t_x/Mt_2 = \sigma_u^2$, for both LMS and ϵ-NLMS. Then,

$$\frac{2t_1\nu - \mu_0 t_3}{\mu_0 M t_2} \approx \sigma_u^2\frac{||h_P||^2}{||h_M||^2}\left[2\left(\frac{1+\text{SNR}^{-1}}{\mu_0}\right) - 1 + \frac{||\bar{h}_N||^2}{||h_P||^2}\right]. \tag{3.44}$$

Substituting (3.44) in (3.39), we have

$$\begin{aligned}\text{MSD}(\infty) &\approx \frac{\left(\sigma_v^2 + \sigma_u^2||\bar{h}_N||^2\right) + \sigma_u^2\frac{||h_P||^2}{||h_M||^2}\left[2\left(\frac{1+\text{SNR}^{-1}}{\mu_0}\right) - 1 + \frac{||\bar{h}_N||^2}{||h_P||^2}\right]||\bar{h}_N||^2}{\sigma_u^2\frac{||h_P||^2}{||h_M||^2}\left[2\left(\frac{1+\text{SNR}^{-1}}{\mu_0}\right) - 1 + \frac{||\bar{h}_N||^2}{||h_P||^2}\right]}\\ &= \frac{\sigma_u^2\frac{||h_P||^2}{||h_M||^2}\left[\left(\frac{1}{\text{SNR}} + \frac{||\bar{h}_N||^2}{||h_P||^2}\right)||h_M||^2 + \left(2\left(\frac{1+\text{SNR}^{-1}}{\mu_0}\right) - 1 + \frac{||\bar{h}_N||^2}{||h_P||^2}\right)||\bar{h}_N||^2\right]}{\sigma_u^2\frac{||h_P||^2}{||h_M||^2}\left[2\left(\frac{1+\text{SNR}^{-1}}{\mu_0}\right) - 1 + \frac{||\bar{h}_N||^2}{||h_P||^2}\right]}\\ &= \frac{\frac{||h_M||^2}{\text{SNR}} + \left(2\left(\frac{1+\text{SNR}^{-1}}{\mu_0}\right) - 1\right)||\bar{h}_N||^2 + \frac{||\bar{h}_N||^2||h_M||^2 + ||\bar{h}_N||^4}{||h_P||^2}}{\left[2\left(\frac{1+\text{SNR}^{-1}}{\mu_0}\right) - 1 + \frac{||\bar{h}_N||^2}{||h_P||^2}\right]}.\end{aligned} \tag{3.45}$$

Note that $||\bar{h}_N||^2||h_M||^2 + ||\bar{h}_N||^4 = ||\bar{h}_N||^2(||h_P||^2 - ||\bar{h}_N||^2) + ||\bar{h}_N||^4 = ||h_P||^2||\bar{h}_N||^2$. Then,

$$\begin{aligned}\text{MSD}(\infty) &\approx \frac{\frac{||h_P||^2}{\text{SNR}} - \frac{||\bar{h}_N||^2}{\text{SNR}} + \left(2\left(\frac{1+\text{SNR}^{-1}}{\mu_0}\right) - 1\right)||\bar{h}_N||^2 + ||\bar{h}_N||^2}{2\left(\frac{1+\text{SNR}^{-1}}{\mu_0}\right) - 1 + \frac{||\bar{h}_N||^2}{||h_P||^2}}\\ &= \frac{\frac{||h_P||^2}{\text{SNR}} + \frac{||\bar{h}_N||^2}{\text{SNR}}\left[2\text{SNR}\left(\frac{1+\text{SNR}^{-1}}{\mu_0}\right) - 1\right]}{2\left(\frac{1+\text{SNR}^{-1}}{\mu_0}\right) - 1 + \frac{||\bar{h}_N||^2}{||h_P||^2}}\\ &= \frac{\frac{||h_P||^2}{\text{SNR}}\left[1 + \frac{||\bar{h}_N||^2}{||h_P||^2}(2\frac{\text{SNR}+1}{\mu_0} - 1)\right]}{\left[2\left(\frac{1+\text{SNR}^{-1}}{\mu_0}\right) - 1\right]\left[1 + \frac{||\bar{h}_N||^2}{||h_P||^2}\left(\frac{\mu_0}{2-\mu_0+2\text{SNR}^{-1}}\right)\right]}.\end{aligned} \tag{3.46}$$

Substituting (3.42) and (3.46) in (3.35), we have, finally,

$$\boxed{\eta(M,\mu_0) \approx s_0\frac{1}{M}\left(\frac{||h_M||^2}{||h_P||^2}\right)^2\frac{\left[1 + s_2\frac{||\bar{h}_N||^2}{||h_P||^2}\right]}{\left[1 + s_1\frac{||\bar{h}_N||^2}{||h_P||^2}\right]},} \tag{3.47}$$

where

$$s_0 \triangleq \frac{\mu_0\,(2-\mu_0)\,\mathrm{SNR}\left[\frac{2}{\mu_0}(\mathrm{SNR}+1)-\mathrm{SNR}\right]}{\mathrm{SNR}+1}, \tag{3.48}$$

$$s_1 \triangleq \frac{2}{\mu_0}\,(\mathrm{SNR}+1)-1, \tag{3.49}$$

$$s_2 \triangleq \frac{\mu_0}{2-\mu_0+2\mathrm{SNR}^{-1}} \tag{3.50}$$

are terms that depend exclusively on the SNR and the normalized step size μ_0.

3.3.1 Finding the optimal length

The expression (3.47) of the overall performance is composed by the terms $||h_M||^2$ and $||\bar{h}_N||^2$, which depend on the impulse response and are therefore unknown. However, if we assume that the impulse response has an exponential decay envelope, as described in Section 3.1, we can make reasonable approximations for these terms.

Consider that the impulse response is infinitely long and has exponential decay envelope of the form (3.1). If we approximate $||h_M||^2$ and $||\bar{h}_N||^2$ considering only their exponential envelope, as in (3.2) and (3.3), the overall performance (3.47) becomes

$$\begin{aligned}\eta(M,\mu_0) &\approx \frac{s_0}{M}\left(1-\mathrm{e}^{-2\alpha M}\right)^2\left(\frac{1+s_2\mathrm{e}^{-2\alpha M}}{1+s_1\mathrm{e}^{-2\alpha M}}\right)\\ &= s_0\left[\frac{1+(s_2-2)\mathrm{e}^{-2\alpha M}+(1-2s_2)\mathrm{e}^{-4\alpha M}+s_2\mathrm{e}^{-6\alpha M}}{M+s_1M\mathrm{e}^{-2\alpha M}}\right],\end{aligned} \tag{3.51}$$

which is a transcendental equation in terms of M. In Appendix 3.C, we show how to find M that maximizes (3.51) recursively, by considering M as a continuous variable. We then compute M^o via the auxiliar variable x^o as

$$\boxed{M^o \approx \left\lceil\frac{x^o}{2\alpha}\right\rceil.} \tag{3.52}$$

We round the value to the following integer because larger lengths suffer less undermodeling effects, although the difference of one tap is often negligible.

As shown in Appendix 3.C, the quantity x^o is a function of the SNR and the normalized step size μ_0. Its behavior as a function of μ^o, for different values of the SNR, is depicted in Figure 13. We can also approximate x^o empirically as an explicit function of the SNR and μ_0 as

$$\boxed{x^o \approx -0.265\mu_0^{\mathrm{dB}}+\ln\left(5.65+\exp\left(1.6+0.272\mathrm{SNR}^{\mathrm{dB}}+0.0006\mathrm{SNR}^{\mathrm{dB}}\mu_0^{\mathrm{dB}}\right)\right),} \tag{3.53}$$

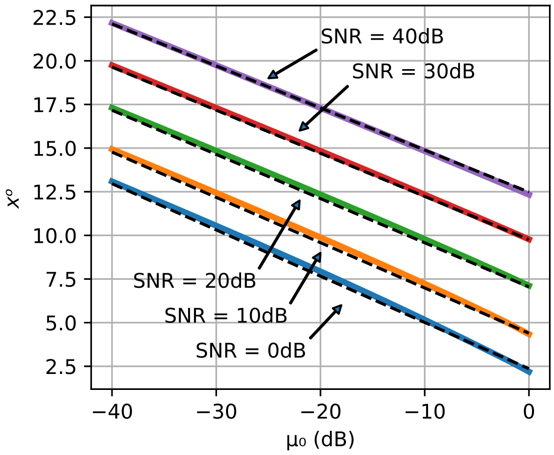


Figure 13: Behavior of $x^o \times \mu_0$ for distinct values of SNR. Solid lines: exact values obtained via (3.81) in Appendix; dash-lines: aproximate values obtained via (3.53).

where $\text{SNR}^{\text{dB}} = 10\log_{10}\text{SNR}$ and $\mu_0^{\text{dB}} = 10\log_{10}\mu_0$ are the quantities in dB. This is a good approximation for values of μ_0^{dB} in the range from -40dB to 0dB, and for values of SNR^{dB} from 0dB to 40dB.

The computation of the optimal length requires some parameters that might be unknown in practice. Directly from (3.52) and (3.53), we must know the SNR and the exponential decay α. Moreover, to compute the actual step size μ, we must compute the optimal step size $\mu^o = 1/\nu$, which, for the ϵ-NLMS algorithm (3.29), also depends on[4] the SNR and α, and for the LMS algorithm (3.23), depends on the SNR, α and σ_u^2. If these parameters are not available for the practitioner, in Appendix 3.D we propose a method to estimate them experimentally, with the aid of an adaptive filter.

The proposed method allows us to design the filter length M, provided some environment parameters are known (or estimated), and given a specified normalized step size μ_0 (and thus the step size μ is also determined). The value of μ_0 may be specified by considering how fast the convergence rate is required. By setting μ_0 closer to one, we obtain faster convergence rate, at the cost of a higher steady-state error. By reducing μ_0, usually in orders of magnitude, we obtain lower steady-state error, but the convergence rate becomes slower. In summary, the proposed design follow the steps:

1. Specify $0 < \mu_0 \leq 1$;
2. Get environment parameters SNR, α and σ_u^2 (if not given, estimate as suggested in Appendix 3.D);
3. Compute the filter length $M = M^o$ via (3.53) and (3.52);

[4] Considering that the term $||h_M||^2$ in (3.23) and (3.29) can be approximated as (3.2).

4. Compute the optimal step size μ^o via (3.23), (3.29) and (3.17);

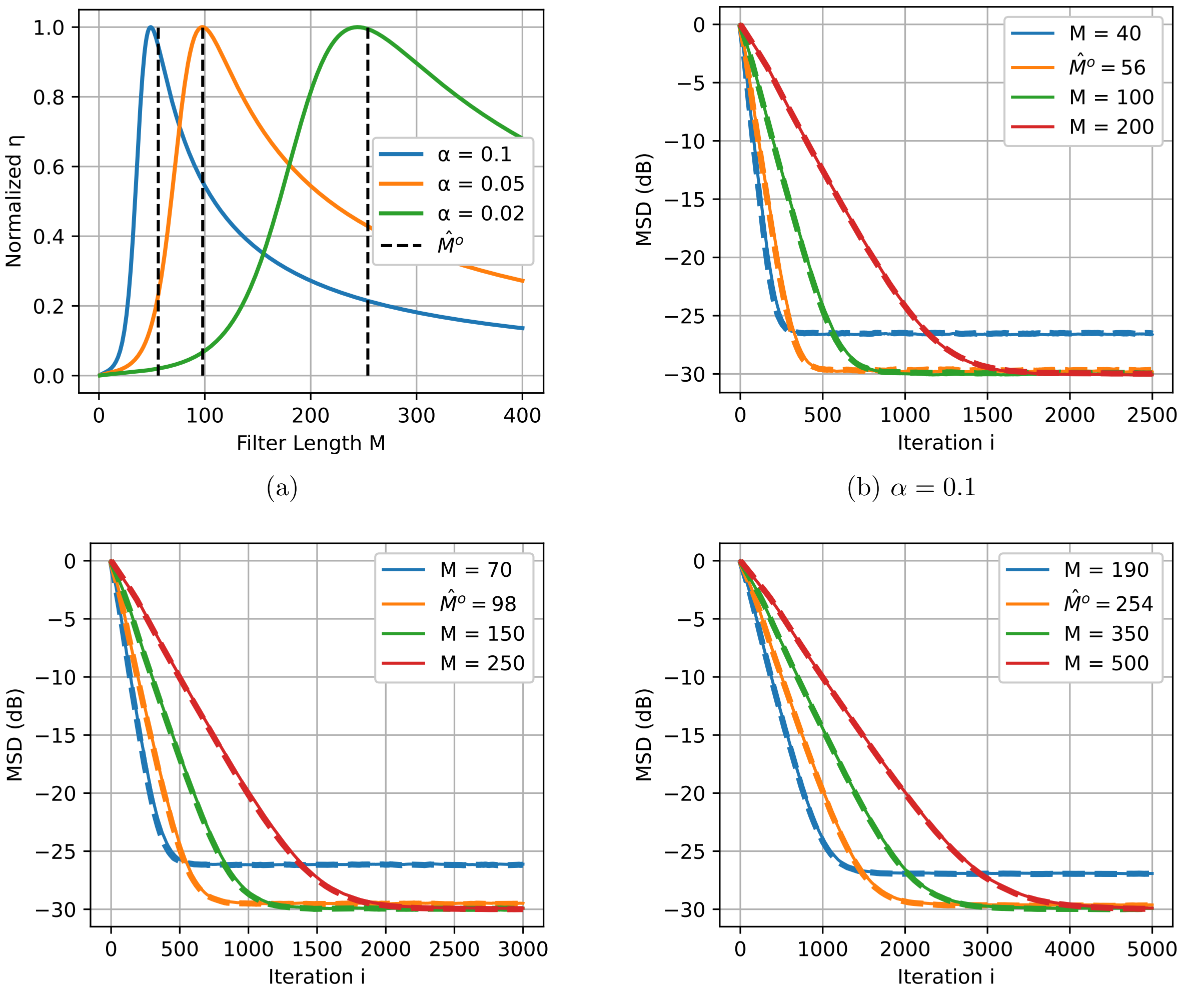
5. Compute the step size μ from relation (3.36).

3.3.2 Simulations

We present simulations to show how a filter with the optimal length performs compared to filters with other lengths. In all the following scenarios, we simulate white input of variance $\sigma_u^2 = 1$, impulse responses with length $P = 500$, that can be regarded as infinite in the simulated scenarios, where $||h_P||^2 = 1$ and $h_b(k)$ are Gaussian i.i.d. sequences. For the ϵ-NLMS algorithm, we use $\epsilon = 0.0001$. The MSD performance curves are obtained as ensemble averages of 1000 runs. We consider estimates $\hat{M}^o$ and $\hat{\mu^o}$ for the optimal length and the optimal step size, respectively, because we compute them via the estimated parameters $\hat{\alpha}$, $\hat{\sigma}_u^2$ and $\hat{\text{SNR}}$ (estimated according to Appendix 3.D).

Figure 14a shows curves $\eta \times M$ for different values of α, where η is computed with the theoretical model (3.51) for exponential envelope approximation. For all curves we considered $\mu_0 = 1$ and $\sigma_v^2 = 0.001$ (SNR = 30dB). In order to fit distinct curves into scale, we normalized η so that the maximum of each curve is one. The estimated optimal lengths $\hat{M}^o$, computed with the estimated parameters, are also shown. As expected from (3.52), α scales the curves along the horizontal axis. Each of the Figures 14b-14d compares the MSD performance curves of filters with distinct lengths for each scenario depicted in Figure 14a. The performance curves for the LMS (solid lines) and for the ϵ-NLMS (dash-lines) algorithms are shown, and note that they are very similar. For each scenario, the filter with optimal length $\hat{M}^o$ exhibits steady-state error very close to filters with larger lengths, but exhibits much faster convergence rate. Any filter of larger length (including the length with which we simulated the “infinitely” long impulse response) would not improve the steady-state performance in a relevant manner, and would only decrease the rate of convergence and increase the computational cost. Filters with length shorter than $\hat{M}^o$ have even faster convergence rate, but their steady-state performance becomes severely degraded.

Figure 15 shows analogous simulations, but comparing different scenarios of SNR. For all curves we considered $\mu_0 = 1$ and $\alpha = 0.05$. Now, note that the curves $\eta \times M$ for different SNR are roughly shifts of the others rather than a rescaling, as in the previous case. Comparing the performance curves for filters of different lengths, we also observe a behavior similar to that of the previous case, with the exception that, as we decrease the SNR, the filter with the optimal length exhibits a noticeably higher steady-state



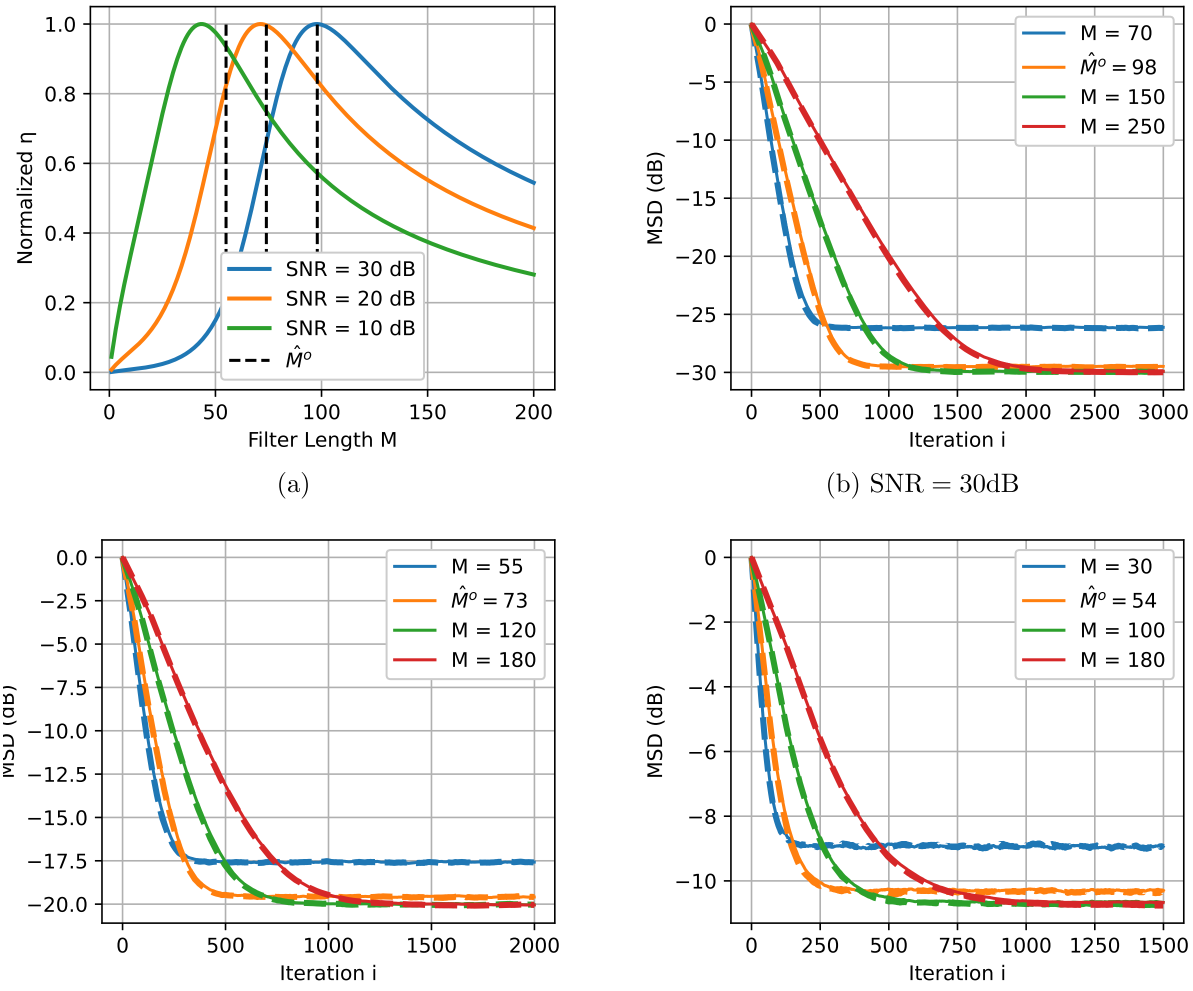
(a)

(b) $\alpha = 0.1$

(c) $\alpha = 0.05$

(d) $\alpha = 0.02$

Figure 14: Simulations for $\mu_0 = 1$, SNR $= 30$dB, and for different values of α: (a) theoretical curves of $\eta \times M$; (b-d) comparing MSD performance curves of filters with different lengths (solid lines: LMS; dash-lines: ϵ-NLMS).



(a)

(b) SNR = 30dB

(c) SNR = 20dB

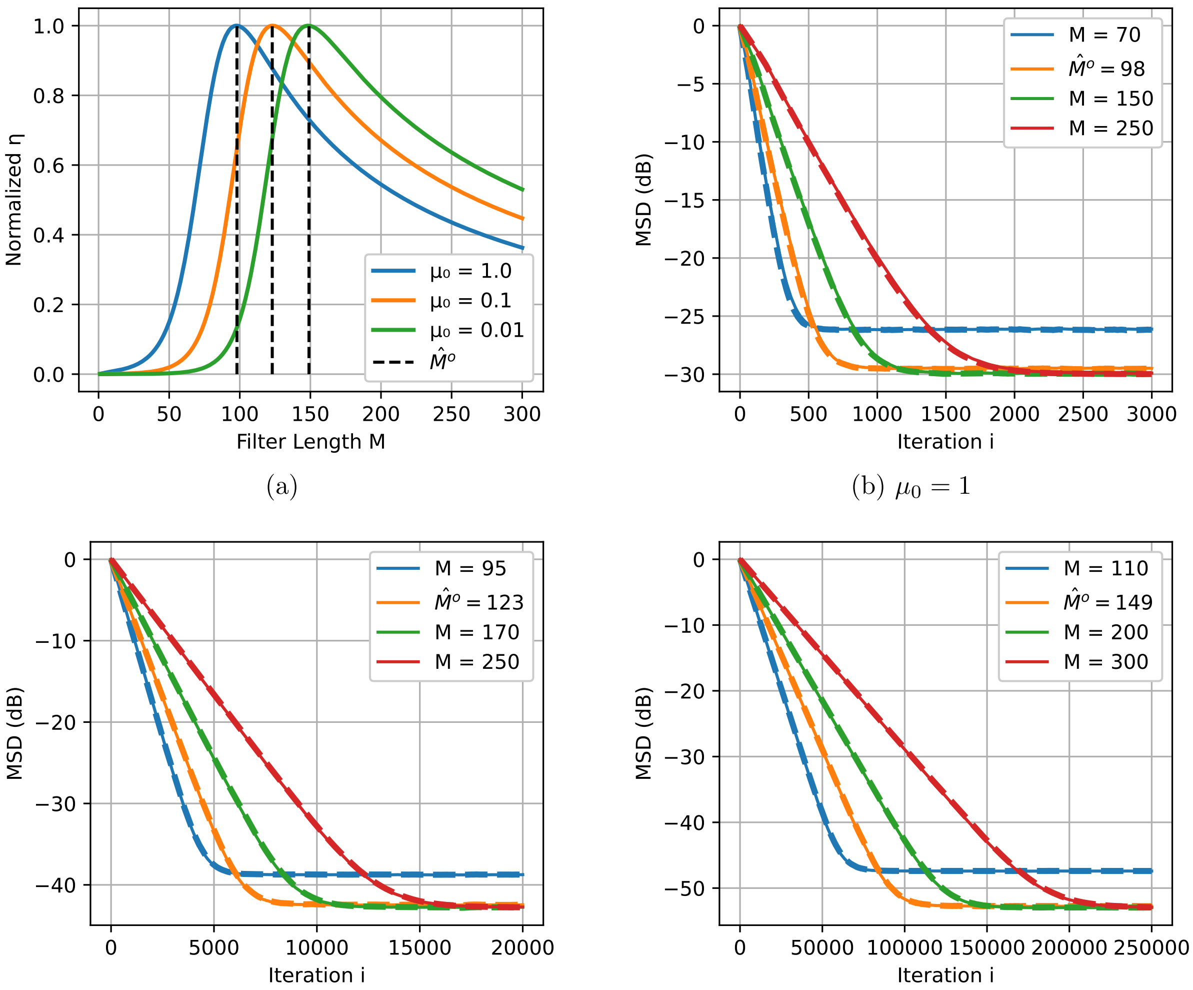
(d) SNR = 10dB

Figure 15: Simulations for $\mu_0 = 1$, $\alpha = 0.05$ and for different values of SNR: (a) theoretical curves of $\eta \times M$; (b-d) comparing MSD performance curves of filters with different lengths (solid lines: LMS; dash-lines: ϵ-NLMS).

error compared to the filters of larger lengths, although it is still a subtle increase when considering the learning curve as a whole.

Figure 16 compares the filter performance when different normalized step sizes μ_0 are used. We fix $\alpha = 0.05$ and SNR $=$ 30dB. We observe, again, that the filter with the optimal length is the one with the fastest convergence rate that keeps the steady-state performance comparable to the performance of longer filters. All in all, from the simulated results, the optimal length seems to optimize, approximately and not in a strict sense, the convergence rate of the algorithm subject to a given steady-state value, that depends on the environment parameters and on the choice of the normalized step size.

Although the optimal length was designed specifically for the case of white input, Figure 17 shows how a filter with such a length behaves in correlated input scenario. The



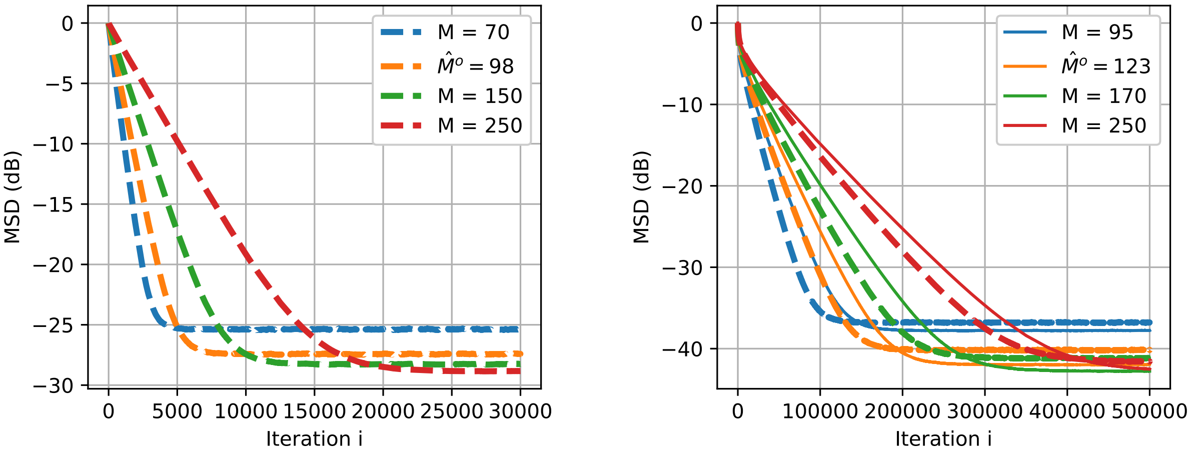
(a)

(b) $\mu_0 = 1$

(c) $\mu_0 = 0.1$

(d) $\mu_0 = 0.01$

Figure 16: Simulations for SNR = 30dB, $\alpha = 0.05$ and for different values of μ_0: (a) theoretical curves of $\eta \times M$; (b-d) comparing MSD performance curves of filters with different lengths (solid lines: LMS; dash-lines: ϵ-NLMS).



(a) $\mu_0 = 1$

(b) $\mu_0 = 0.1$

Figure 17: MSD performance for $\sigma_v^2 = 0.001$, $\alpha = 0.05$, where the input signal is an AR(1) process with parameter $\rho = 0.95$ (correlated input); curves for different values of μ_0 (solid lines: LMS; dash-lines: ϵ-NLMS).

simulations consider the same scenarios of Figures 16b and 16c, now with correlated input signal as an AR(1) process of parameter $\rho = 0.95$. As we use the same design method, the curves $\eta \times M$ would be the same as in Figure 16a. Now, the LMS algorithm no longer converges for $\mu_0 = 1$ and then its curves are not shown. The filter with optimal length still converges faster than filters with other lengths, but now the steady-state has noticeable undermodeling. When the LMS converges, now we can note significant difference between the LMS and the ϵ-NLMS algorithms.

3.4 Length design within a combination

As observed in the simulation results of Subsection 3.3.2, the proposed method chooses an LMS/ϵ-NLMS filter with fast convergence rate while keeping the steady-state performance comparable to the minimum possible for a given environment scenario and a given value of normalized step size. This is particularly interesting when designing the filter with the step size of fastest convergence rate, that is, for unit normalized step size $\mu_0 = 1$. However, a very fast filter will suffer from a very high steady-state error that might be prohibitive in practical applications.

We can correct the steady-state performance if we use such a fast filter within a combination, so that we can ensure a low steady-state error by running a slow filter in parallel. In fact, it is customary to use combinations of filters to combine a fast filter, though with poor steady-state performance, to a slow filter with low steady-state error.

By this point of view, this strategy can also be interpreted as reducing the computational complexity of the combination, since we are taking for the fast filter a filter with length usually smaller than the filter that gives the lowest steady-state error in the combination.

Some works in the literature regard combinations in which the fast filter has shorter length. [65] proposes a shorter filter in a combination of ϵ-NLMS filters, but does not propose any method to design the filter length. [20] proposes a combination in which the length of both filters are iteratively adapted. [66] adjusts recursively the length of the fast filter to track a sparse impulse response.

In this section, we propose, concretely, that the fast filter in a combination, of either LMS or ϵ-NLMS algorithms of fixed length, be designed with the optimal length M^o (3.52) and step size μ^o (3.17). In the simulations that follow in this section, we employ combinations with power normalization (2.30) and the weights feedback scheme (2.32).

3.4.1 Design of the weights feedback cycle length

One of the practical problems when using parallel combinations with weight feedback is the design of the feedback cycle length L. [34] proposed the design of L based on the instant in which the performance curve of the fast filter transit from transient to steady-state. In their derivation, the authors consider the steady-state error as the minimum MSE. This can be a good approximation for relatively small step size and for non-undermodeled filters. However, in our case, the fast filter is undermodeled and has large step size ($\mu_0 = 1$), so the excess MSE in steady-state is no longer negligible. Hence, we extend the method proposed in [34] to include such factors.

For the white input case, the mean-square error is obtained from the mean-square deviation as (from (2.34) and (2.42))

$$\text{MSE}(i) = \sigma_v^2 + \sigma_u^2 E||\tilde{\boldsymbol{w}}_{P,i-1}^M||^2. \tag{3.54}$$

From (3.11), note that MSE(i) has also convergence rate γ. The initial value is MSE$(0) = \sigma_v^2 + \sigma_u^2||h_P||^2 = \sigma_d^2$, that is, the variance of the system output. Using the steady-state MSD (3.24) and (3.30), the steady-state MSE is, from the error relations in Section 2.6,

$$\text{MSE}(\infty) = \left(\sigma_v^2 + \sigma_u^2||\bar{h}_N||^2\right)(1+\varepsilon), \tag{3.55}$$

where $\varepsilon = \frac{\mu M \sigma_u^2}{2-\mu\sigma_u^2 M}$ for the LMS algorithm, and $\varepsilon = \frac{\mu}{2-\mu}$ for the ϵ-NLMS algorithm. As we consider $\mu_0 = 1$, then $\mu = \mu^o$. We choose the feedback cycle length L as the instant in

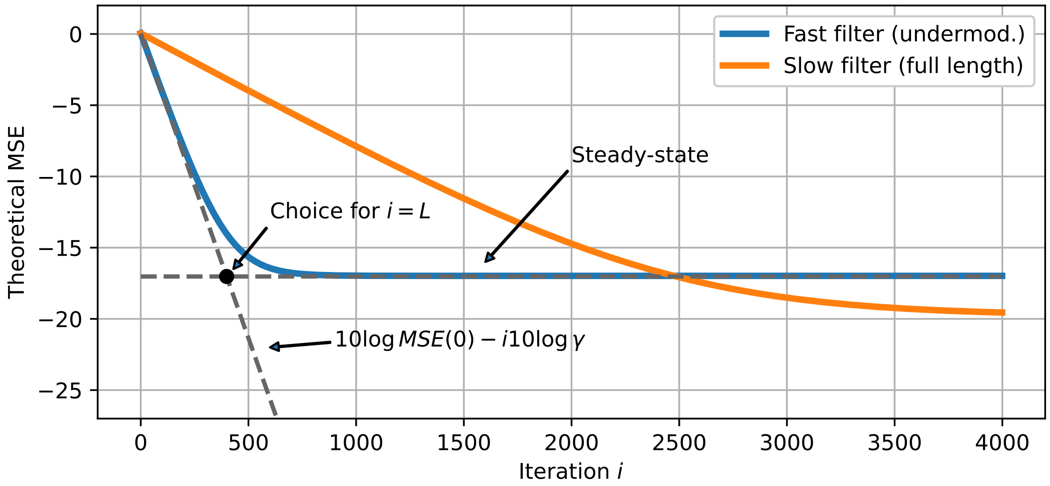


Figure 18: Method proposed to design the weights feedback cycle length.

which the transient of the MSE, approximated by the exponential decay factor γ (a line in log-scale) reaches the steady-state value (3.55), as illustrated in Figure 18. Thus, L is so that

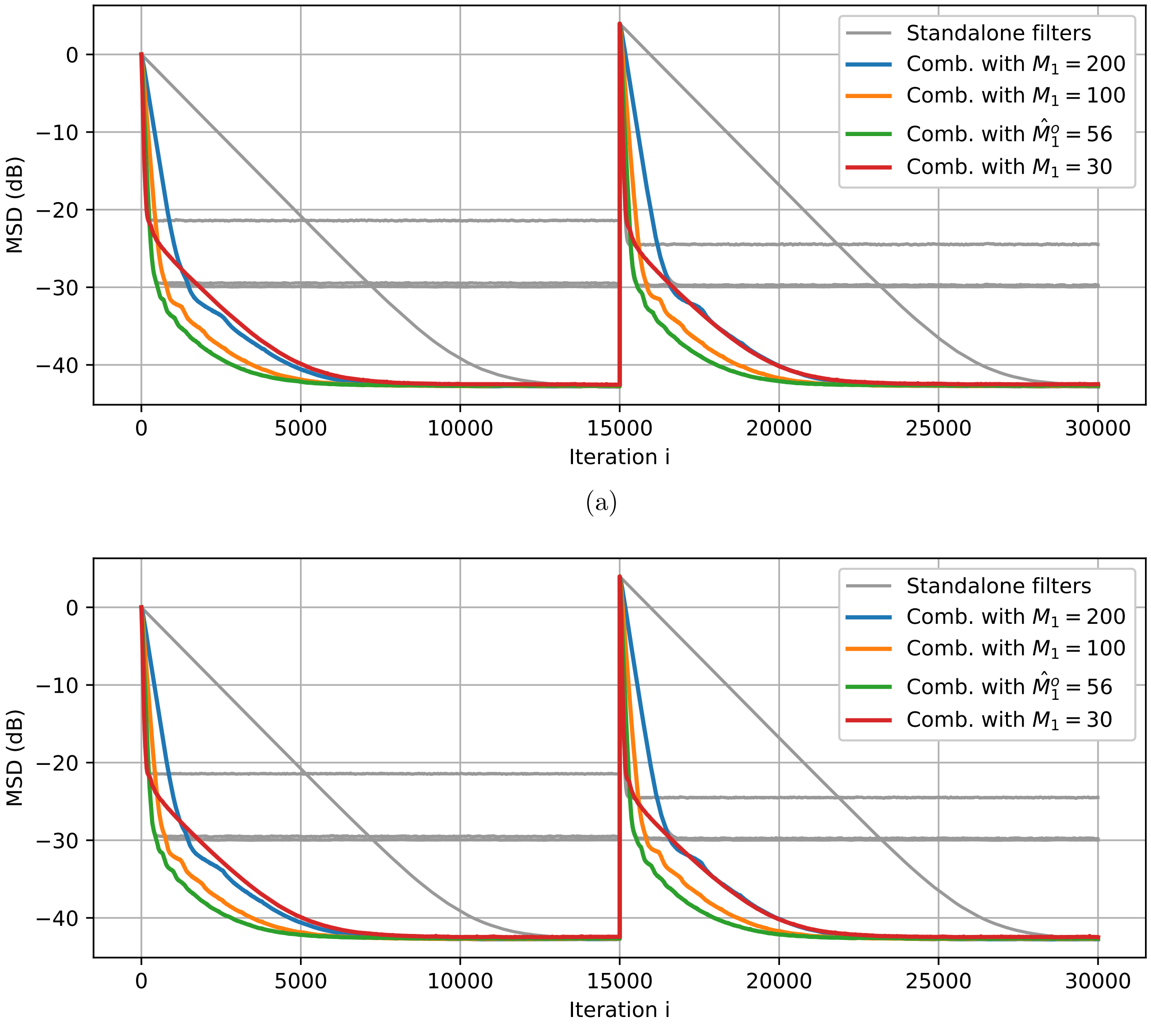
$$10\log(\gamma^L \text{MSE}(0)) = 10\log\left(E\boldsymbol{e}^2(\infty)\right), \tag{3.56}$$

$$L = \frac{\log \text{MSE}(\infty) - \log\sigma_d^2}{\log\gamma}. \tag{3.57}$$

Assuming exponential impulse response, the term $||\bar{h}_N||^2$ can be approximated as in (3.2) and is therefore determined by $||h_P||^2$ and α. The parameters α, σ_d^2, σ_u^2 and σ_v^2, if not available, can be estimated as suggested in Appendix 3.D.

3.4.2 Simulations

Now we present simulated performance curves for the proposed combination of adaptive filters. We consider the same background scenarios of the simulations in Subsection 3.3.2, with parameters $\sigma_u^2 = 1$, $||h_P||^2 = 1$, $h_b(k)$ as Gaussian i.i.d. sequences and $\epsilon = 0.0001$. For each simulated scenario, we compare different combinations of filters, with slow filters of the same length M_2, but with fast filters of distinct lengths M_1. The fast filters run with step size $\mu_1 = \mu_1^o$ (varies with M_1) and the slow filter with $\mu_2 = 0.1\mu_2^o$. The optimal step sizes and the optimal lengths are computed as in the previous simulation, from estimated parameters. In all cases, we set the supervisor parameters $\mu_a = 1$, power normalization of lowpass factor $b = 0.95$, saturation $a_+ = 4$, threshold $\delta = 1$ and, for each combination, the feedback period L is computed as proposed in (3.57). After the learning curves reach steady-state, we introduce an abrupt change into the impulse response, turning it into another Gaussian i.i.d. sequence with the same exponential decay



(a)

(b)

Figure 19: MSD performance of combinations of filters in the case of white input, $\alpha = 0.1$ and SNR= 30 dB, for (a) LMS algorithm and (b) ϵ-NLMS algorithm.

as before, in order to test the reconvergence ability of the algorithm.

Figure 19 depicts the case of impulse response of exponential decay $\alpha = 0.1$ and SNR $= 30$ dB. In Figure 19a, both filters in each combination are LMS filters, and in Figure 19b all filters employ the ϵ-NLMS algorithm. For all combinations, we consider $M_2 = 200$. The combinations perform very similarly with either the LMS or the ϵ-NLMS algorithms. Because of the combination, all curves have the same level of steady-state error. They have, however, distinct overall convergence rates and computational complexities. When reducing the length M_1 of the fast filter down to $M_1 = \hat{M}_1^o$, the combination becomes faster and less complex. The combination with $M = \hat{M}_1^o$ has initial convergence about three times faster and has about 2/3 of the complexity[5] of the

[5] We consider complexity in terms of the total number of weights of the combination. In this case, the combination with $M_1 = 56$ has $200 + 56$ weights, and the combination with $M_1 = 200$ has $200 + 200$

combination with $M_1 = 200$. If we reduce even more the length of the fast filter, the initial convergence is slightly improved, but the overall convergence becomes slower because the fast filter reaches a higher level of steady-state error, and the supervisor makes the slower filter take over earlier.

After the abrupt change, the filters reconverge with similar convergence rate and reach the same steady-state level as in the first convergence. However, there are differences regarding the performance of the weights feedback. First, the cycle length is computed using the undermodeling quantity $\bar{h}_N$, estimated considering only the exponential envelope. The exact value of $\bar{h}_N$, however, varies independently of the exponential envelope, as we can notice from the difference on the steady-state error of the filter with $M_1 = 30$, before and after the abrupt change. Futhermore, in (3.57) we considered that the initial weight error is so that $||\tilde{w}^M_{P,-1}||^2 = ||h_P||^2$, which happens when the filter is initialized in $w_{M,-1} = 0_M$. With the reconvergence, the filter starts converging from some other value of $w_{M,-1}$, which caused the algorithm to anticipate the ideal weight feedback instant.

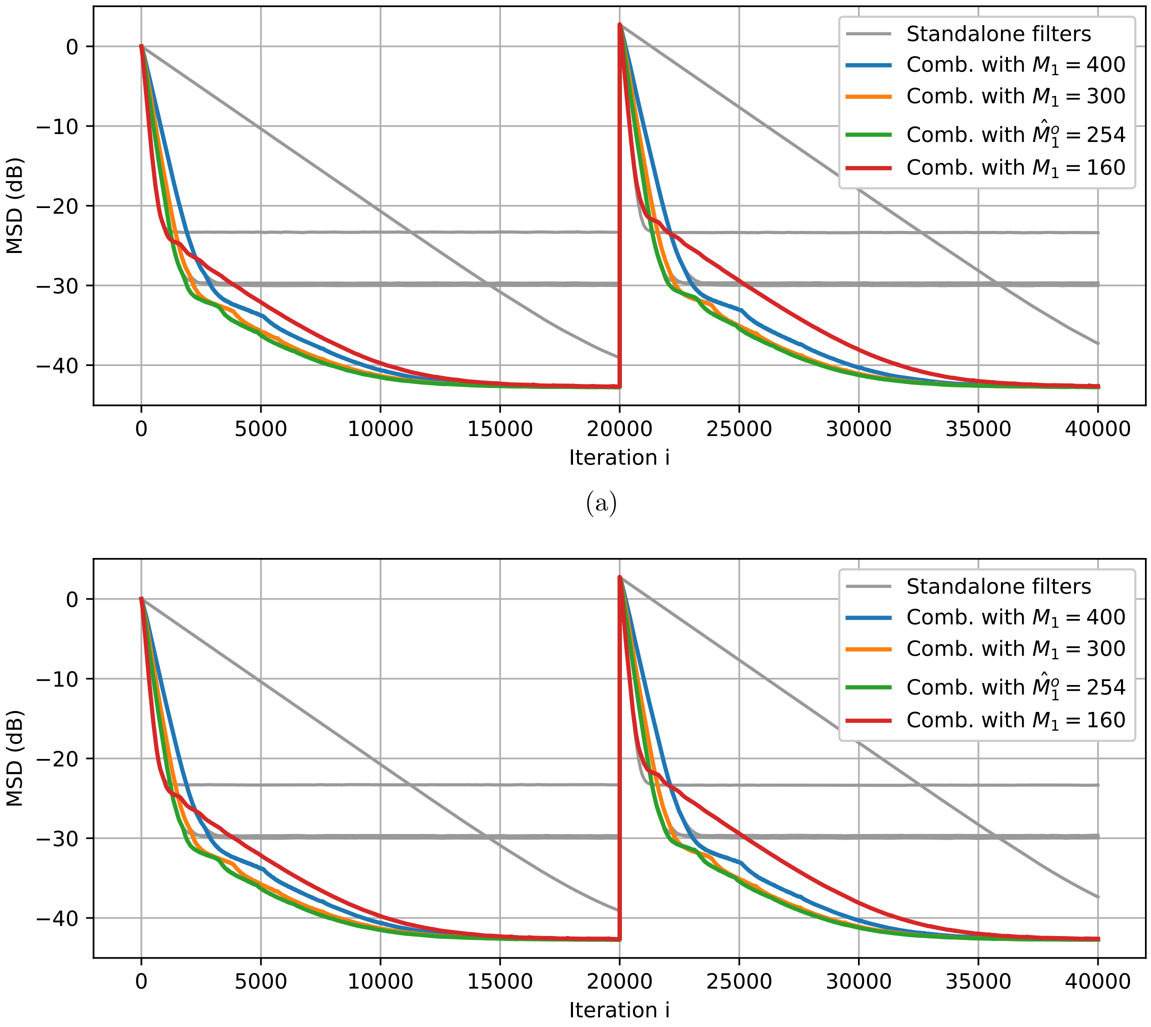
Figure 20 presents analogous plots for the scenario of SNR $=$ 30 dB and smoother impulse response of $\alpha = 0.02$, considering $M_2 = 400$, and Figure 21 presents the plots for low SNR scenario, where SNR $=$ 10 dB, $\alpha = 0.02$ and $M_2 = 300$. In both cases, the behavior of the combinations by varying the length M_1 of the fast filter is similar to the case in Figure 19.

Figure 22 shows the performance curves of the combinations when the input is correlated with $\rho = 0.95$. We consider a scenario similar to the case in Figure 19, with $\alpha = 0.1$, SNR $=$ 30 dB, $M_2 = 200$, but now $\mu_2 = 0.2\mu_2^o$. As in the simulations of Figure 17a, the fast LMS filter does not converge, and thus we show only the curves for the ϵ-NLMS algorithm. The behavior of the combinations in the correlated input scenario is consistent with the previous results for the white input cases.

3.A Computation of fourth-order moments

In this appendix, we compute the fourth-order moment matrix $E\boldsymbol{u}^{\mathrm{T}}_{M,i}\boldsymbol{u}_{M,i}\boldsymbol{u}^{\mathrm{T}}_{M,i}\boldsymbol{u}_{M,i}$ from (3.20), and the the fourth-order moment matrices $E\boldsymbol{u}^{\mathrm{T}}_{M,i}\boldsymbol{u}_{M,i}\boldsymbol{u}^{\mathrm{T}}_{M,i}\bar{\boldsymbol{u}}_{N,i}$ and $E\bar{\boldsymbol{u}}^{\mathrm{T}}_{N,i}\boldsymbol{u}_{M,i}\boldsymbol{u}^{\mathrm{T}}_{M,i}\bar{\boldsymbol{u}}_{N,i}$ from (4.42) (from the following chapter). As in many texts in the literature [9, 35, 58, 59], we derive these quantities by assuming that the input $\boldsymbol{u}(i)$ is a Gaussian process.

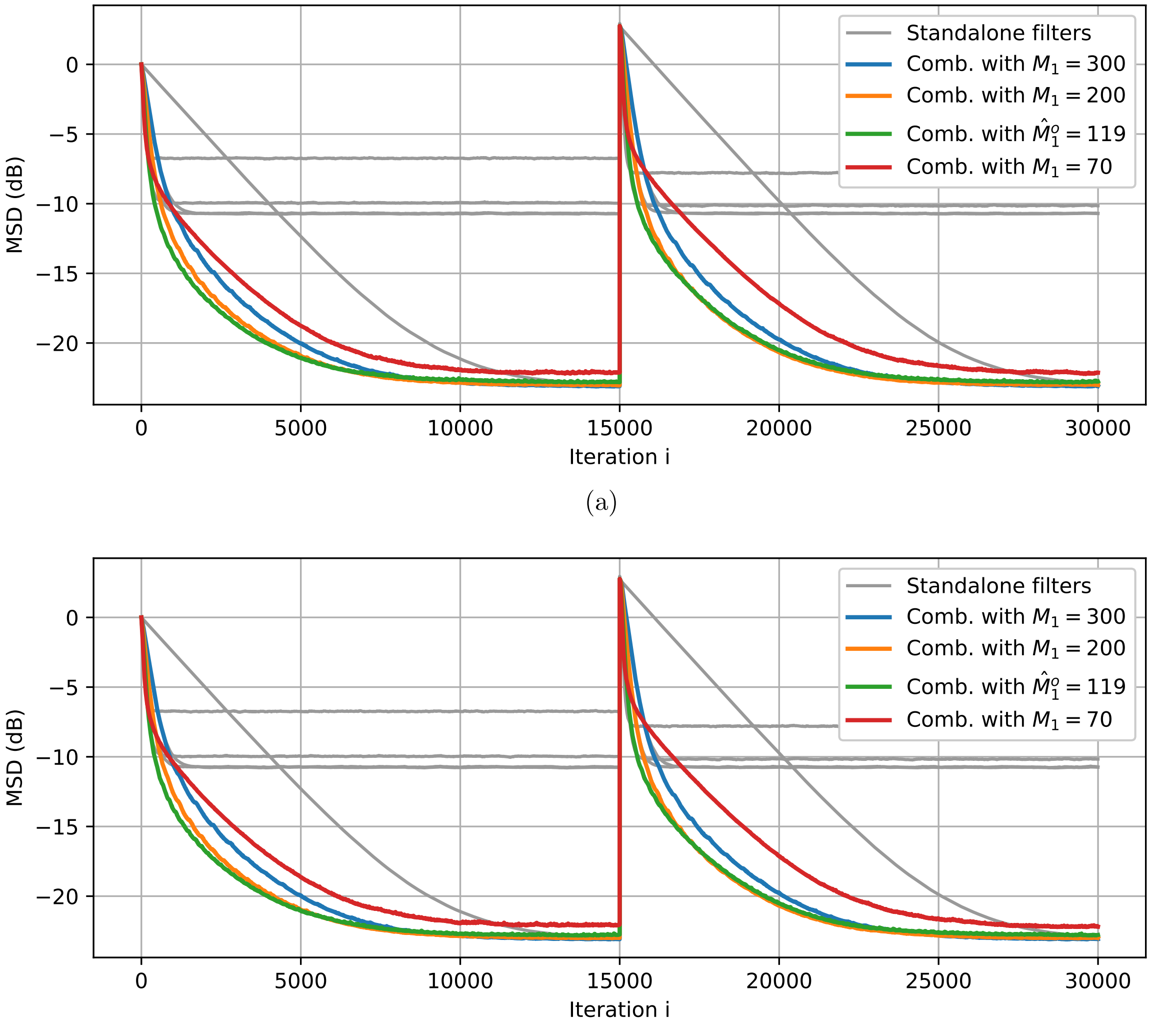
weights.



(a)

(b)

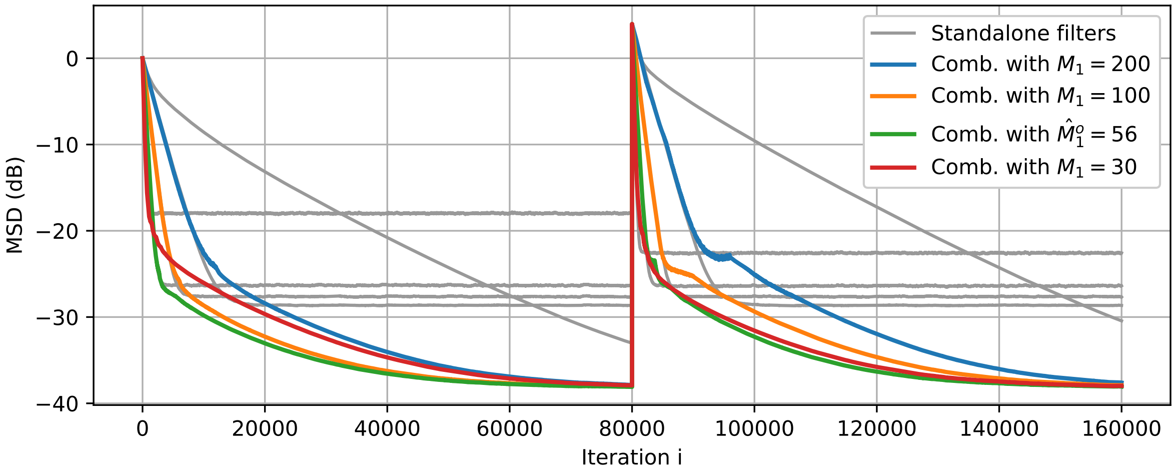
Figure 20: MSD performance of combinations of filters in the case of white input, $\alpha = 0.02$ and SNR= 30 dB, for (a) LMS algorithm and (b) ϵ-NLMS algorithm.



(a)

(b)

Figure 21: MSD performance of combinations of filters in the case of white input, $\alpha = 0.02$ and SNR$=$ 10 dB, for (a) LMS algorithm and (b) ϵ-NLMS algorithm.



(a)

Figure 22: MSD performance of combinations of filters in the case of correlated input $\rho = 0.95$, $\alpha = 0.1$ and SNR$=$ 30 dB, for the ϵ-NLMS algorithm.

Higher-order moments of multivariate Gaussian variables can be computed from second-order Gaussian moments. Let $\boldsymbol{x}_1$, $\boldsymbol{x}_2$, $\boldsymbol{x}_3$ and $\boldsymbol{x}_4$ be zero-mean Gaussian variables that are mutually correlated. From Isserlis' theorem [67], the fourth-order moment $E\boldsymbol{x}_1\boldsymbol{x}_2\boldsymbol{x}_3\boldsymbol{x}_4$ can be computed as

$$E\boldsymbol{x}_1\boldsymbol{x}_2\boldsymbol{x}_3\boldsymbol{x}_4 = E\left[\boldsymbol{x}_1\boldsymbol{x}_2\right]E\left[\boldsymbol{x}_3\boldsymbol{x}_4\right] + E\left[\boldsymbol{x}_1\boldsymbol{x}_3\right]E\left[\boldsymbol{x}_2\boldsymbol{x}_4\right] + E\left[\boldsymbol{x}_1\boldsymbol{x}_4\right]E\left[\boldsymbol{x}_2\boldsymbol{x}_3\right]. \tag{3.58}$$

Now, let us consider Gaussian random vectors[6] $\boldsymbol{x}$, $\boldsymbol{y}$ and $\boldsymbol{z}$, each of them of arbitrary length. We wish to compute the moment matrix

$$X = E\boldsymbol{y}\boldsymbol{x}^{\mathrm{T}}\boldsymbol{x}\boldsymbol{z}^{\mathrm{T}}. \tag{3.59}$$

Matrix X yields all the fourth-order moment matrices in (3.20), (3.27) and (4.42), depending on how we specify the vectors $\boldsymbol{x}$, $\boldsymbol{y}$ and $\boldsymbol{z}$. For[7] $\boldsymbol{x} = \boldsymbol{y} = \boldsymbol{z} = \boldsymbol{u}_{M,i}^{\mathrm{T}}$, X becomes $E\boldsymbol{u}_{M,i}^{\mathrm{T}}\boldsymbol{u}_{M,i}\boldsymbol{u}_{M,i}^{\mathrm{T}}\boldsymbol{u}_{M,i}$. For $\boldsymbol{x} = \boldsymbol{y} = \boldsymbol{u}_{M,i}^{\mathrm{T}}$ and $\boldsymbol{z} = \bar{\boldsymbol{u}}_{N,i}^{\mathrm{T}}$, X becomes $E\boldsymbol{u}_{M,i}^{\mathrm{T}}\boldsymbol{u}_{M,i}\boldsymbol{u}_{M,i}^{\mathrm{T}}\bar{\boldsymbol{u}}_{N,i}$. Finally, for $\boldsymbol{x} = \boldsymbol{u}_{M,i}^{\mathrm{T}}$ and $\boldsymbol{y} = \boldsymbol{z} = \bar{\boldsymbol{u}}_{N,i}^{\mathrm{T}}$, X becomes $E\bar{\boldsymbol{u}}_{N,i}^{\mathrm{T}}\boldsymbol{u}_{M,i}\boldsymbol{u}_{M,i}^{\mathrm{T}}\bar{\boldsymbol{u}}_{N,i}$.

Let us define the autocorrelation matrices

$$R_x = E\boldsymbol{x}\boldsymbol{x}^{\mathrm{T}}, \quad R_y = E\boldsymbol{y}\boldsymbol{y}^{\mathrm{T}}, \quad R_z = E\boldsymbol{z}\boldsymbol{z}^{\mathrm{T}}, \tag{3.60}$$

and the cross-correlation matrices

$$R_{xy} = E\boldsymbol{x}\boldsymbol{y}^{\mathrm{T}}, \quad R_{xz} = E\boldsymbol{x}\boldsymbol{z}^{\mathrm{T}}, \quad R_{yz} = E\boldsymbol{y}\boldsymbol{z}^{\mathrm{T}}. \tag{3.61}$$

Note that the term $\boldsymbol{x}^{\mathrm{T}}\boldsymbol{x} = \sum_i \boldsymbol{x}(i)\boldsymbol{x}(i)$ in (3.59) is a scalar, then it can pre-multiply the whole term. Each element $X(k,\ell)$ can then be written as

$$X(k,\ell) = E\left[\left(\sum_i \boldsymbol{x}(i)\boldsymbol{x}(i)\right)\boldsymbol{y}(k)\boldsymbol{z}(\ell)\right] = \sum_i E\boldsymbol{x}(i)\boldsymbol{x}(i)\boldsymbol{y}(k)\boldsymbol{z}(\ell). \tag{3.62}$$

Using Isserlis' theorem, and rearranging conveniently, we have:

$$\begin{aligned} X(k,\ell) = \sum_i E\left[\boldsymbol{x}(i)\boldsymbol{x}(i)\right]E\left[\boldsymbol{y}(k)\boldsymbol{z}(\ell)\right] + \sum_i E\left[\boldsymbol{x}(i)\boldsymbol{y}(k)\right]E\left[\boldsymbol{x}(i)\boldsymbol{z}(\ell)\right] \\ + \sum_i E\left[\boldsymbol{x}(i)\boldsymbol{z}(\ell)\right]E\left[\boldsymbol{x}(i)\boldsymbol{y}(k)\right]. \end{aligned} \tag{3.63}$$

Note that $\sum_i E\left[\boldsymbol{x}(i)\boldsymbol{x}(i)\right] = \mathrm{Tr}R_x$ is a scalar, and $E\left[\boldsymbol{y}(k)\boldsymbol{z}(\ell)\right]$ is the $\{k,\ell\}$-th element of matrix R_{yz}. The second and the third terms on the right are, respectively, the $\{k,\ell\}$-th

[6] These variable names have no relation to variables of the same name along the text.

[7] Recall that the regressor vectors $\boldsymbol{u}_{M,i}$ exclusively are row vectors.

element of the matrices $R_{xy}^{\mathrm{T}}R_{xz}$ and $\left(R_{xz}^{\mathrm{T}}R_{xy}\right)^{\mathrm{T}} = R_{xy}^{\mathrm{T}}R_{xz}$. Then:

$$X = R_{yz}\mathrm{Tr}R_x + 2R_{xy}^{\mathrm{T}}R_{xz}. \tag{3.64}$$

For the moments in our problem, we have

$$E\boldsymbol{u}_{M,i}^{\mathrm{T}}\boldsymbol{u}_{M,i}\boldsymbol{u}_{M,i}^{\mathrm{T}}\boldsymbol{u}_{M,i} = R_{u,M}\mathrm{Tr}R_{u,M} + 2(R_{u,M})^2, \tag{3.65}$$

$$E\boldsymbol{u}_{M,i}^{\mathrm{T}}\boldsymbol{u}_{M,i}\boldsymbol{u}_{M,i}^{\mathrm{T}}\bar{\boldsymbol{u}}_{N,i} = R_{u,M,N}\mathrm{Tr}R_{u,M} + 2R_{u,M}R_{u,M,N}, \tag{3.66}$$

$$E\bar{\boldsymbol{u}}_{N,i}^{\mathrm{T}}\boldsymbol{u}_{M,i}\boldsymbol{u}_{M,i}^{\mathrm{T}}\bar{\boldsymbol{u}}_{N,i} = R_{u,N}\mathrm{Tr}R_{u,M} + 2R_{u,M,N}^{\mathrm{T}}R_{u,M,N}. \tag{3.67}$$

3.B Aproximations for the moments in the analysis of ϵ-NLMS algorithm

We can analytically approximate the moments $T_{1,M}$, $T_{2,M}$ and $T_{3,M}$ from (3.9), for the ϵ-NLMS algorithm, if we assume that the input signal is white and Gaussian, so that $||\boldsymbol{u}_{M,i}||^2$ can be considered statistically independent from any function of $\boldsymbol{u}_{M,i}$ [10, 59, 64, 68], and considering $\epsilon \approx 0$. Consider a random matrix $\boldsymbol{U}_M$ that depends on $\boldsymbol{u}_{M,i}$ (that later will be used to form matrices $T_{1,M}$, $T_{2,M}$ and $T_{3,M}$). Under the assumptions above:

$$E\boldsymbol{U}_M = E\left[\frac{\boldsymbol{U}_M||\boldsymbol{u}_{M,i}||^2}{||\boldsymbol{u}_{M,i}||^2}\right] \approx E\frac{\boldsymbol{U}_M}{||\boldsymbol{u}_{M,i}||^2}E||\boldsymbol{u}_{M,i}||^2, \tag{3.68}$$

where $E||\boldsymbol{u}_{M,i}||^2 = M\sigma_u^2$, and

$$E\boldsymbol{U}_M = E\left[\frac{\boldsymbol{U}_M\left(||\boldsymbol{u}_{M,i}||^2\right)^2}{\left(||\boldsymbol{u}_{M,i}||^2\right)^2}\right] \approx E\frac{\boldsymbol{U}_M}{\left(||\boldsymbol{u}_{M,i}||^2\right)^2}E\left(||\boldsymbol{u}_{M,i}||^2\right)^2, \tag{3.69}$$

where the scalar moment $E\left(||\boldsymbol{u}_{M,i}||^2\right)^2$ can be written as

$$E\big[\left(\boldsymbol{u}^2(i) + \cdots + \boldsymbol{u}^2(i-M+1)\right)\left(\boldsymbol{u}^2(i) + \cdots + \boldsymbol{u}^2(i-M+1)\right)\big]. \tag{3.70}$$

For every term in the sum on the left, there are $M-1$ terms on the sum on the right that are uncorrelated to it, yielding $M(M-1)$ terms of expected value σ_u^4. Yet, the distributive multiplication yields M 4th-order moments, that, as zero-mean Gaussian variables, have expected value $3\sigma_u^4$. Thus, this moment becomes

$$E\left(||\boldsymbol{u}_{M,i}||^2\right)^2 = M(M-1)\sigma_u^4 + M3\sigma_u^4 = M(M+2)\sigma_u^4. \tag{3.71}$$

For $\boldsymbol{U}_M = \boldsymbol{u}_{M,i}^{\mathrm{T}}\boldsymbol{u}_{M,i}$, we obtain from (3.68)

$$T_{1,M} = E\frac{\boldsymbol{u}_{M,i}^{\mathrm{T}}\boldsymbol{u}_{M,i}}{||\boldsymbol{u}_{M,i}||^2} \approx \frac{E\boldsymbol{u}_{M,i}^{\mathrm{T}}\boldsymbol{u}_{M,i}}{E||\boldsymbol{u}_{M,i}||^2} = \frac{R_{u,M}}{M\sigma_u^2} = \frac{1}{M}I_M. \tag{3.72}$$

Considering $\boldsymbol{U}_M = \boldsymbol{u}_{M,i}^{\mathrm{T}}\boldsymbol{u}_{M,i}$ in (3.69), we obtain

$$T_{2,M} = E\frac{\boldsymbol{u}_{M,i}^{\mathrm{T}}\boldsymbol{u}_{M,i}}{\left(||\boldsymbol{u}_{M,i}||^2\right)^2} \approx \frac{E\boldsymbol{u}_{M,i}^{\mathrm{T}}\boldsymbol{u}_{M,i}}{E\left(||\boldsymbol{u}_{M,i}||^2\right)^2} = \frac{R_{u,M}}{M(M+2)\sigma_u^4} = \frac{1}{M(M+2)\sigma_u^2}I_M, \tag{3.73}$$

and considering $\boldsymbol{U}_M = \boldsymbol{u}_{M,i}^{\mathrm{T}}\boldsymbol{u}_{M,i}\boldsymbol{u}_{M,i}^{\mathrm{T}}\boldsymbol{u}_{M,i}$ in (3.69), and using the 4th-moment result (3.65), we obtain

$$T_{3,M} = E\frac{\boldsymbol{u}_{M,i}^{\mathrm{T}}\boldsymbol{u}_{M,i}\boldsymbol{u}_{M,i}^{\mathrm{T}}\boldsymbol{u}_{M,i}}{\left(||\boldsymbol{u}_{M,i}||^2\right)^2} \approx \frac{E\boldsymbol{u}_{M,i}^{\mathrm{T}}\boldsymbol{u}_{M,i}\boldsymbol{u}_{M,i}^{\mathrm{T}}\boldsymbol{u}_{M,i}}{E\left(||\boldsymbol{u}_{M,i}||^2\right)^2} = \frac{(M+2)\sigma_u^4}{M(M+2)\sigma_u^4} = \frac{1}{M}I_M. \tag{3.74}$$

3.C Recursive computation of the optimum length

In this appendix, we derive M that maximizes $\eta(M,\mu_0)$ given in (3.51), considering exponential impulse response. For convenience of notation, here we write η only as a function of M. Defining

$$\varphi(M) \triangleq 1 + (s_2-2)\mathrm{e}^{-2\alpha M} + (1-2s_2)\mathrm{e}^{-4\alpha M} + s_2\mathrm{e}^{-6\alpha M} \tag{3.75}$$

and

$$\xi(M) \triangleq M + s_1 M\mathrm{e}^{-2\alpha M}, \tag{3.76}$$

where s_1 and s_2, defined in (3.49) and (3.50), respectively, are held as constants. Then,

$$\eta(M) = \frac{s_0\varphi(M)}{\xi(M)}, \tag{3.77}$$

where s_0 is defined in (3.48). Considering M as a continuous variable, the value M^o that maximizes $\eta(M)$ is so that

$$\frac{\partial\eta(M)}{\partial M} = \frac{s_0\left(\varphi'(M)\xi(M) - \varphi(M)\xi'(M)\right)}{\xi^2(M)} = 0, \tag{3.78}$$

where $\varphi'(M)$ and $\xi'(M)$ are, respectively, the derivatives of $\varphi(M)$ and $\xi(M)$ in respect to M. As $\xi(M)$ is a bounded function for finite M, then M^o satisfies

$$\varphi'(M)\xi(M) - \varphi(M)\xi'(M) = 0. \tag{3.79}$$

Substituting (3.75) and (3.76) in (3.79), and defining $x \triangleq 2\alpha M$, we have

$$\begin{aligned} \mathrm{e}^{-x}\big[(2+s_1-s_2)x+(2-s_1-s_2)\big] \\ +\,\mathrm{e}^{-2x}\big[2(2s_2-1)x+2(s_1+s_2)-(1+s_1s_2)\big] \\ -\,\mathrm{e}^{-3x}\big[(2s_1+3s_2+3s_1s_2+2s_1s_2^2)x+s_1s_2(1-2s_2)\big] \\ -\,\mathrm{e}^{-4x}\big[2s_1s_2x+s_1s_2\big]=1. \end{aligned} \tag{3.80}$$

As (3.80) is a transcendental equation on x, we cannot solve it explicitly, and we must compute x recursively. Taking e^{-x} in evidence in (3.80), and taking the natural logarithm of both sides, then the solution x^o to (3.80) can be computed recursively as

$$\begin{aligned} x(k)=\ln\Bigg[&(2+s_1-s_2)x(k-1)+(2-s_1-s_2) \\ &+\mathrm{e}^{-x(k-1)}\big[2(2s_2-1)x(k-1)+2(s_1+s_2)-(1+s_1s_2)\big] \\ &-\mathrm{e}^{-2x(k-1)}\big[(2s_1+3s_2+3s_1s_2+2s_1s_2^2)x(k-1)+s_1s_2(1-2s_2)\big] \\ &-\mathrm{e}^{-3x(k-1)}\big[2s_1s_2x(k-1)+s_1s_2\big]\Bigg], \end{aligned} \tag{3.81}$$

so that $x^o = \lim_{k\to\infty} x(k)$.

3.D Estimating unknown environment parameters

The methods proposed in Sections 3.3 and 3.4 rely on quantities usually unknown about the environment, namely, σ_u^2, σ_v^2, σ_d^2, SNR and α. In this appendix, we show that we can estimate these parameters by observing signals $u(i)$ and $d(i)$ and using the weights of a pilot adaptive filter.

The procedure is divided in three parts. First, we estimate the noise variance σ_v^2 by taking the mean of the squared values of the system output samples, provided that the input of the system is null (from (2.5), when $u_{P,i} = 0_P$ the output is solely the noise $v(i)$). We can acquire null input by either turning the input down, if possible, or acquiring the input during silent tracks. As the signal $v(i)$ is an i.i.d. sequence, the standard deviation of the estimation is, when using K samples, $\sigma_v^2\sqrt{2/K}$ [69].

Second, we estimate σ_u^2 and σ_d^2 by measuring the mean-square values of the system input and output, respectively, now considering that the system is operating with integral input. From the measures so far, we can estimate the SNR from

$$\frac{\sigma_d^2}{\sigma_v^2} = \frac{\sigma_v^2 + \sigma_u^2 ||h_P||^2}{\sigma_v^2} = 1 + \mathrm{SNR}. \tag{3.82}$$

Finally, we estimate the exponential decay α of the system impulse response with the aid of an adaptive filter. We measure an approximate impulse response from the filter weights $w(k)$, for $k = 0, \ldots, M-1$, at steady-state. The filter length M can be chosen loosely, as long as it enables the filter to capture the decay behavior, and the excess mean-square error in steady-state must be reasonably low (using, for example, $\mu = 0.1\mu^o$). We estimate α by minimizing the squared error between $|w(k)|$ and the exponential model $\beta \mathrm{e}^{-\alpha k}$ in logarithm scale, for $k = 0, \ldots, M-1$, as [69]

$$\begin{bmatrix} \beta \\ -\alpha \end{bmatrix} = \left(X^{\mathrm{T}} X\right)^{-1} X^{\mathrm{T}} y, \tag{3.83}$$

where

$$X = \begin{bmatrix} 1 & 0 \\ 1 & 1 \\ \vdots & \vdots \\ 1 & M-1 \end{bmatrix}, \quad y = \begin{bmatrix} \log |w(0)| \\ \log |w(1)| \\ \vdots \\ \log |w(M-1)| \end{bmatrix}. \tag{3.84}$$

4 ANALYSIS OF UNDERMODELED FILTERS IN NONSTATIONARY SCENARIO

We have studied so far in this text the cases in which the underlying environment is stationary. The model of a time-invariant impulse response, however, may not be representative in many practical applications where the environment is constantly changing, such as in acoustic paths [70] and in communication channels [47, 71].

In this chapter, we derive the tracking performance, which is the steady-state performance under time-variant systems, for the undermodeled LMS and ϵ-NLMS algorithms. Although the analyses of these algorithms for nonstationary models (time-variant impulse responses) [2, 3, 35, 72, 73] and for undermodeled filters [9, 10] are separately available in the literature, the joint analysis as we derive is a novelty, and, as we will show, simultaneous nonstationarity and undermodeling yields terms that are not trivially obtained by separate analysis.

Since we focus on the analysis of steady-state performance, we opt to use the energy conservation relation [35] in the derivations. We employ the impulse response model (2.6), that considers general mean h_P and decay rate c. Many authors approximate such decay rate c to 1 in the algorithm analyses [2, 4, 35], but we keep $|c| < 1$ in our derivations since an unstable mean-square behavior in the impulse response is not suitable to analyze undermodeled filters.

We also derive the optimal step size under such time-variant model for the white input case. Differently from what we consider in the previous chapter, we now consider the optimal step size as the one that minimizes the steady-state EMSE. We show that such optimal step size may not exist, and we derive the condition for its existence.

This chapter is organized as follow: in Section 4.1, we carry out a preliminar analysis for the mean and mean-square behavior of the impulse response model (2.6); in Section 4.2, we derive the mean behavior of the general data-normalized algorithm, and the mean-square behavior in steady-state using the energy conservation relation; we derive the results, in particular, for the LMS and ϵ-NLMS algorithms; in Section 4.3, we derive the step size that minimizes the steady-state EMSE when the input signal is white; finally, in Section 4.4, we present simulated curves to better understand the theoretical results obtained, and to compare with experimental data.

4.1 Behavior of the impulse response

Before we begin the algorithm performance analysis, we need to derive the mean and the mean-square behavior of the impulse response model (2.6).

First, let us analyze the mean and mean-square behaviors of the internal state $\boldsymbol{\theta}_{P,i}$. Recall, from the second equation in (2.6), that

$$\boldsymbol{\theta}_{P,i} = c\boldsymbol{\theta}_{P,i-1} + \boldsymbol{q}_{P,i}, \tag{4.1}$$

where $|c| < 1$, and $\boldsymbol{q}_{P,i}$ is as an zero-mean i.i.d. process with covariance matrix $E\boldsymbol{q}_{P,i}\boldsymbol{q}_{P,i}^{\mathrm{T}} = Q_P$. As $\boldsymbol{q}_{P,i}$ is zero-mean, we have $E\boldsymbol{\theta}_{P,\infty} = 0_P$.

Define the covariance matrix of $\boldsymbol{\theta}_{P,i}$ as $\Theta_{P,i} \triangleq E\boldsymbol{\theta}_{P,i}\boldsymbol{\theta}_{P,i}^{\mathrm{T}}$. Then, using (4.1), we can write it as

$$\begin{aligned}\Theta_{P,i} &\triangleq E\boldsymbol{\theta}_{P,i}\boldsymbol{\theta}_{P,i}^{\mathrm{T}} = E(c\boldsymbol{\theta}_{P,i-1} + \boldsymbol{q}_{P,i})(c\boldsymbol{\theta}_{P,i-1} + \boldsymbol{q}_{P,i})^{\mathrm{T}} \\ &= c^2 E\boldsymbol{\theta}_{P,i-1}\boldsymbol{\theta}_{P,i-1}^{\mathrm{T}} + cE\boldsymbol{\theta}_{P,i-1}\boldsymbol{q}_{P,i}^{\mathrm{T}} + cE\boldsymbol{q}_{P,i}\boldsymbol{\theta}_{P,i-1}^{\mathrm{T}} + E\boldsymbol{q}_{P,i}\boldsymbol{q}_{P,i}^{\mathrm{T}} \\ &= c^2\Theta_{P,i-1} + Q_P, \end{aligned} \tag{4.2}$$

and, as $i \to \infty$,

$$\Theta_{P,\infty} = \frac{1}{1-c^2} Q_P. \tag{4.3}$$

Note that $E||\boldsymbol{\theta}_{P,i}||^2 = \mathrm{Tr}\Theta_{P,i}$, and similarly, as $i \to \infty$,

$$\mathrm{Tr}\Theta_{P,\infty} = \frac{\mathrm{Tr}Q_P}{1-c^2}. \tag{4.4}$$

From the first equation in (2.6), the impulse response $\boldsymbol{h}_{P,i}$ is obtained from the internal state simply by adding the deterministic term h_P, as

$$\boldsymbol{h}_{P,i} = h_P + \boldsymbol{\theta}_{P,i}, \tag{4.5}$$

and so the mean and mean-square behaviors of $\boldsymbol{h}_{P,i}$ are straightforward. The mean behavior is

$$E\boldsymbol{h}_{P,i} = h_P + E\boldsymbol{\theta}_{P,i}, \tag{4.6}$$

and $E\boldsymbol{h}_{P,\infty} = h_P$. As $\boldsymbol{h}_{P,i}$ differs from $\boldsymbol{\theta}_{P,i}$ only by the mean, the covariance matrix $\boldsymbol{h}_{P,i}$ equals the covariance matrix $\Theta_{P,i}$:

$$E(\boldsymbol{h}_{P,i} - h_P)(\boldsymbol{h}_{P,i} - h_P)^{\mathrm{T}} = E\boldsymbol{\theta}_{P,i}\boldsymbol{\theta}_{P,i}^{\mathrm{T}} = \Theta_{P,i}. \tag{4.7}$$

The correlation matrix of $\boldsymbol{h}_{P,i}$ is

$$\begin{aligned} E\boldsymbol{h}_{P,i}\boldsymbol{h}_{P,i}^{\mathrm{T}} &= E\left(h_P + \boldsymbol{\theta}_{P,i}\right)\left(h_P + \boldsymbol{\theta}_{P,i}\right)^{\mathrm{T}} \\ &= h_P h_P^{\mathrm{T}} + \Theta_{P,i} + h_P E\boldsymbol{\theta}_{P,i}^{\mathrm{T}} + E\boldsymbol{\theta}_{P,i} h_P^{\mathrm{T}} \\ &= h_P h_P^{\mathrm{T}} + \Theta_{P,i}. \end{aligned} \tag{4.8}$$

As $i \to \infty$, using (4.3),

$$E\boldsymbol{h}_{P,\infty}\boldsymbol{h}_{P,\infty}^{\mathrm{T}} = h_P h_P^{\mathrm{T}} + \frac{1}{1-c^2} Q_P, \tag{4.9}$$

and similarly

$$E||\boldsymbol{h}_{P,\infty}||^2 = ||h_P||^2 + \frac{\mathrm{Tr} Q_P}{1-c^2}. \tag{4.10}$$

We assume that, as the adaptive filter begins to run at $i = 0$, the impulse response has already reached steady-state, that is, we assume that the environment model above starts to run at some instant distant in the past.

Else, note that, for $|c| < 1$, the quantities in steady-state (4.3), (4.4) and (4.10) are finite real numbers. For $|c| = 1$, on the other hand, these quantities grow unboundedly. As seen in the analyses of the previous chapter, the performance of undermodeled algorithms depends on how the energy of the impulse response is distributed along the taps, and we show in this chapter that this is also true for the nonstationary case. Therefore, in order to take only the cases in which the algorithm performance is stable, we consider that $|c| < 1$.

4.1.1 Notes about the covariance matrix Q_P

In the definitions of $\boldsymbol{q}_{P,i}$, stated back in Subsection 2.2.2 and reminded above in this section, we only specified the temporal property for $\boldsymbol{q}_{P,i}$ — an i.i.d. sequence. This, however, does not tell us anything about matrix Q_P, that defines how the elements of the vector $\boldsymbol{q}_{P,i}$ are correlated to each other at a given time instant.

In the literature, tracking analyses are carried out assuming a general matrix Q_P [2, 35]. In the analysis that follows in this text, for the case of undermodeled filters, we make two simplifying assumptions about the elements of $\boldsymbol{q}_{P,i}$: (i) each element of the vector $\boldsymbol{q}_{P,i}$ is independent of the others, so that matrix Q_P is a diagonal matrix; and (ii) the variance of elements in $\boldsymbol{q}_{P,i}$ follows an exponential envelope of decay α. Under these assumptions, the elements of $\boldsymbol{q}_{P,i}$ can be written as

$$\boldsymbol{q}_{P,i}(k) = e^{-\alpha k}\boldsymbol{z}_{P,i}(k), \text{ for } k = 0, 1, \ldots, P-1, \tag{4.11}$$

where $\boldsymbol{z}_{P,i}(k)$ has variance σ_z^2 and is independent of $\boldsymbol{z}_{P,j}(\ell)$ if either $i \neq j$ or $k \neq \ell$. With this considered, the trace of Q_P is given by

$$\text{Tr}Q_P = \sum_{k=0}^{P-1} e^{-2\alpha k} E\boldsymbol{z}_{P,i}^2(k) = \sigma_z^2 \left(\frac{1-e^{-2\alpha P}}{1-e^{-2\alpha}}\right). \tag{4.12}$$

The exponential envelope assumption for $\boldsymbol{q}_{P,i}(k)$, together with a similar assumption for the mean h_P, leads $\boldsymbol{h}_{P,i}$ to have an exponential envelope. As observed in the previous chapter, an exponential impulse response is meaningful in many practical applications and provides a mathematical parameter to evaluate how undermodeling affects the performance of an adaptive algorithm.

Assuming that the elements in $\boldsymbol{q}_{P,i}$ are independent from one another implies that the entries $\boldsymbol{\theta}_{P,i}(k)$, for $k = 0, 1, \ldots, P-1$, are independent processes (and so are the entries of $\boldsymbol{h}_{P,i}$). This induces a reasonable statistical simplification when considering the partitions of these vector quantities, as we address in the following subsection.

4.1.2 Vector partitions

In the analyses of undermodeled filters, it is necessary to split the impulse response vector into two parts, as in the previous chapters. Similarly to (2.11), we define, for the nonstationary case,

$$\boldsymbol{q}_{P,i} = \begin{bmatrix} \boldsymbol{q}_{M,i} \\ \bar{\boldsymbol{q}}_{N,i} \end{bmatrix}, \quad \boldsymbol{\theta}_{P,i} = \begin{bmatrix} \boldsymbol{\theta}_{M,i} \\ \bar{\boldsymbol{\theta}}_{N,i} \end{bmatrix}, \quad \boldsymbol{h}_{P,i} = \begin{bmatrix} \boldsymbol{h}_{M,i} \\ \bar{\boldsymbol{h}}_{N,i} \end{bmatrix}, \tag{4.13}$$

where $\boldsymbol{q}_{M,i}, \boldsymbol{\theta}_{M,i}, \boldsymbol{h}_{M,i} \in \mathbb{R}^{M\times 1}$ and $\bar{\boldsymbol{q}}_{N,i}, \bar{\boldsymbol{\theta}}_{N,i}, \bar{\boldsymbol{h}}_{N,i} \in \mathbb{R}^{N\times 1}$. Considering the independence assumptions from Subsection 4.1.1, none of the quantities $\boldsymbol{q}_{M,i}$, $\boldsymbol{\theta}_{M,i}$ and $\boldsymbol{h}_{M,i}$ are statistically dependent of any of the quantities $\bar{\boldsymbol{q}}_{N,i}$, $\bar{\boldsymbol{\theta}}_{N,i}$ and $\bar{\boldsymbol{h}}_{N,i}$.

Consequently, the mean and mean-square behavior of these vector partitions are simply the partition of the results from the previous subsection. Vectors $\boldsymbol{q}_{M,i}$, $\bar{\boldsymbol{q}}_{N,i}$, $\boldsymbol{\theta}_{M,i}$ and $\bar{\boldsymbol{\theta}}_{N,i}$ are zero-mean, and $E\boldsymbol{h}_{M,i} = h_M$ and $E\bar{\boldsymbol{h}}_{N,i} = \bar{h}_N$, where $h_P = \begin{bmatrix} h_M^{\text{T}} & \bar{h}_N^{\text{T}} \end{bmatrix}^{\text{T}}$.

For the mean-square quantities, define $Q_M \triangleq E\boldsymbol{q}_{M,i}\boldsymbol{q}_{M,i}^{\mathrm{T}}$ and $\bar{Q}_N \triangleq E\bar{\boldsymbol{q}}_{N,i}\bar{\boldsymbol{q}}_{N,i}^{\mathrm{T}}$. Then:

$$\begin{aligned}\mathrm{Tr}Q_P = E\boldsymbol{q}_{P,i}^{\mathrm{T}}\boldsymbol{q}_{P,i} &= E\begin{bmatrix}\boldsymbol{q}_{M,i}^{\mathrm{T}} & \bar{\boldsymbol{q}}_{N,i}^{\mathrm{T}}\end{bmatrix}\begin{bmatrix}\boldsymbol{q}_{M,i} \\ \bar{\boldsymbol{q}}_{N,i}\end{bmatrix} \\ &= E\boldsymbol{q}_{M,i}^{\mathrm{T}}\boldsymbol{q}_{M,i} + E\bar{\boldsymbol{q}}_{N,i}^{\mathrm{T}}\bar{\boldsymbol{q}}_{N,i} \\ &= \mathrm{Tr}Q_M + \mathrm{Tr}\bar{Q}_N. \end{aligned} \tag{4.14}$$

Analogously, if we define $\Theta_{M,i} \triangleq E\boldsymbol{\theta}_{M,i}\boldsymbol{\theta}_{M,i}^{\mathrm{T}}$ and $\bar{\Theta}_{N,i} \triangleq E\bar{\boldsymbol{\theta}}_{N,i}\bar{\boldsymbol{\theta}}_{N,i}^{\mathrm{T}}$, then

$$\mathrm{Tr}\Theta_{P,i} = \mathrm{Tr}\Theta_{M,i} + \mathrm{Tr}\bar{\Theta}_{N,i}, \tag{4.15a}$$

$$E||\boldsymbol{h}_{P,i}||^2 = E||\boldsymbol{h}_{M,i}||^2 + E||\bar{\boldsymbol{h}}_{N,i}||^2, \tag{4.15b}$$

and, in steady-state,

$$\mathrm{Tr}\Theta_{M,\infty} = \frac{\mathrm{Tr}Q_M}{1-c^2}, \tag{4.16a}$$

$$\mathrm{Tr}\bar{\Theta}_{N,\infty} = \frac{\mathrm{Tr}\bar{Q}_N}{1-c^2}. \tag{4.16b}$$

If we assume that elements of the vector $\boldsymbol{q}_{P,i}$ are independent, under the exponential model (4.11), then

$$\mathrm{Tr}Q_M = \sigma_z^2\left(\frac{1-e^{-2\alpha M}}{1-e^{-2\alpha}}\right), \tag{4.17a}$$

$$\mathrm{Tr}\bar{Q}_N = \sigma_z^2\left(\frac{e^{-2\alpha M}-e^{-2\alpha P}}{1-e^{-2\alpha}}\right) = \sigma_z^2 e^{-2\alpha M}\left(\frac{1-e^{-2\alpha N}}{1-e^{-2\alpha}}\right). \tag{4.17b}$$

4.2 Tracking performance analysis

In this section, we derive the excess MSE for the nonstationary environment. We carry out the derivations considering a data-normalized algorithm, and then derive the particular cases for the LMS and the ϵ-NLMS algorithms, similarly to what is done in Section 3.2. We begin the analysis by deriving some fundamental relations and assumptions; we then derive the mean behavior, and finally we derive the mean-square performance by using the energy conservation relation.

The full and partial a priori estimation errors $\boldsymbol{e}_a(i)$ and $\boldsymbol{e}_{a,M}(i)$, defined respectively in (2.38) and (2.45), are related as

$$\begin{aligned}\boldsymbol{e}_a(i) &= \boldsymbol{u}_{P,i}\left(\boldsymbol{h}_{P,i} - \begin{bmatrix}\boldsymbol{w}_{M,i-1}\\ 0_N\end{bmatrix}\right) = \begin{bmatrix}\boldsymbol{u}_{M,i} & \bar{\boldsymbol{u}}_{N,i}\end{bmatrix}\begin{bmatrix}\boldsymbol{h}_{M,i} - \boldsymbol{w}_{M,i-1}\\ \bar{\boldsymbol{h}}_{N,i}\end{bmatrix}\\ &= \boldsymbol{e}_{a,M}(i) + \bar{\boldsymbol{u}}_{N,i}\bar{\boldsymbol{h}}_{N,i}. \end{aligned} \tag{4.18}$$

Recall from (2.40) and (2.44) that the weight error vectors consider impulse response and filter weight of the same time instant. Using (4.1) and (4.5), we can relate the expression $\boldsymbol{h}_{M,i} - \boldsymbol{w}_{M,i-1}$ to the weight error vector $\tilde{\boldsymbol{w}}_{M,i-1}$ as

$$\begin{aligned}\boldsymbol{h}_{M,i} - \boldsymbol{w}_{M,i-1} &= h_M + c\boldsymbol{\theta}_{M,i-1} + \boldsymbol{q}_{M,i} - \boldsymbol{w}_{M,i-1}\\ &= \underbrace{h_M + \boldsymbol{\theta}_{M,i-1}}_{\boldsymbol{h}_{M,i-1}} + (c-1)\boldsymbol{\theta}_{M,i-1} + \boldsymbol{q}_{M,i} - \boldsymbol{w}_{M,i-1}\\ &= \tilde{\boldsymbol{w}}_{M,i-1} + \boldsymbol{q}_{M,i} - (1-c)\boldsymbol{\theta}_{M,i-1}. \end{aligned} \tag{4.19}$$

The filter output error is given by

$$\begin{aligned}\boldsymbol{e}(i) &= \boldsymbol{d}(i) - \boldsymbol{u}_{M,i}\boldsymbol{w}_{M,i-1}\\ &= \boldsymbol{v}(i) + \boldsymbol{u}_{M,i}\boldsymbol{h}_{M,i} + \bar{\boldsymbol{u}}_{N,i}\bar{\boldsymbol{h}}_{N,i} - \boldsymbol{u}_{M,i}\boldsymbol{w}_{M,i-1}\\ &= \boldsymbol{v}(i) + \boldsymbol{u}_{M,i}(\boldsymbol{h}_{M,i} - \boldsymbol{w}_{M,i-1}) + \bar{\boldsymbol{u}}_{N,i}\bar{\boldsymbol{h}}_{N,i}\\ &= \boldsymbol{v}(i) + \boldsymbol{e}_{a,M}(i) + \bar{\boldsymbol{u}}_{N,i}\bar{\boldsymbol{h}}_{N,i}\\ &= \boldsymbol{v}(i) + \boldsymbol{e}_a(i). \end{aligned} \tag{4.20}$$

For the analysis that follows, we consider that $\boldsymbol{u}_{P,i}$ is independent of $\boldsymbol{u}_{P,j}$ for $i \neq j$ [1,4,35]. Summarizing our independence assumptions, we have that:

1. Random processes $\boldsymbol{u}_{P,i}$, $\boldsymbol{v}(i)$ and $\boldsymbol{q}_{P,i}$ are mutually independent;

2. $\boldsymbol{u}_{P,i}$, $\boldsymbol{v}(i)$ and $\boldsymbol{q}_{P,i}$ are i.i.d. processes;

3. $\boldsymbol{h}_{P,i}$ and $\boldsymbol{\theta}_{P,i}$ are independent of the processes $\boldsymbol{u}_{P,i}$ and $\boldsymbol{v}(i)$; they are also independent of $\boldsymbol{q}_{P,j}$, but only for $i < j$;

4. $\tilde{\boldsymbol{w}}^M_{P,i}$ is independent of $\boldsymbol{v}(j)$, $\boldsymbol{q}_{P,j}$ and $\boldsymbol{u}_{P,j}$ for $j > i$, but is dependent for $j \leq i$; $\tilde{\boldsymbol{w}}^M_{P,i}$ is correlated to $\boldsymbol{\theta}_{P,j}$ and $\boldsymbol{h}_{P,j}$ for any j.

These assumptions apply analogously for vector partitions. Also, considering the assumptions in Subsection 4.1.1 for $\boldsymbol{q}_{P,i}$, any upper partition is independent of any lower partition (the only exception is $\tilde{\boldsymbol{w}}_{M,i}$, which is correlated to $\bar{\boldsymbol{\theta}}_{N,j}$ and $\bar{\boldsymbol{h}}_{N,j}$, for any j).

Taking the expectation of the squared-norm of (4.18), the excess MSE can be decomposed as

$$E\boldsymbol{e}_a^2(i) = E\boldsymbol{e}_{a,M}^2(i) + E\bar{\boldsymbol{h}}_{N,i}^{\mathrm{T}}\bar{\boldsymbol{u}}_{N,i}^{\mathrm{T}}\bar{\boldsymbol{u}}_{N,i}\bar{\boldsymbol{h}}_{N,i} + 2E\bar{\boldsymbol{h}}_{N,i}^{\mathrm{T}}\bar{\boldsymbol{u}}_{N,i}^{\mathrm{T}}\boldsymbol{u}_{M,i}(\boldsymbol{h}_{M,i} - \boldsymbol{w}_{M,i-1}). \tag{4.21}$$

Using (4.19), the third term in the right of (4.21) can be decomposed as

$$\begin{aligned}2E\bar{\boldsymbol{h}}_{N,i}^{\mathrm{T}}\bar{\boldsymbol{u}}_{N,i}^{\mathrm{T}}\boldsymbol{u}_{M,i}(\boldsymbol{h}_{M,i} - \boldsymbol{w}_{M,i-1}) &= 2E\bar{\boldsymbol{h}}_{N,i}^{\mathrm{T}}\bar{\boldsymbol{u}}_{N,i}^{\mathrm{T}}\boldsymbol{u}_{M,i}\tilde{\boldsymbol{w}}_{M,i-1}\\ &+ 2E\bar{\boldsymbol{h}}_{N,i}^{\mathrm{T}}\bar{\boldsymbol{u}}_{N,i}^{\mathrm{T}}\boldsymbol{u}_{M,i}\boldsymbol{q}_{M,i} - 2(1-c)E\bar{\boldsymbol{h}}_{N,i}^{\mathrm{T}}\bar{\boldsymbol{u}}_{N,i}^{\mathrm{T}}\boldsymbol{u}_{M,i}\boldsymbol{\theta}_{M,i-1}.\end{aligned} \tag{4.22}$$

As $\boldsymbol{q}_{M,i}$ and $\boldsymbol{\theta}_{M,i-1}$ are zero-mean and independent from $\bar{\boldsymbol{h}}_{N,i}^{\mathrm{T}}\bar{\boldsymbol{u}}_{N,i}^{\mathrm{T}}\boldsymbol{u}_{M,i}$, the second and third terms in the right of (4.22) are zero. The EMSE (4.21) thus becomes

$$\boxed{\mathrm{EMSE}(i) = E\boldsymbol{e}_a^2(i) = E\boldsymbol{e}_{a,M}^2(i) + \zeta_{um}(i)} \tag{4.23}$$

where the undermodeling term $\zeta_{um}(i)$ is

$$\zeta_{um}(i) \triangleq E\bar{\boldsymbol{h}}_{N,i}^{\mathrm{T}}\bar{\boldsymbol{u}}_{N,i}^{\mathrm{T}}\bar{\boldsymbol{u}}_{N,i}\bar{\boldsymbol{h}}_{N,i} + 2E\bar{\boldsymbol{h}}_{N,i}^{\mathrm{T}}\bar{\boldsymbol{u}}_{N,i}^{\mathrm{T}}\boldsymbol{u}_{M,i}\tilde{\boldsymbol{w}}_{M,i-1}. \tag{4.24}$$

In Appendices 4.A and 4.B, we show how to compute the terms in $\zeta_{um}(i)$ from the autocorrelation matrix $R_{h,N,i} \triangleq E\bar{\boldsymbol{h}}_{N,i}\bar{\boldsymbol{h}}_{N,i}^{\mathrm{T}}$ and the cross-correlation matrix $R_{wh,M,N,i} \triangleq E\tilde{\boldsymbol{w}}_{M,i-1}\bar{\boldsymbol{h}}_{N,i}^{\mathrm{T}}$.

Before we proceed to the derivation of $E\boldsymbol{e}_{a,M}^2(i)$, we first address the mean behavior, whose result will be required subsequently.

4.2.1 Mean analysis

Consider the general data-normalized algorithm

$$\boldsymbol{w}_{M,i} = \boldsymbol{w}_{M,i-1} + \mu\frac{\boldsymbol{u}_{M,i}^{\mathrm{T}}}{g\left(\boldsymbol{u}_{M,i}\right)}\boldsymbol{e}(i). \tag{4.25}$$

Subtracting both sides of this expression from $\boldsymbol{h}_{M,i}$, and expanding $\boldsymbol{e}(i)$ as (4.20), we have

$$\boldsymbol{h}_{M,i} - \boldsymbol{w}_{M,i} = \boldsymbol{h}_{M,i} - \boldsymbol{w}_{M,i-1} - \mu\frac{\boldsymbol{u}_{M,i}^{\mathrm{T}}}{g\left(\boldsymbol{u}_{M,i}\right)}\boldsymbol{e}(i), \tag{4.26}$$

$$\boldsymbol{h}_{M,i} - \boldsymbol{w}_{M,i} = \boldsymbol{h}_{M,i} - \boldsymbol{w}_{M,i-1} - \mu\frac{\boldsymbol{u}_{M,i}^{\mathrm{T}}}{g\left(\boldsymbol{u}_{M,i}\right)}(\boldsymbol{v}(i) + \boldsymbol{u}_{P,i}\boldsymbol{h}_{P,i} - \boldsymbol{u}_{M,i}\boldsymbol{w}_{M,i-1}), \tag{4.27}$$

$$\tilde{\boldsymbol{w}}_{M,i} = \left(I_M - \mu\frac{\boldsymbol{u}_{M,i}^{\mathrm{T}}\boldsymbol{u}_{M,i}}{g\left(\boldsymbol{u}_{M,i}\right)}\right)(\boldsymbol{h}_{M,i} - \boldsymbol{w}_{M,i-1}) - \frac{\mu\boldsymbol{u}_{M,i}^{\mathrm{T}}}{g\left(\boldsymbol{u}_{M,i}\right)}\boldsymbol{v}(i) - \mu\frac{\boldsymbol{u}_{M,i}^{\mathrm{T}}\bar{\boldsymbol{u}}_{N,i}}{g\left(\boldsymbol{u}_{M,i}\right)}\bar{\boldsymbol{h}}_{N,i}. \tag{4.28}$$

Expanding $(\boldsymbol{h}_{M,i} - \boldsymbol{w}_{M,i-1})$ as (4.19), we have the recursion for the partial weight error vector as

$$\tilde{\boldsymbol{w}}_{M,i} = \left(I_M - \mu \frac{\boldsymbol{u}_{M,i}^{\mathrm{T}} \boldsymbol{u}_{M,i}}{g\left(\boldsymbol{u}_{M,i}\right)} \right) \tilde{\boldsymbol{w}}_{M,i-1} - \mu \frac{\boldsymbol{u}_{M,i}^{\mathrm{T}}}{g\left(\boldsymbol{u}_{M,i}\right)} \boldsymbol{v}(i) - \mu \frac{\boldsymbol{u}_{M,i}^{\mathrm{T}} \bar{\boldsymbol{u}}_{N,i}}{g\left(\boldsymbol{u}_{M,i}\right)} \bar{\boldsymbol{h}}_{N,i} + \left(I_M - \mu \frac{\boldsymbol{u}_{M,i}^{\mathrm{T}} \boldsymbol{u}_{M,i}}{g\left(\boldsymbol{u}_{M,i}\right)} \right) \big(\boldsymbol{q}_{M,i} - (1-c) \boldsymbol{\theta}_{M,i-1} \big). \quad (4.29)$$

Taking expectation, as $\boldsymbol{v}(i)$, $\boldsymbol{q}_{M,i}$ and $\boldsymbol{\theta}_{M,i-1}$ are zero-mean and uncorrelated to $\boldsymbol{u}_{M,i}$, the terms with them become zero. Also, note that $\bar{\boldsymbol{h}}_{N,i} = \bar{h}_N + \bar{\boldsymbol{\theta}}_{N,i}$, where $\bar{\boldsymbol{\theta}}_{N,i}$ is zero-mean and independent of $\boldsymbol{u}_{M,i}$. With the independence assumptions, $\tilde{\boldsymbol{w}}_{M,i-1}$ is independent of $\boldsymbol{u}_{M,i}$, then (4.29) becomes

$$E\tilde{\boldsymbol{w}}_{M,i} = \left(I_M - \mu T_{1,M} \right) E\tilde{\boldsymbol{w}}_{M,i-1} - \mu T_{4,M,N} \bar{h}_N, \quad (4.30)$$

where ($T_{1,M}$ is the same as previously defined in (3.9))

$$T_{1,M} \triangleq E \frac{\boldsymbol{u}_{M,i}^{\mathrm{T}} \boldsymbol{u}_{M,i}}{g\left(\boldsymbol{u}_{M,i}\right)} \quad \text{and} \quad T_{4,M,N} \triangleq E \frac{\boldsymbol{u}_{M,i}^{\mathrm{T}} \bar{\boldsymbol{u}}_{N,i}}{g\left(\boldsymbol{u}_{M,i}\right)}. \quad (4.31)$$

Then, in steady-state, we have

$$\boxed{\mathrm{E}\tilde{\boldsymbol{w}}_{M,\infty} = -T_{1,M}^{-1} T_{4,M,N} \bar{h}_N.} \quad (4.32)$$

4.2.2 Energy conservation relation

In this subsection, we derive the steady-state behavior of $E\boldsymbol{e}_{a,M}^2(i)$ using the energy conservation relation.

Considering the definition of $\boldsymbol{e}_{a,M}(i)$ in (2.45) and defining the a posteriori estimation error $\boldsymbol{e}_{p,M}(i) \triangleq \boldsymbol{u}_{M,i}(\boldsymbol{h}_{M,i} - \boldsymbol{w}_{M,i})$, we have, by pre-multiplying both sides of (4.26) by $\boldsymbol{u}_{M,i}$,

$$\boldsymbol{e}_{p,M}(i) = \boldsymbol{e}_{a,M}(i) - \mu \frac{||\boldsymbol{u}_{M,i}||^2}{g\left(\boldsymbol{u}_{M,i}\right)} \boldsymbol{e}(i). \quad (4.33)$$

Isolating $\boldsymbol{e}(i)$ in (4.33) and substituting it in (4.26), the scalars μ and $g(\boldsymbol{u}_{M,i})$ are cancelled, and we have

$$(\boldsymbol{h}_{M,i} - \boldsymbol{w}_{M,i}) + \frac{\boldsymbol{u}_{M,i}^{\mathrm{T}} \boldsymbol{e}_{a,M}(i)}{||\boldsymbol{u}_{M,i}||^2} = (\boldsymbol{h}_{M,i} - \boldsymbol{w}_{M,i-1}) + \frac{\boldsymbol{u}_{M,i}^{\mathrm{T}} \boldsymbol{e}_{p,M}(i)}{||\boldsymbol{u}_{M,i}||^2}. \quad (4.34)$$

Taking squared-norm and expectation, the cross-terms are cancelled and we obtain the *energy conservation relation* [35]

$$\underbrace{E||\boldsymbol{h}_{M,i}-\boldsymbol{w}_{M,i}||^2}_{(I)}+\underbrace{E\frac{\boldsymbol{e}^2_{a,M}(i)}{||\boldsymbol{u}_{M,i}||^2}}_{(II)}=\underbrace{E||\boldsymbol{h}_{M,i}-\boldsymbol{w}_{M,i-1}||^2}_{(III)}+\underbrace{E\frac{\boldsymbol{e}^2_{p,M}(i)}{||\boldsymbol{u}_{M,i}||^2}}_{(IV)}. \tag{4.35}$$

Note that (I) is $E||\tilde{\boldsymbol{w}}_{M,i}||^2$. In the following, we compute (III) and (IV) and we will see that the term (II) will be cancelled.

First, let us rewrite (III) in terms of $E||\tilde{\boldsymbol{w}}_{M,i-1}||^2$. Taking the squared norm of (4.19), we have

$$E||\boldsymbol{h}_{M,i}-\boldsymbol{w}_{M,i-1}||^2=E||\tilde{\boldsymbol{w}}_{M,i-1}||^2+\tau_{ns}(i-1), \tag{4.36}$$

where the term τ_{ns}, related to the nonstationarity, is

$$\tau_{ns}(i)\triangleq \mathrm{Tr}Q_M+(1-c)^2\mathrm{Tr}\Theta_{M,i}-2(1-c)\mathrm{Tr}R_{w\theta,M,i}, \tag{4.37}$$

and $R_{w\theta,M,i}\triangleq E\tilde{\boldsymbol{w}}^{\mathrm{T}}_{M,i}\boldsymbol{\theta}_{M,i}$. In appendix 4.B, we show how to compute the cross-correlation term $R_{w\theta,M,i}$ in steady-state.

For (IV), let us rewrite $\boldsymbol{e}_{p,M}(i)$ in terms of $\boldsymbol{e}_{a,M}(i)$ (using (4.20) and (4.33)):

$$\boldsymbol{e}_{p,M}(i)=\left(1-\mu\frac{||\boldsymbol{u}_{M,i}||^2}{g\left(\boldsymbol{u}_{M,i}\right)}\right)\boldsymbol{e}_{a,M}(i)-\mu\frac{||\boldsymbol{u}_{M,i}||^2}{g\left(\boldsymbol{u}_{M,i}\right)}\boldsymbol{v}(i)-\mu\frac{||\boldsymbol{u}_{M,i}||^2\bar{\boldsymbol{u}}_{N,i}}{g\left(\boldsymbol{u}_{M,i}\right)}\bar{\boldsymbol{h}}_{N,i}. \tag{4.38}$$

Taking the square, dividing by $||\boldsymbol{u}_{M,i}||^2$ and taking expectation, the cross-terms with $\boldsymbol{v}(i)$ are zero, and then

$$\begin{aligned}E\frac{\boldsymbol{e}^2_{p,M}(i)}{||\boldsymbol{u}_{M,i}||^2}=E&\left[\left(\frac{1-2\mu\frac{||\boldsymbol{u}_{M,i}||^2}{g\left(\boldsymbol{u}_{M,i}\right)}+\mu^2\frac{||\boldsymbol{u}_{M,i}||^4}{g^2\left(\boldsymbol{u}_{M,i}\right)}}{||\boldsymbol{u}_{M,i}||^2}\right)\boldsymbol{e}^2_{a,M}(i)\right]\\&+\mu^2E\frac{||\boldsymbol{u}_{M,i}||^2}{g^2\left(\boldsymbol{u}_{M,i}\right)}\boldsymbol{v}^2(i)+\mu^2E\bar{\boldsymbol{h}}^{\mathrm{T}}_{N,i}\frac{\bar{\boldsymbol{u}}^{\mathrm{T}}_{N,i}||\boldsymbol{u}_{M,i}||^2\bar{\boldsymbol{u}}_{N,i}}{g^2\left(\boldsymbol{u}_{M,i}\right)}\bar{\boldsymbol{h}}_{N,i}\\&-2\mu E\bar{\boldsymbol{h}}^{\mathrm{T}}_{N,i}\frac{\bar{\boldsymbol{u}}^{\mathrm{T}}_{N,i}}{g\left(\boldsymbol{u}_{M,i}\right)}\left(1-\mu\frac{||\boldsymbol{u}_{M,i}||^2}{g\left(\boldsymbol{u}_{M,i}\right)}\right)\boldsymbol{e}_{a,M}(i).\end{aligned} \tag{4.39}$$

We can expand $\boldsymbol{e}_{a,M}(i)$ in the fourth term on the right side in (4.39) using (2.45). Yet, as the terms $\boldsymbol{q}_{M,i}$ and $\boldsymbol{\theta}_{M,i-1}$ in $(\boldsymbol{h}_{M,i}-\boldsymbol{w}_{M,i-1})$ are zero-mean and uncorrelated to $\bar{\boldsymbol{h}}_{N,i}$, then

$$\begin{aligned}E\frac{\boldsymbol{e}^2_{p,M}(i)}{||\boldsymbol{u}_{M,i}||^2}=E\frac{\boldsymbol{e}^2_{a,M}(i)}{||\boldsymbol{u}_{M,i}||^2}-2\mu E\frac{\boldsymbol{e}^2_{a,M}(i)}{g\left(\boldsymbol{u}_{M,i}\right)}+\mu^2E\frac{||\boldsymbol{u}_{M,i}||^2\boldsymbol{e}^2_{a,M}(i)}{g^2\left(\boldsymbol{u}_{M,i}\right)}\\+\mu^2\sigma^2_v\mathrm{Tr}T_{2,M}+\mu^2E\bar{\boldsymbol{h}}^{\mathrm{T}}_{N,i}T_{6,N}\bar{\boldsymbol{h}}_{N,i}-2\mu E\bar{\boldsymbol{h}}^{\mathrm{T}}_{N,i}\left(T_{4,M,N}-\mu T_{5,M,N}\right)^{\mathrm{T}}\tilde{\boldsymbol{w}}_{M,i-1},\end{aligned} \tag{4.40}$$

where $T_{2,M}$ was defined previously in (3.9) as

$$T_{2,M} \triangleq E\frac{\boldsymbol{u}_{M,i}^{\mathrm{T}}\boldsymbol{u}_{M,i}}{g^2(\boldsymbol{u}_{M,i})} \tag{4.41}$$

and

$$T_{5,M,N} \triangleq E\frac{\boldsymbol{u}_{M,i}^{\mathrm{T}}||\boldsymbol{u}_{M,i}||^2\bar{\boldsymbol{u}}_{N,i}}{g^2\left(\boldsymbol{u}_{M,i}\right)} \quad \text{and} \quad T_{6,N} \triangleq E\frac{\bar{\boldsymbol{u}}_{N,i}^{\mathrm{T}}||\boldsymbol{u}_{M,i}||^2\bar{\boldsymbol{u}}_{N,i}}{g^2\left(\boldsymbol{u}_{M,i}\right)}. \tag{4.42}$$

Defining τ_{um} as the quantity that collects the undermodeling effect in (4.40)

$$\tau_{um}(i) \triangleq \mu E\bar{\boldsymbol{h}}_{N,i}^{\mathrm{T}}T_{6,N}\bar{\boldsymbol{h}}_{N,i} - 2E\bar{\boldsymbol{h}}_{N,i}^{\mathrm{T}}\left(T_{4,M,N} - \mu T_{5,M,N}\right)^{\mathrm{T}}\tilde{\boldsymbol{w}}_{M,i-1}, \tag{4.43}$$

then (4.40) becomes

$$E\frac{\boldsymbol{e}_{p,M}^2(i)}{||\boldsymbol{u}_{M,i}||^2} = E\frac{\boldsymbol{e}_{a,M}^2(i)}{||\boldsymbol{u}_{M,i}||^2} - 2\mu E\frac{\boldsymbol{e}_{a,M}^2(i)}{g\left(\boldsymbol{u}_{M,i}\right)} + \mu^2 E\frac{||\boldsymbol{u}_{M,i}||^2\boldsymbol{e}_{a,M}^2(i)}{g^2\left(\boldsymbol{u}_{M,i}\right)} + \mu^2\sigma_v^2\mathrm{Tr}T_{2,M} + \mu\tau_{um}(i). \tag{4.44}$$

The expectation terms in (4.43) can be computed with the results in Appendices 4.A and 4.B.

Now, substituting (4.36) and (4.40) back into the energy conservation relation (4.35), and considering steady-state $i \to \infty$ so that $E||\tilde{\boldsymbol{w}}_{M,i}||^2 = E||\tilde{\boldsymbol{w}}_{M,i-1}||^2$, we obtain the *steady-state variance relation*

$$\boxed{2\mu E\frac{\boldsymbol{e}_{a,M}^2(\infty)}{g\left(\boldsymbol{u}_{M,\infty}\right)} - \mu^2 E\frac{||\boldsymbol{u}_{M,\infty}||^2\boldsymbol{e}_{a,M}^2(\infty)}{g^2\left(\boldsymbol{u}_{M,\infty}\right)} = \mu^2\sigma_v^2\mathrm{Tr}T_{2,M} + \tau_{ns}(\infty) + \mu\tau_{um}(\infty).} \tag{4.45}$$

In order to isolate the term $E\boldsymbol{e}_{a,M}^2(\infty)$, we must make further independence assumptions and consider specific algorithms to specify the function $g(\cdot)$.

4.2.3 Performance for the LMS algorithm

For the LMS algorithm, $g\left(\boldsymbol{u}_{M,i}\right) = 1$ and the moment matrices become

$$T_{1,M} = T_{2,M} = E\boldsymbol{u}_{M,i}^{\mathrm{T}}\boldsymbol{u}_{M,i} = R_{u,M}, \tag{4.46}$$

$$T_{4,M,N} = E\boldsymbol{u}_{M,i}^{\mathrm{T}}\bar{\boldsymbol{u}}_{N,i} = R_{u,M,N}. \tag{4.47}$$

For the fourth-order moment matrices, we assume that the input is Gaussian and, from the results in Appendix 3.A, we have

$$T_{5,M,N} = E\boldsymbol{u}_{M,i}^{\mathrm{T}}||\boldsymbol{u}_{M,i}||^2\bar{\boldsymbol{u}}_{N,i} = \left(2R_{u,M} + \mathrm{Tr}R_{u,M}I_M\right)R_{u,M,N}, \tag{4.48}$$

$$T_{6,N} = E\bar{\boldsymbol{u}}_{N,i}^{\mathrm{T}}||\boldsymbol{u}_{M,i}||^2\bar{\boldsymbol{u}}_{N,i} = 2R_{u,M,N}^{\mathrm{T}}R_{u,M,N} + R_{u,N}(\mathrm{Tr}R_{u,M}). \tag{4.49}$$

The cross-correlation matrices $R_{w\theta,M,\infty}$ and $R_{wh,M,N,\infty}$, necessary to compute (4.24), (4.37) and (4.43), are, from the results (4.106) and (4.112) in appendix,

$$R_{w\theta,M,\infty} = \frac{1}{1+c}\left[(1-c)I_M + \mu c R_{u,M}\right]^{-1}(I_M - \mu R_{u,M})\,Q_M \tag{4.50}$$

and

$$R_{wh,M,N,\infty} = -R_{u,M}^{-1}R_{u,M,N}\bar{h}_N\bar{h}_N^{\mathrm{T}} - \frac{1}{1-c^2}\left[\left(\frac{1-c}{c\mu}\right)I_M + R_{u,M}\right]^{-1}R_{u,M,N}\bar{Q}_N. \tag{4.51}$$

Recalling from Section 4.1 that $R_{h,N,\infty} = E\bar{\boldsymbol{h}}_{N,\infty}\bar{\boldsymbol{h}}_{N,\infty}^{\mathrm{T}} = \bar{h}_N\bar{h}_N^{\mathrm{T}} + \frac{\bar{Q}_N}{1-c^2}$, using the results in Appendix 4.A to compute bilinear forms and using the cross-correlation matrices above, the nonstationarity and undermodeling terms (4.37), (4.43) and (4.24) become

$$\boxed{\tau_{ns}(\infty) = \frac{2}{1+c}\left[\mathrm{Tr}Q_M - (1-c)\mathrm{Tr}\left[\left[(1-c)I_M + \mu c R_{u,M}\right]^{-1}(I_M - \mu R_{u,M})\,Q_M\right]\right],} \tag{4.52}$$

$$\boxed{\begin{aligned}\tau_{um}(\infty) = \mu\sum\left(2R_{u,M,N}^{\mathrm{T}}R_{u,M,N} + R_{u,N}(\mathrm{Tr}R_{u,M})\right)\odot\left(\bar{h}_N\bar{h}_N^{\mathrm{T}} + \frac{\bar{Q}_N}{1-c^2}\right)\\ -2\sum\left[\left((1-\mu\mathrm{Tr}R_{u,M})I_M - 2\mu R_{u,M}\right)R_{u,M,N}\right]\odot R_{wh,M,N,\infty},\end{aligned}} \tag{4.53}$$

$$\boxed{\zeta_{um}(\infty) = \sum R_{u,N}\odot\left(\bar{h}_N\bar{h}_N^{\mathrm{T}} + \frac{\bar{Q}_N}{1-c^2}\right) + 2\sum R_{u,M,N}\odot R_{wh,M,N,\infty}.} \tag{4.54}$$

In order to isolate $E\boldsymbol{e}_{a,M}^2(\infty)$ in (4.45), we assume that $||\boldsymbol{u}_{M,\infty}||^2$ is independent of $\boldsymbol{e}_{a,M}^2(\infty)$, which is a reasonable approximation in steady-state [35]. Then,

$$\mu E\boldsymbol{e}_{a,M}^2(\infty)\,(2 - \mu\mathrm{Tr}R_{u,M}) = \mu^2\sigma_v^2\mathrm{Tr}R_{u,M} + \tau_{ns}(\infty) + \mu\tau_{um}(\infty) \tag{4.55}$$

$$E\boldsymbol{e}_{a,M}^2(\infty) = \frac{\mu M\sigma_v^2\sigma_u^2 + \mu^{-1}\tau_{ns}(\infty) + \tau_{um}(\infty)}{2 - \mu M\sigma_u^2}. \tag{4.56}$$

Using (4.56) in (4.23), we have

$$\boxed{\mathrm{EMSE}(\infty) = \frac{\mu M\sigma_v^2\sigma_u^2 + \mu^{-1}\tau_{ns}(\infty) + \tau_{um}(\infty)}{2 - \mu M\sigma_u^2} + \zeta_{um}(\infty),} \tag{4.57}$$

where $\tau_{ns}(\infty)$, $\tau_{um}(\infty)$ and $\zeta_{um}(\infty)$ are given, respectively, by (4.52), (4.53) and (4.54).

This result, although general enough to comprise any input signal correlation, filter length and time-variant impulse response under the model (2.6), is quite complex. The expression (4.57) can be simplified under some particular conditions, such as for stationary systems, full-length filters or white input signal.

When the system is stationary, matrix Q_P is identically zero, making $\tau_{ns}(\infty) = 0$ and the terms that depend on Q_P in $\tau_{um}(\infty)$ and $\zeta_{um}(\infty)$ vanish. These terms become

$$\begin{aligned}\tau_{um}(\infty) = \mu \sum \big(2R^{\mathrm{T}}_{u,M,N} R_{u,M,N} + R_{u,N}(\mathrm{Tr} R_{u,M})\big) \odot \bar{h}_N \bar{h}_N^{\mathrm{T}} \\ - 2\sum \Big[\big((1-\mu \mathrm{Tr} R_{u,M}) I_M - 2\mu R_{u,M}\big) R_{u,M,N}\Big] \odot R_{wh,M,N,\infty},\end{aligned} \quad (4.58)$$

$$\zeta_{um}(\infty) = \sum R_{u,N} \odot \bar{h}_N \bar{h}_N^{\mathrm{T}} + 2\sum R_{u,M,N} \odot R_{wh,M,N,\infty}, \quad (4.59)$$

$$R_{wh,M,N,\infty} = -R^{-1}_{u,M} R_{u,M,N} \bar{h}_N \bar{h}_N^{\mathrm{T}}. \quad (4.60)$$

When $M = P$, the terms $R_{u,N}$, $R_{u,M,N}$ and $\bar{\boldsymbol{h}}_{N,i}$ vanish, and consequently $\tau_{um}(\infty) = 0$ and $\zeta_{um}(\infty) = 0$.

When the input signal is white, we can isolate the EMSE in the variance relation (4.45) without the rough independence assumption between $||\boldsymbol{u}_{M,\infty}||^2$ and $\boldsymbol{e}^2_{a,M}(\infty)$. In Appendix 4.C, it is shown that, when input is white,

$$E||\boldsymbol{u}_{M,\infty}||^2 \boldsymbol{e}^2_{a,M}(\infty) = (M+2)\sigma_u^2 E\boldsymbol{e}^2_{a,M}(\infty), \quad (4.61)$$

so that (4.57) becomes

$$\mathrm{EMSE}(\infty) = \frac{\mu M \sigma_v^2 \sigma_u^2 + \mu^{-1}\tau_{ns}(\infty) + \tau_{um}(\infty)}{2 - \mu\sigma_u^2(M+2)} + \zeta_{um}(\infty). \quad (4.62)$$

Furthermore, uncorrelation of samples of $\boldsymbol{u}(i)$ implies that $R_{u,P} = \sigma_u^2 I_P$ and consequently $R_{u,M,N} = 0_{M,N}$ and $R_{wh,M,N,\infty} = 0_{M,N}$, which simplify most of the expressions $\tau_{ns}(\infty)$, $\tau_{um}(\infty)$ and $\zeta_{um}(\infty)$, becoming:

$$\begin{aligned}\tau_{ns}(\infty) &= \frac{2}{1+c}\left[\mathrm{Tr} Q_M - (1-c)\mathrm{Tr}\left[\left((1-c)+\mu c\sigma_u^2\right)^{-1} I_M \left(1-\mu\sigma_u^2\right) I_M Q_M\right]\right] \\ &= \frac{2}{1+c}\left[1 - \frac{(1-c)(1-\mu\sigma_u^2)}{1-c+\mu c\sigma_u^2}\right] \mathrm{Tr} Q_M \\ &= \left(\frac{2}{1+c}\right)\left(\frac{\mu\sigma_u^2}{1-c+\mu c\sigma_u^2}\right) \mathrm{Tr} Q_M \\ &= \frac{2\mu\sigma_u^2 \mathrm{Tr} Q_M}{(1+c)(1-c+\mu c\sigma_u^2)},\end{aligned} \quad (4.63)$$

$$\begin{aligned}\tau_{um}(\infty) &= \mu R_{u,N}(\mathrm{Tr}R_{u,M}) \odot \left(\bar{h}_N\bar{h}_N^{\mathrm{T}} + \frac{\bar{Q}_N}{1-c^2}\right)\\ &= \mu\sigma_u^4 M\left(||\bar{h}_N||^2 + \frac{\mathrm{Tr}\bar{Q}_N}{1-c^2}\right),\end{aligned} \tag{4.64}$$

$$\begin{aligned}\zeta_{um}(\infty) &= R_{u,N} \odot \left(\bar{h}_N\bar{h}_N^{\mathrm{T}} + \frac{\bar{Q}_N}{1-c^2}\right)\\ &= \sigma_u^2\left(||\bar{h}_N||^2 + \frac{\mathrm{Tr}\bar{Q}_N}{1-c^2}\right).\end{aligned} \tag{4.65}$$

4.2.4 Performance for the ϵ-NLMS algorithm

The evaluation of the moment matrices (4.31), (4.41) and (4.42) for the ϵ-NLMS algorithm is, under general input correlation conditions, a very complex task. In the literature, such analyses for the ϵ-NLMS algorithm are traditionally carried out for some specific conditions on the eigenvalues of $R_{u,P}$ and assuming that the input signal is Gaussian [10, 59, 68]. Only recently a more general analysis has been derived [74]. In this section, for simplicity, we focus the case in which all the eigenvalues of $R_{u,P}$ are equal, that is, when the input signal is white.

Recalling that for the ϵ-NLMS algorithm $g(\boldsymbol{u}_{M,i}) = \epsilon + ||\boldsymbol{u}_{M,i}||^2$, and assuming that $\boldsymbol{u}(i)$ is Gaussian and white and that $\epsilon \approx 0$, the moment matrices $T_{1,M}$ and $T_{2,M}$ can be approximated, from Appendix 3.B, as

$$T_{1,M} \approx \frac{1}{M}I_M \quad \text{and} \quad T_{2,M} \approx \frac{1}{M(M+2)\sigma_u^2}I_M. \tag{4.66}$$

Using a similar procedure, and with the results of the fourth-order moments in Appendix 3.A, the other moment matrices become

$$T_{4,M,N} \approx \frac{E\boldsymbol{u}_{M,i}^{\mathrm{T}}\bar{\boldsymbol{u}}_{N,i}}{E||\boldsymbol{u}_{M,i}||^2} = \frac{R_{u,M,N}}{M\sigma_u^2} = 0_{M,N}, \tag{4.67}$$

$$T_{5,M,N} \approx \frac{E\boldsymbol{u}_{M,i}^{\mathrm{T}}||\boldsymbol{u}_{M,i}||^2\bar{\boldsymbol{u}}_{N,i}}{E\left(||\boldsymbol{u}_{M,i}||^2\right)^2} = \frac{\left(2R_{u,M} + \mathrm{Tr}R_{u,M}I_M\right)R_{u,M,N}}{M(M+2)\sigma_u^4} = 0_{M,N}, \tag{4.68}$$

$$T_{6,N} \approx \frac{E\bar{\boldsymbol{u}}_{N,i}^{\mathrm{T}}||\boldsymbol{u}_{M,i}||^2\bar{\boldsymbol{u}}_{N,i}}{E\left(||\boldsymbol{u}_{M,i}||^2\right)^2} = \frac{2R_{u,M,N}^{\mathrm{T}}R_{u,M,N} + R_{u,N}(\mathrm{Tr}R_{u,M})}{M(M+2)\sigma_u^4} = \frac{1}{M+2}I_N. \tag{4.69}$$

The cross-correlation matrices $R_{w\theta,M,\infty}$ and $R_{wh,M,N,\infty}$ become, then,

$$\begin{aligned} R_{w\theta,M,\infty} &= \frac{1}{1+c}\left[(1-c)+\frac{\mu c}{M}\right]^{-1}\left(1-\frac{\mu}{M}\right)Q_M \\ &= \frac{M-\mu}{(1+c)\,(M-Mc+\mu c)}Q_M \end{aligned} \tag{4.70}$$

and

$$R_{wh,M,N,\infty} = 0_{M,N}. \tag{4.71}$$

The nonstationarity and undermodeling terms (4.37), (4.43) and (4.24), then, become

$$\boxed{\tau_{ns}(\infty) = \frac{2\mu \mathrm{Tr} Q_M}{(1+c)\,(M-Mc+\mu c)},} \tag{4.72}$$

$$\boxed{\begin{aligned} \tau_{um}(\infty) = \mu E\bar{\boldsymbol{h}}_{N,\infty}^{\mathrm{T}} T_{6,N}\bar{\boldsymbol{h}}_{N,\infty} &= \frac{\mu}{M+2}E||\bar{\boldsymbol{h}}_{N,\infty}||^2 \\ &= \frac{\mu}{M+2}\left(||\bar{h}_N||^2+\frac{\mathrm{Tr}\bar{Q}_N}{1-c^2}\right), \end{aligned}} \tag{4.73}$$

$$\boxed{\zeta_{um}(\infty) = E\bar{\boldsymbol{h}}_{N,\infty}^{\mathrm{T}} R_{u,N}\bar{\boldsymbol{h}}_{N,\infty} = \sigma_u^2 E||\bar{\boldsymbol{h}}_{N,\infty}||^2 = \sigma_u^2\left(||\bar{h}_N||^2+\frac{\mathrm{Tr}\bar{Q}_N}{1-c^2}\right).} \tag{4.74}$$

As in the LMS case, we assume that $||\boldsymbol{u}_{M,\infty}||^2$ is independent of $\boldsymbol{e}_{a,M}^2(\infty)$. Then, the variance relation (4.45) becomes

$$2\mu E\frac{\boldsymbol{e}_{a,M}^2(\infty)}{||\boldsymbol{u}_{M,\infty}||^2} - \mu^2 E\frac{||\boldsymbol{u}_{M,\infty}||^2\boldsymbol{e}_{a,M}^2(\infty)}{(||\boldsymbol{u}_{M,\infty}||^2)^2} = \mu^2\sigma_v^2\mathrm{Tr}T_{2,M} + \tau_{ns}(\infty) + \mu\tau_{um}(\infty), \tag{4.75}$$

$$E\frac{\boldsymbol{e}_{a,M}^2(\infty)}{||\boldsymbol{u}_{M,\infty}||^2} = \frac{\frac{\mu\sigma_v^2}{(M+2)\sigma_u^2} + \mu^{-1}\tau_{ns}(\infty) + \tau_{um}(\infty)}{2-\mu}, \tag{4.76}$$

$$E\boldsymbol{e}_{a,M}^2(\infty) = \frac{M\sigma_u^2}{2-\mu}\left(\frac{\mu\sigma_v^2}{(M+2)\sigma_u^2} + \mu^{-1}\tau_{ns}(\infty) + \tau_{um}(\infty)\right). \tag{4.77}$$

Considering the approximation for large M, and substituting in (4.23), we have

$$\boxed{\mathrm{EMSE}(\infty) = \frac{\mu\sigma_v^2}{2-\mu} + \frac{M\sigma_u^2}{2-\mu}\left(\mu^{-1}\tau_{ns}(\infty)+\tau_{um}(\infty)\right) + \zeta_{um}(\infty),} \tag{4.78}$$

where $\tau_{ns}(\infty)$, $\tau_{um}(\infty)$ and $\zeta_{um}(\infty)$ are given, respectively, by (4.72), (4.73) and (4.74).

4.3 Optimal step size for white input

In the stationary scenario, treated in Chapter 3, we attain lower and lower steady-state error level as we reduce the step size value. In the nonstationary scenario, in constrast, this does not occur: reducing undefinitely the step size may not improve and might even degrade the filter performance since the filter becomes unable to track the constant changes in the system. A question that arises, consequently, is if there is a step size value that maximizes the tracking performance, that is, that minimizes the steady-state error of the filter. We call such step size value as the *optimal step size* for the tracking case.

In the literature, the optimal step size has been derived for some different models [2, 3, 35]. In the following, we derive the optimal step size for the nonstationary model (2.6), that we employed along this chapter, considering the undermodeling and white input case, and then we analyze its existence condition: is there always a step size that minimizes the steady-state error?

The steady-state EMSE expressions of the LMS and the $\epsilon-$NLMS algorithms for white input, respectively, (4.62) and (4.78), considering the approximation $M \approx M+2$, can be unifiedly written as

$$\text{EMSE}(\infty) = \frac{\chi_0\chi_3\mu}{2-\mu\chi_3} + \frac{M\sigma_u^2\chi_1}{\chi_3(2-\mu\chi_3)(\chi_2+\mu)} + \left(\chi_0 - \sigma_v^2\right) \tag{4.79}$$

where

$$\chi_0 \triangleq \sigma_v^2 + \sigma_u^2\left(||\bar{h}_N||^2 + \frac{\text{Tr}\bar{Q}_N}{1-c^2}\right), \qquad \chi_1 \triangleq \frac{2\text{Tr}Q_M}{c(1+c)}, \tag{4.80}$$

$$\chi_2 \triangleq \begin{cases} \frac{1-c}{c\sigma_u^2} & \text{, for LMS,} \\ \frac{M(1-c)}{c} & \text{, for } \epsilon\text{-NLMS} \end{cases}, \qquad \chi_3 \triangleq \begin{cases} M\sigma_u^2 & \text{, for LMS,} \\ 1 & \text{, for } \epsilon\text{-NLMS} \end{cases}. \tag{4.81}$$

Taking the derivative of (4.79) in terms of μ, we have

$$\begin{aligned}\frac{\partial \text{EMSE}(\infty)}{\partial\mu} &= \frac{2\chi_0\chi_3}{\left(2-\mu\chi_3\right)^2} - \frac{M\sigma_u^2\chi_1\left[2-\chi_2\chi_3-2\chi_3\mu\right]}{\chi_3\left(2-\mu\chi_3\right)^2\left(\chi_2+\mu\right)^2} \\ &= \frac{2\chi_0\chi_3^2\left(\chi_2+\mu\right)^2 + M\sigma_u^2\chi_1\left[\chi_2\chi_3+2\chi_3\mu-2\right]}{\chi_3\left(2-\mu\chi_3\right)^2\left(\chi_2+\mu\right)^2}.\end{aligned} \tag{4.82}$$

The optimal step size μ^o, that minimizes EMSE(∞), is so that (4.82) equals zero. As $0 < \mu < 2/\chi_3$ for stable convergence (for both LMS and ϵ-NLMS) [12], the denominator of (4.82) cannot zero the expression. Then, the optimal step size is such that

$$2\chi_0\chi_3^2\left[(\mu^o)^2 + 2\chi_2\mu^o + \chi_2^2\right] + M\sigma_u^2\chi_1\left[\chi_2\chi_3 + 2\chi_3\mu^o - 2\right] = 0, \tag{4.83}$$

$$2\chi_0\chi_3^2(\mu^o)^2 + 2\chi_3\left(2\chi_0\chi_2\chi_3 + M\sigma_u^2\chi_1\right)\mu^o + 2\chi_0\chi_2^2\chi_3^2 + M\sigma_u^2\chi_1\left[\chi_2\chi_3 - 2\right] = 0. \tag{4.84}$$

Then, as a quadratic function of μ^o, and because step size is positive, the only solution to (4.84) is

$$\boxed{\mu^o = -\mu' + \sqrt{(\mu')^2 - \mu''},} \tag{4.85}$$

where

$$\mu' = \frac{2\chi_3\left(2\chi_0\chi_2\chi_3 + M\sigma_u^2\chi_1\right)}{4\chi_0\chi_3^2} = \chi_2 + \frac{M\sigma_u^2\chi_1}{2\chi_0\chi_3} \tag{4.86}$$

and

$$\begin{aligned}\mu'' &= \frac{2\chi_0\chi_2^2\chi_3^2 + M\sigma_u^2\chi_1\left[\chi_2\chi_3 - 2\right]}{2\chi_0\chi_3^2} \\ &= \chi_2^2 + \frac{\chi_2 M\sigma_u^2\chi_1}{2\chi_0\chi_3} - \frac{M\sigma_u^2\chi_1}{\chi_0\chi_3^2} = \chi_2\mu' - \frac{M\sigma_u^2\chi_1}{\chi_0\chi_3^2}.\end{aligned} \tag{4.87}$$

The only terms that determine whether the LMS or the ϵ-NLMS algorithm is cosidered are χ_2 and χ_3, computed as in (4.81).

4.3.1 Existence condition for the optimal step size

As we can notice from (4.85), the optimal step size does not always exist. Analytically, if $\mu'' \geq 0$, then μ^o would be zero or negative, which are not valid values for a step size, and thus an optimal step size would not exist. This means that, when $\mu'' \geq 0$, as we decrease undefinitely the step size towards zero, the steady-state error continually decreases, but converging to a nonzero value. In practice, as we will see in the simulations of Section 4.4, decreasing the step size from a certain point on does not yield relevant improvement in the steady-state performance for these cases.

Let us derive with more detail the conditions under which the optimal step size exists. For the LMS algorithm, substituting χ_1, χ_2 and χ_3 from (4.80) and (4.81) into (4.87), we have

$$\mu'' = \left(\frac{1-c}{c\sigma_u^2}\right)^2 + \left(\frac{1-c}{c\sigma_u^2}\right)\frac{\mathrm{Tr}Q_M}{c(1+c)\chi_0} - \frac{2\mathrm{Tr}Q_M}{M\sigma_u^2 c(1+c)\chi_0}. \tag{4.88}$$

The optimal step size exists if $\mu'' < 0$, that is, if

$$\left(\frac{1-c}{c\sigma_u^2}\right)^2 + \left[\frac{1-c}{c\sigma_u^2} - \frac{2}{M\sigma_u^2}\right]\frac{\mathrm{Tr}Q_M}{c(1+c)\chi_0} < 0, \tag{4.89}$$

$$\left(\frac{2}{M} - \frac{1-c}{c}\right)\frac{\mathrm{Tr}Q_M}{c(1+c)\chi_0} > \frac{(1-c)^2}{c^2\sigma_u^2}, \tag{4.90}$$

$$\left(\frac{2}{M} - \frac{1-c}{c}\right)\left(\frac{c}{1-c}\right)\frac{\mathrm{Tr}Q_M}{1-c^2} > \frac{\chi_0}{\sigma_u^2}. \tag{4.91}$$

Then, substituting χ_0 from (4.80) into (4.91), we have

$$\boxed{\left(\frac{2c}{M(1-c)}-1\right)\frac{\mathrm{Tr}Q_M}{1-c^2} > \frac{\sigma_v^2}{\sigma_u^2}+||\bar{h}_N||^2+\frac{\mathrm{Tr}\bar{Q}_N}{1-c^2}.} \tag{4.92}$$

Analogously, for the ϵ-NLMS algorithm,

$$\mu'' = \frac{M^2(1-c)^2}{c^2}+M\sigma_u^2\left(\frac{M(1-c)}{c}\right)\frac{\mathrm{Tr}Q_M}{c(1+c)\chi_0}-2M\sigma_u^2\frac{\mathrm{Tr}Q_M}{c(1+c)\chi_0}. \tag{4.93}$$

Imposing $\mu'' < 0$, it follows that

$$\frac{M^2(1-c)^2}{c^2}+M\sigma_u^2\left(\frac{M(1-c)}{c}-2\right)\frac{\mathrm{Tr}Q_M}{c(1+c)\chi_0} < 0, \tag{4.94}$$

$$\sigma_u^2\left(2-\frac{M(1-c)}{c}\right)\frac{\mathrm{Tr}Q_M}{c(1+c)\chi_0} > \frac{M(1-c)^2}{c^2}, \tag{4.95}$$

$$\left(2-\frac{M(1-c)}{c}\right)\left(\frac{c}{M(1-c)}\right)\frac{\mathrm{Tr}Q_M}{1-c^2} > \frac{\chi_0}{\sigma_u^2}, \tag{4.96}$$

which yields the same existence condition in (4.92).

From the right-hand side of (4.92), note that the optimal step size μ^o exists if undermodeling (given by the terms $||\bar{h}_N||^2$ and $\mathrm{Tr}\bar{Q}_N$) is sufficiently low, and if the SNR (proportional to the inverse of σ_v^2/σ_u^2) is sufficiently high. Also, the left-hand side of (4.92) must be positive for the inequality to hold, and so a necessary (but not sufficient) condition is

$$\frac{2c}{M(1-c)}-1 > 0, \tag{4.97}$$

which, after trivial manipulation and considering the bound $|c| < 1$, yields

$$\frac{M}{M+2} < c < 1. \tag{4.98}$$

Recall that all these derivations come from (4.79), which assumes the approximation for large length $M \approx M+2$. Using this approximation also in (4.98) gives a contradiction, since the equality $c = 1$ is not allowed. Instead, we can interpret the lower bound $M/(M+2)$ as a value very close to one for large M, and we will indeed verify, in the simulations in Section 4.4, that the optimal step size exists only for c very close to one.

The conditional existence of the optimal step size in tracking has also been treated in [75], but considering the case of general input correlation and no undermodeling. In the derivations above, by considering the case of white input, we could derive a closed-form expression for the existence condition (4.92), also including the undermodeling effect.

When the filter is not undermodeled (the two last terms in the right on (4.92) do not exist, by construction) and making $c \to 1$, note that condition (4.92) always holds because the term on the left grows unboundedly to positive infinity. This is consistent with the results in the literature [2,35], in which the optimal step size always exist for the random-walk model (and not undermodeled filter).

4.4 Simulations

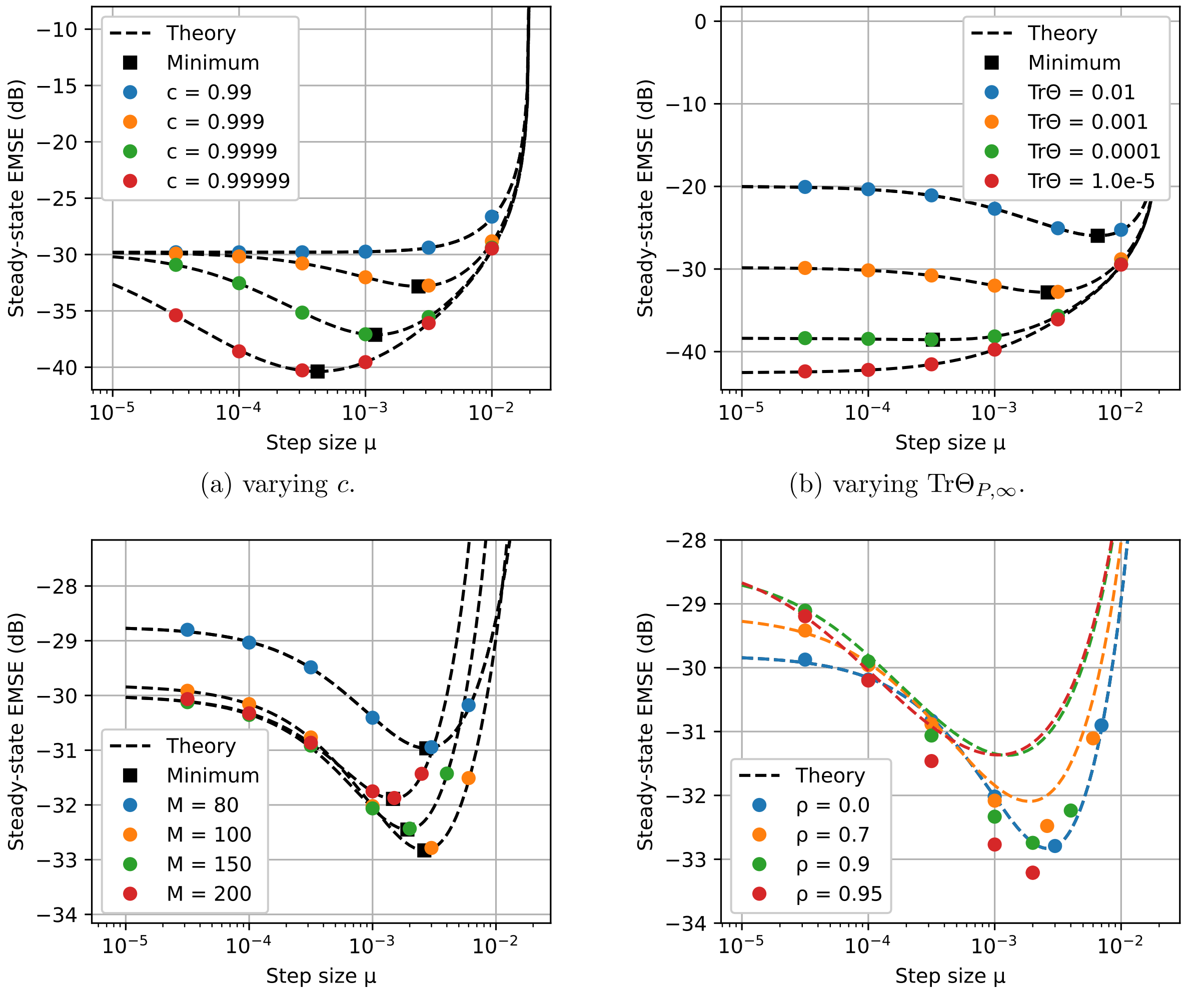
We present simulations to visualize the behavior of the tracking performance derived in this chapter, under distinct environment conditions. For all the following simulations, we consider $\sigma_u^2 = 1$, $\sigma_v^2 = 0.001$, and impulse response with decay $\alpha = 0.05$ (for both h_P and $q_{P,i}$), $||h_P||^2 = 1$, $P = 500$, that can be regarded as infinite for the decay α employed, and the mean h_P is a pure exponential. The other environment parameters and the step size value changes along the tests.

Figure 23 shows curves of EMSE(∞) × μ for the LMS algorithm, alternating which parameters are fixed and which ones are varied. We compare theoretical curves (dash-lines), computed as (4.57), to steady-state values obtained experimentally (colored dots). The experimental measures are obtained from approximate learning curves of EMSE that are the ensemble average of 300 realizations, from which we compute the average of 10000 samples in steady-state.

Figure 23a shows the case in which we fix $\text{Tr}\Theta_{P,\infty} = 0.001$, $M = 100$, white input $\rho = 0$ and vary c (note that, by fixing $\text{Tr}\Theta_{P,\infty}$ and varying c, we are varying $\text{Tr}Q_P$ according to relation (4.4)). As c gets closer to 1 (the time-varying component in the impulse response is more correlated and slow), the steady-state error can attain lower values. This happens because slower variations in the impulse response are easier to track. However, as we take smaller and smaller step sizes, the steady-state error increases because the filter becomes unable to track the system variations. Note that all the curves converge to the same value of EMSE(∞) as $\mu \to 0$. Indeed, we can verify, from (4.62), that

$$\begin{aligned}\lim_{\mu\to 0} \text{EMSE}(\infty) &= \frac{1}{2}\frac{2\sigma_u^2 \text{Tr}Q_M}{(1+c)(1-c)} + \sigma_u^2\left(||\bar{h}_N||^2 + \frac{\text{Tr}\bar{Q}_N}{1-c^2}\right)\\ &= \sigma_u^2\left(\frac{\text{Tr}Q_M}{1-c^2} + \frac{\text{Tr}\bar{Q}_N}{1-c^2} + ||\bar{h}_N||^2\right) = \sigma_u^2\left(\text{Tr}\Theta_{P,\infty} + ||\bar{h}_N||^2\right). \qquad (4.99)\end{aligned}$$

which is constant for fixed $\text{Tr}\Theta_{P,\infty}$ and M. This means that when the filter is much slower than the impulse response variations, the filter gets stuck on the impulse response mean h_P, and EMSE(∞) depends mostly on $\text{Tr}\Theta_{P,\infty}$, which is related to the variance of the



(a) varying c.

(b) varying $\mathrm{Tr}\Theta_{P,\infty}$.

(c) varying M.

(d) varying ρ.

Figure 23: Comparing EMSE(∞) × μ curves for the LMS algorithm, under different scenario conditions; dash-lines: theoretical curves; dots: experimental data. Parameters (if a specific parameter is not varied): $\mathrm{Tr}\Theta_{P,\infty} = 0.001$, $M = 100$, $\rho = 0$ (white input), $c = 0.999$, $\sigma_u^2 = 1$, $\sigma_v^2 = 0.001$, $\alpha = 0.05$.

entries of $\boldsymbol{\theta}_{P,\infty}$.

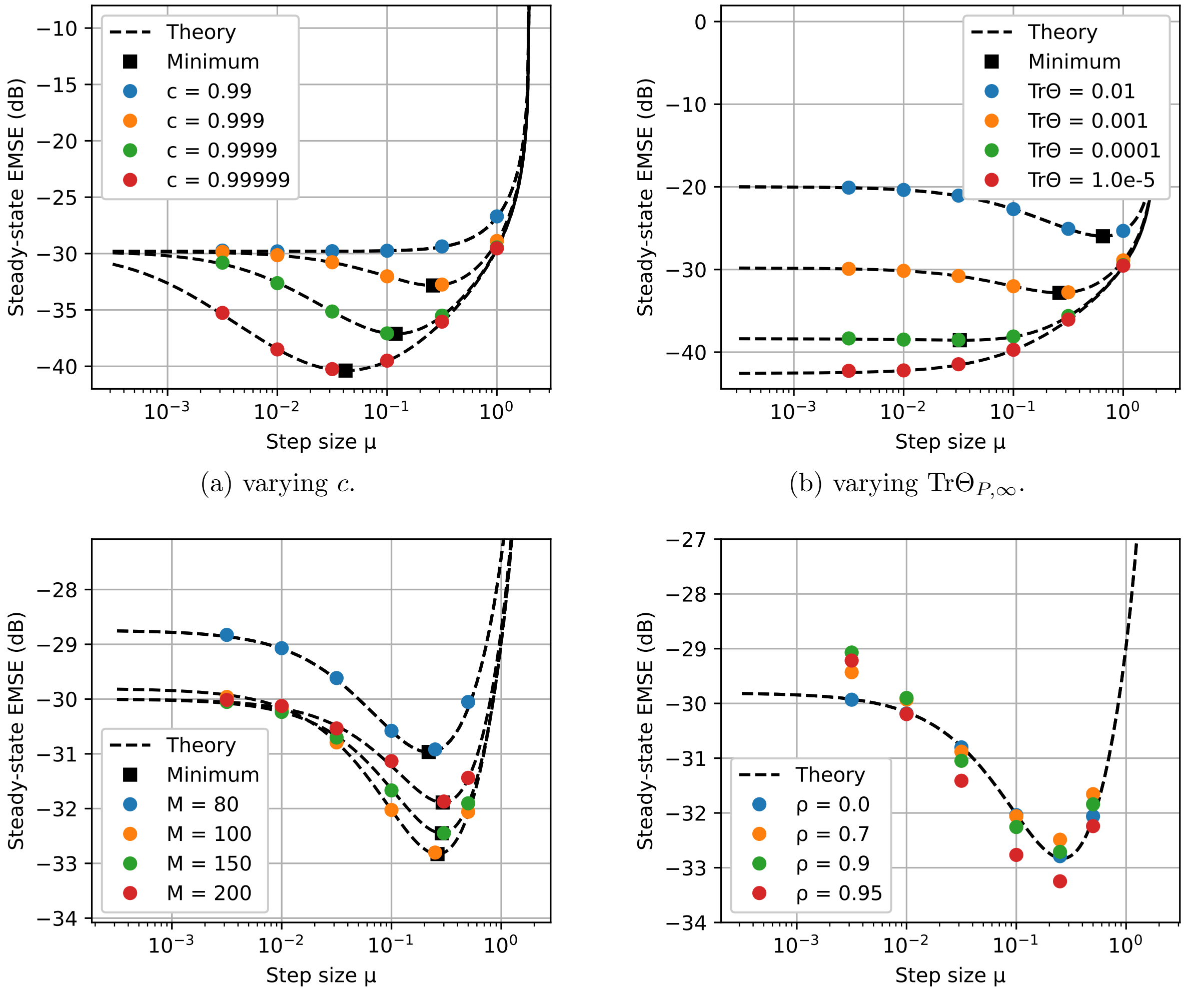
The squares point out the theoretical minimum steady-state EMSE value obtained with the optimal step size (4.85). Note that they are indeed the minimum of each curve. Note that the curve for $c = 0.99$ is a crescent function and does not have a minimum value (although satisfying the necessary condition (4.98)). The value of the optimal step size computed via (4.85), in this case, is negative. And although this curve does not have a minimum value, note that the improvement of the tracking performance by reducing the step size to values smaller than 0.001 is negligible.

Figure 23b shows a similar plot, now fixing $c = 0.999$, $M = 100$, $\rho = 0$ and showing curves for distinct $\mathrm{Tr}\Theta_{P,\infty}$. As expected, larger values of $\mathrm{Tr}\Theta_{P,\infty}$ yield larger error in steady-state. For very small values of $\mathrm{Tr}\Theta_{P,\infty}$, there is no optimal step size and the tracking performance continually improves as we decrease the step size. Figure 23c shows these curves when fixing $c = 0.999$, $\mathrm{Tr}\Theta_{P,\infty} = 0.001$, $\rho = 0$ and varying M. As we decrease M from 200 to 100, the minimum EMSE(∞) continually decreases, but EMSE(∞) for $M = 80$ suffers too much undermodeling. Figure 23d presents these results for different input correlation conditions, for $c = 0.999$, $M = 100$ and $\mathrm{Tr}\Theta_{P,\infty} = 0.001$. Note that, for μ smaller than 10^{-4}, the theoretical values are reasonably accurate. For step sizes larger than that, however, the theoretical model does not adhere very well to experimental data for correlated input. This can be explained by the fact that the independence assumptions are less accurate for large step sizes, in special for correlated input. In particular, for undermodeled filters, we have additional terms $\bar{\boldsymbol{u}}_{N,i}$, that are still very correlated to $\boldsymbol{u}_{M,j}$, for $i \neq j$, when ρ is high.

Figure 24 presents plots similar to those in Figure 23, now for the ϵ-NLMS algorithm. The results are very similar to those obtained for the LMS algorithm. A remarkable difference is in the step size values, that are two orders of magnitude higher than those for the LMS. Another difference is the absence of theoretical curves when $\rho \neq 0$, which we do not derive in this text.

Now, let us observe the behavior of EMSE(∞) as function of the filter length M. Figure 25a shows curves EMSE(∞) × M using the LMS algorithm, for different values of step size. We consider the same environment scenario as in the simulations above in this section, and we fix $\mathrm{Tr}\Theta_{P,\infty} = 0.01$, $c = 0.999$, and we consider white input $\rho = 0$. For small step size ($\mu = 0.001$), the performance is roughly constant for $M = 100$ or higher, which implies no advantage in using a filter larger than that. For $\mu = 0.003$, EMSE(∞) reduces and the advantage of using a filter around $M = 80$ and $M = 100$ rather than larger lengths becomes more evident. As we use larger step sizes, filters with larger lengths becomes severely degraded. We also show the minimum EMSE(∞) for every filter length, attained using the optimal step size (4.85), by the solid gray curve. Figure 25b shows the performance curves of some of the filters that are close to this minimum. These performance curves are obtained from the ensemble average of 1000 realizations. Note that, as we decrease the filter length down to $M = 90$, we achieve both lower steady-state error and higher convergence rate, and naturally reduced complexity, evidencing that excessive taps in nonstationary environment are very disadvantageous.

Figure 26 presents plots analogous to those in Figure 25, now for impulse response decay with $\alpha = 0.1$. The curves EMSE(∞)×M are similar to the previous simulation, but, as expected for steeper impulse response, the lowest steady-state error value is attained



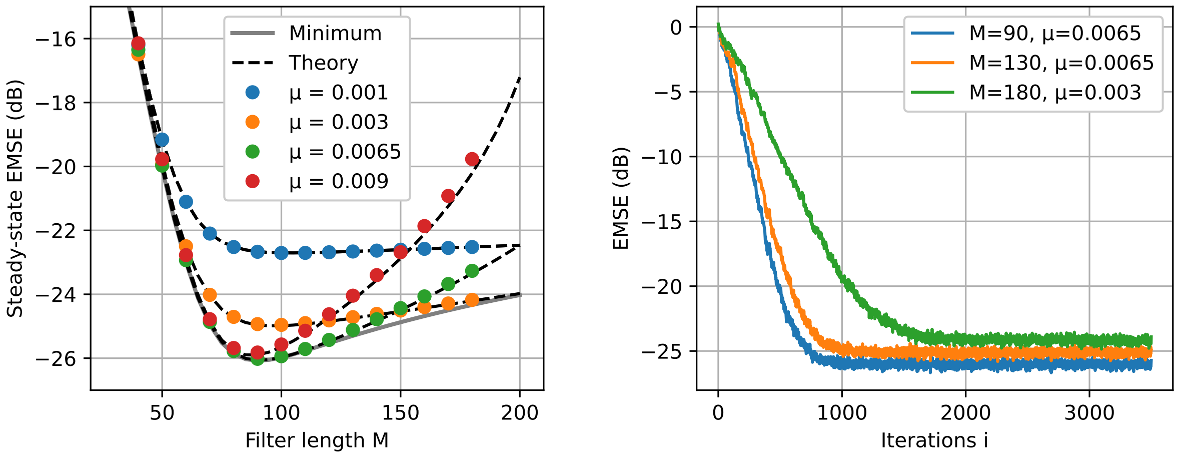
(a) varying c.

(b) varying $\mathrm{Tr}\Theta_{P,\infty}$.

(c) varying M.

(d) varying ρ.

Figure 24: Comparing EMSE(∞) × μ curves for the ϵ-NLMS algorithm, under different scenario conditions; dash-lines: theoretical curves; dots: experimental data. Parameters (if a specific parameter is not varied): $\mathrm{Tr}\Theta_{P,\infty} = 0.001$, $M = 100$, $\rho = 0$ (white input), $c = 0.999$, $\sigma_u^2 = 1$, $\sigma_v^2 = 0.001$, $\alpha = 0.05$.



(a)

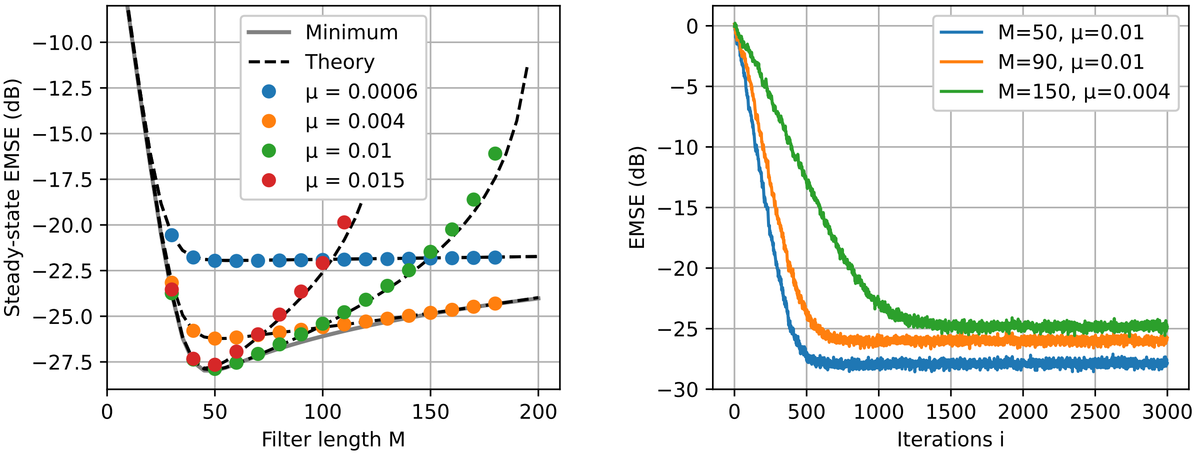
(b)

Figure 25: (a) Curves EMSE(∞) × M for the LMS algorithm; dash-lines: theoretical values; dots: experimental data; (b) Comparing performance curves of filters of different lengths. Parameters: $\mathrm{Tr}\Theta_{P,\infty} = 0.01$, $\rho = 0$ (white input), $c = 0.999$, $\sigma_u^2 = 1$, $\sigma_v^2 = 0.001$, $\alpha = 0.05$.

for smaller length, around $M = 40$ and $M = 60$. Also, the performance curves in Figure 26b, of filters whose parameters M and μ are taken close to the gray line in Figure 26a (minimum steady-state EMSE), show more evident advantages of filters with smaller length.

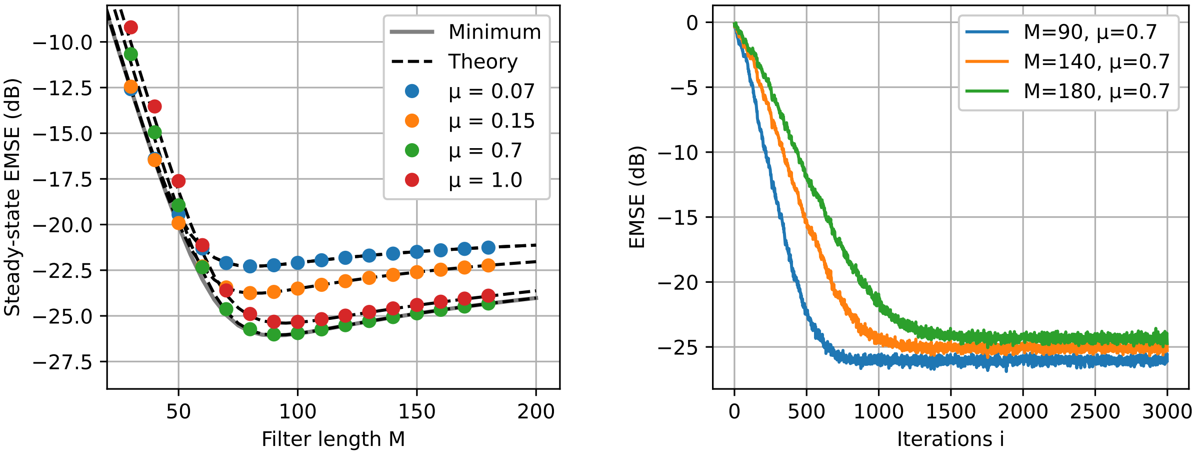
Figure 27 presents simulations for the same scenario in Figure 25, but using the ϵ-NLMS algorithm. Now, curves of different step size have similar shape, differing mostly by a vertical displacement, and does not increase more abruptly for large values of μ. In this sense, note that the optimal step size is roughly the same for all filter lengths, of about $\mu = 0.7$. The performance curves shown in Figure 27b, for filters with the optimal step size and with different lengths, are still very similar to those for the LMS case, where filters with smaller lengths, within a certain range of values, exhibit both better convergence rate and better steady-state performance.

Because shorter filters exhibit both advantages in the transient and in the steady-state performance when compared to longer ones in nonstationary scenario, a combination of filters as proposed in the stationary case would not provide considerable performance improvement, as the supervisor would choose the undermodeled filter all the time. In fact, the combination would only increase the computational cost in this case.



(a) (b)

Figure 26: (a) Curves EMSE(∞) × M for the LMS algorithm, for $\alpha = 0.1$; dash-lines: theoretical values; dots: experimental data; (b) Comparing performance curves of filters of different lengths. Parameters: $\text{Tr}\Theta_{P,\infty} = 0.01$, $\rho = 0$ (white input), $c = 0.999$, $\sigma_u^2 = 1$, $\sigma_v^2 = 0.001$, $\alpha = 0.1$.



(a) (b)

Figure 27: (a) Curves EMSE(∞) × M for the ϵ-NLMS algorithm; dash-lines: theoretical values; dots: experimental data; (b) Comparing performance curves of filters of different lengths.

4.A Computation of bilinear forms

In this chapter, we often deal with bilinear forms[1]

$$b = E\boldsymbol{x}^{\mathrm{T}} A \boldsymbol{y}, \tag{4.100}$$

where $\boldsymbol{x}$ and $\boldsymbol{y}$ are correlated random vectors, and thus we cannot compute the expected values separately. This bilinear form can be written as the summation

$$b = E\sum_i \boldsymbol{x}(i) \sum_j A(i,j)\boldsymbol{y}(j) = \sum_i \sum_j A(i,j) E\boldsymbol{x}(i)\boldsymbol{y}(j). \tag{4.101}$$

which is simply the sum of all elements of the entrywise product of A and $E\boldsymbol{x}\boldsymbol{y}^{\mathrm{T}}$, that is,

$$b = \sum A \odot E\boldsymbol{x}\boldsymbol{y}^{\mathrm{T}}, \tag{4.102}$$

As so, the moment $E\boldsymbol{x}\boldsymbol{y}^{\mathrm{T}}$ can be computed apart from the matrix A. Note that, when $A = I$, $b = \boldsymbol{x}^{\mathrm{T}}\boldsymbol{y}$ is the trace of matrix $E\boldsymbol{x}\boldsymbol{y}^{\mathrm{T}}$.

4.B Computation of cross-correlations of the weight error vector

In this appendix, we derive the correlations matrices $R_{w\theta,M,i} = E\tilde{\boldsymbol{w}}_{M,i}\boldsymbol{\theta}_{M,i}^{\mathrm{T}}$ and $R_{wh,M,N,i} = E\tilde{\boldsymbol{w}}_{M,i-1}\bar{\boldsymbol{h}}_{N,i}^{\mathrm{T}}$ in steady-state. Matrix $R_{wh,M,N,i}$ does not appear explicitly in the derivations, and rather appear through the bilinear form $E\tilde{\boldsymbol{w}}_{M,i-1}^{\mathrm{T}} A_{M,N}\bar{\boldsymbol{h}}_{N,i}$, for some matrix $A_{M,N}$, in equations (4.24) and (4.43). Using the results from Appendix 4.A, we need the correlation matrix $R_{wh,M,N,i}$ to compute this bilinear form.

Expanding $\tilde{\boldsymbol{w}}_{M,i}$ by the weight error recursion (4.29), and $\boldsymbol{\theta}_{M,i}$ as in (4.1), the cross-

[1] The variable names in this appendix are illustrative and do not refer to specific variables defined in the main text.

correlation matrix $R_{w\theta,M,i}$ can be written as

$$R_{w\theta,M,i} = E\left(I_M - \mu\frac{\boldsymbol{u}_{M,i}^{\mathrm{T}}\boldsymbol{u}_{M,i}}{g\left(\boldsymbol{u}_{M,i}\right)}\right)\tilde{\boldsymbol{w}}_{M,i-1}\left(c\boldsymbol{\theta}_{M,i-1}^{\mathrm{T}} + \boldsymbol{q}_{M,i}^{\mathrm{T}}\right)$$
$$- \mu E\left(\boldsymbol{v}(i) + \bar{\boldsymbol{u}}_{N,i}\bar{\boldsymbol{h}}_{N,i}\right)\frac{\boldsymbol{u}_{M,i}^{\mathrm{T}}}{g\left(\boldsymbol{u}_{M,i}\right)}\left(c\boldsymbol{\theta}_{M,i-1}^{\mathrm{T}} + \boldsymbol{q}_{M,i}^{\mathrm{T}}\right)$$
$$+ E\left(I_M - \mu\frac{\boldsymbol{u}_{M,i}^{\mathrm{T}}\boldsymbol{u}_{M,i}}{g\left(\boldsymbol{u}_{M,i}\right)}\right)\boldsymbol{q}_{M,i}\left(c\boldsymbol{\theta}_{M,i-1}^{\mathrm{T}} + \boldsymbol{q}_{M,i}^{\mathrm{T}}\right)$$
$$- (1-c)E\left(I_M - \mu\frac{\boldsymbol{u}_{M,i}^{\mathrm{T}}\boldsymbol{u}_{M,i}}{g\left(\boldsymbol{u}_{M,i}\right)}\right)\boldsymbol{\theta}_{M,i-1}\left(c\boldsymbol{\theta}_{M,i-1}^{\mathrm{T}} + \boldsymbol{q}_{M,i}^{\mathrm{T}}\right). \quad (4.103)$$

Note that $\boldsymbol{v}(i)$ is independent of all others quantities, $\boldsymbol{q}_{M,i}$ is uncorrelated to all quantities but itself, and $\boldsymbol{\theta}_{M,i-1}$ is uncorrelated to $\bar{\boldsymbol{h}}_{N,i}$. Thus

$$R_{w\theta,M,i} = c\left(I_M - \mu T_{1,M}\right)R_{w\theta,M,i-1} + \left(I_M - \mu T_{1,M}\right)\Big(Q_M - c(1-c)\Theta_{M,i-1}\Big). \quad (4.104)$$

In steady-state, recall that $\Theta_{M,\infty} = (1-c^2)^{-1}Q_M$, then

$$\left[(1-c)I_M + \mu c T_{1,M}\right]R_{w\theta,M,\infty} = \left(I_M - \mu T_{1,M}\right)\left[Q_M - c(1-c)(1-c^2)^{-1}Q_M\right], \quad (4.105)$$

$$\boxed{R_{w\theta,M,\infty} = \frac{1}{1+c}\left[(1-c)I_M + \mu c T_{1,M}\right]^{-1}\left(I_M - \mu T_{1,M}\right)Q_M.} \quad (4.106)$$

For $R_{wh,M,N,i}$, note that

$$R_{wh,M,N,i} = E\tilde{\boldsymbol{w}}_{M,i-1}\bar{\boldsymbol{h}}_{N,i}^{\mathrm{T}} = E\tilde{\boldsymbol{w}}_{M,i-1}\bar{h}_N^{\mathrm{T}} + E\tilde{\boldsymbol{w}}_{M,i-1}\bar{\boldsymbol{\theta}}_{N,i}^{\mathrm{T}}. \quad (4.107)$$

Let us denote $R_{w\theta,M,N,i} \triangleq E\tilde{\boldsymbol{w}}_{M,i-1}\bar{\boldsymbol{\theta}}_{N,i}^{\mathrm{T}}$. Analogously to (4.103), $R_{w\theta,M,N,i}$ can be computed recursively as

$$R_{w\theta,M,N,i} = E\left(I_M - \mu\frac{\boldsymbol{u}_{M,i-1}^{\mathrm{T}}\boldsymbol{u}_{M,i-1}}{g\left(\boldsymbol{u}_{M,i-1}\right)}\right)\tilde{\boldsymbol{w}}_{M,i-2}\left(c\bar{\boldsymbol{\theta}}_{N,i-1}^{\mathrm{T}} + \bar{\boldsymbol{q}}_{N,i}^{\mathrm{T}}\right)$$
$$- \mu E\left(\boldsymbol{v}(i-1) + \bar{\boldsymbol{u}}_{N,i-1}\bar{\boldsymbol{h}}_{N,i-1}\right)\frac{\boldsymbol{u}_{M,i-1}^{\mathrm{T}}}{g\left(\boldsymbol{u}_{M,i-1}\right)}\left(c\bar{\boldsymbol{\theta}}_{N,i-1}^{\mathrm{T}} + \bar{\boldsymbol{q}}_{N,i}^{\mathrm{T}}\right)$$
$$+ E\left(I_M - \mu\frac{\boldsymbol{u}_{M,i-1}^{\mathrm{T}}\boldsymbol{u}_{M,i-1}}{g\left(\boldsymbol{u}_{M,i-1}\right)}\right)\boldsymbol{q}_{M,i-1}\left(c\bar{\boldsymbol{\theta}}_{N,i-1}^{\mathrm{T}} + \bar{\boldsymbol{q}}_{N,i}^{\mathrm{T}}\right)$$
$$- (1-c)E\left(I_M - \mu\frac{\boldsymbol{u}_{M,i-1}^{\mathrm{T}}\boldsymbol{u}_{M,i-1}}{g\left(\boldsymbol{u}_{M,i-1}\right)}\right)\boldsymbol{\theta}_{M,i-2}\left(c\bar{\boldsymbol{\theta}}_{N,i-1}^{\mathrm{T}} + \bar{\boldsymbol{q}}_{N,i}^{\mathrm{T}}\right). \quad (4.108)$$

As $\bar{\boldsymbol{\theta}}_{N,i-1}$ and $\bar{\boldsymbol{q}}_{N,i}$ are zero mean and uncorrelated to $\boldsymbol{\theta}_{M,i-2}$, $\boldsymbol{q}_{M,i-1}$ and $\boldsymbol{v}(i-1)$, then

$$\begin{aligned} R_{w\theta,M,N,i} = c \left(I_M - \mu E \frac{\boldsymbol{u}^{\mathrm{T}}_{M,i-1}\boldsymbol{u}_{M,i-1}}{g\left(\boldsymbol{u}_{M,i-1}\right)} \right) R_{w\theta,M,N,i-1} \\ - \mu E \frac{\boldsymbol{u}^{\mathrm{T}}_{M,i-1}\bar{\boldsymbol{u}}_{N,i-1}}{g\left(\boldsymbol{u}_{M,i-1}\right)} E(\bar{h}_{N,0} + \bar{\boldsymbol{\theta}}_{N,i-1}) \left(c\bar{\boldsymbol{\theta}}^{\mathrm{T}}_{N,i-1} + \bar{\boldsymbol{q}}^{\mathrm{T}}_{N,i} \right), \end{aligned} \tag{4.109}$$

$$R_{w\theta,M,N,i} = c\left(I_M - \mu T_{1,M}\right) R_{w\theta,M,N,i-1} - c\mu T_{4,M,N}\bar{\Theta}_{N,i-1}, \tag{4.110}$$

and, in steady-state,

$$R_{w\theta,M,N,\infty} = -\left[\left(\frac{1-c}{c\mu} \right) I_M + T_{1,M} \right]^{-1} T_{4,M,N}\bar{\Theta}_{N,i-1} \tag{4.111}$$

Back to (4.107), and using the mean behavior (4.32), we have

$$\boxed{R_{wh,M,N,\infty} = -T^{-1}_{1,M} T_{4,M,N}\bar{h}_N\bar{h}^{\mathrm{T}}_N - \left[\left(\frac{1-c}{c\mu} \right) I_M + T_{1,M} \right]^{-1} T_{4,M,N}\frac{\bar{Q}_N}{1-c^2}} \tag{4.112}$$

4.C Simplification of variance relation for white input

When the input signal $\boldsymbol{u}(i)$ is white, we can isolate the EMSE $E\boldsymbol{e}^2_{a,M}(i)$ in the term $E||\boldsymbol{u}_{M,\infty}||^2\boldsymbol{e}^2_{a,M}(i)$, from the variance relation (4.45), without making a strong assumption as used in Section 4.2.2.

Note that, when input is white, $R_{u,M} = \sigma^2_u I_M$ and the partial EMSE can be rewritten as

$$\begin{aligned} E\boldsymbol{e}^2_{a,M}(i) &= E(\boldsymbol{h}_{M,i} - \boldsymbol{w}_{M,i-1})^{\mathrm{T}}\boldsymbol{u}_{M,i}\boldsymbol{u}^{\mathrm{T}}_{M,i}(\boldsymbol{h}_{M,i} - \boldsymbol{w}_{M,i-1}) \\ &= E(\boldsymbol{h}_{M,i} - \boldsymbol{w}_{M,i-1})^{\mathrm{T}}R_{u,M}(\boldsymbol{h}_{M,i} - \boldsymbol{w}_{M,i-1}) \\ &= \sigma^2_u E||\boldsymbol{h}_{M,i} - \boldsymbol{w}_{M,i-1}||^2. \end{aligned} \tag{4.113}$$

The term $E||\boldsymbol{u}_{M,i}||^2\boldsymbol{e}^2_{a,M}(i)$ can be written as

$$E||\boldsymbol{u}_{M,i}||^2\boldsymbol{e}^2_{a,M}(i) = E(\boldsymbol{h}_{M,i} - \boldsymbol{w}_{M,i-1})^{\mathrm{T}}\boldsymbol{u}_{M,i}\boldsymbol{u}^{\mathrm{T}}_{M,i}\boldsymbol{u}_{M,i}\boldsymbol{u}^{\mathrm{T}}_{M,i}(\boldsymbol{h}_{M,i} - \boldsymbol{w}_{M,i-1}), \tag{4.114}$$

where the fourth-order moment can be computed as in Appendix 3.A

$$\begin{aligned} E||\boldsymbol{u}_{M,i}||^2\boldsymbol{e}^2_{a,M}(i) &= E(\boldsymbol{h}_{M,i}-\boldsymbol{w}_{M,i-1})^{\mathrm{T}}\left(2R^2_{u,M}+(\mathrm{Tr}R_{u,M})R_{u,M}\right)(\boldsymbol{h}_{i,M}-\boldsymbol{w}_{M,i-1}) \\ &= E(\boldsymbol{h}_{M,i}-\boldsymbol{w}_{M,i-1})^{\mathrm{T}}\left(2\sigma^4_u I_M+M\sigma^4_u I_M\right)(\boldsymbol{h}_{M,i}-\boldsymbol{w}_{M,i-1}) \\ &= (M+2)\sigma^4_u E||\boldsymbol{h}_{M,i}-\boldsymbol{w}_{M,i-1}||^2. \end{aligned} \tag{4.115}$$

Using the EMSE equivalence in (4.113), we have

$$E||\boldsymbol{u}_{M,i}||^2\boldsymbol{e}^2_{a,M}(i) = (M+2)\sigma^2_u E\boldsymbol{e}^2_{a,M}(i), \tag{4.116}$$

in which the EMSE $E\boldsymbol{e}^2_{a,M}(i)$ is obtained from $E||\boldsymbol{u}_{M,i}||^2\boldsymbol{e}^2_{a,M}(i)$ by a constant factor.

5 CONCLUSION

We presented in this text several analyses of the effects of the filter length on the performance of an adaptive filter, in particular, for the LMS and the ϵ-NLMS algorithms.

For stationary environments, and considering exponential impulse response, we showed that by increasing the filter length we reduce the steady-state error although degrading the convergence rate. For a fair comparison between filters of different lengths, it is important to remark that we adjust the step size to have a constant ratio to the optimal step size μ^o (that maximizes convergence rate).

By increasing the number of filter taps from some point on, however, we obtain almost no improvement in the steady-state performance, which means that, for lengths larger than a certain value, we only degrade the convergence rate. And we proposed a method to find this value of length, by defining and optimizing the overall performance criterion η, which is the ratio between the absolute value of the initial difference of the MSD curve and the steady-state MSD. We showed in simulations that the filter with the length M^o, that maximizes η, has indeed such an optimized overall performance: it has almost the same steady-state performance as longer filters, while exhibiting the fastest convergence rate. A filter with such a length also has reduced complexity when compared to filters with similar performance and larger length.

A filter that maximizes η, and designed with the step size that maximizes the convergence rate, is particularly interesting to be used as the fast filter within a combination. As presented in simulations, while the steady-state is ensured by a slow filter, a combination with such a fast filter has the best convergence rate when compared to combinations whose fast filter has any other length, in addition to the reduced complexity.

We also analyzed the performance of undermodeled filters in nonstationary scenarios. When the impulse response is time-varying, a shorter filter may exhibit both better convergence rate and better steady-state performance. We also derived the step size that minimizes the steady-state error, and showed that, under certain conditions, it does not exist, that is, the steady-state error monotonically decreases (converging to a nonzero positive value) as we reduce the step size towards zero.

As the reader must have noticed, we did not compare the length design proposed in this text to any work in the literature, and this is because we did not find works that are conceptually similar to the proposed method. Works that propose length selection [11,23] aim at the minimization of the EMSE, and do not consider LMS-like algorithms. The proposed method does not deal explicitly with the minimization of the EMSE, and rather focuses on the enhancement of the convergence rate. Works that might be comparable are the variable length algorithms [8,13–18], in the sense that these algorithms usually improve the transient performance by using shorter filters while the algorithm is converging.

As this text is being written, we are also working in some improvements of the results in Section 3.3 for a journal paper. Specifically, the method proposed in this text to design the filter length requires a given value of normalized step size μ_0, which also must be chosen somehow. So, among other improvements, we are deriving a method to design both length and step size of the filter given a specified performance in steady-state error.

The ideas discussed in this text that still leave some unexplored spots, and that can be further investigated in future works, are:

1. The comparison of the proposed length method with other works in the literature, such as the variable length algorithms;

2. The design of the filter length for impulse responses with other shapes than the exponential model, that are meaningful in practice (such as sparse ones);

3. The extension of the derivations (analyses for both stationary and nonstationary, and for length design) to other relevant algorithms, such as RLS, APA (affine projection algorithm), among others;

4. The design of the filter length for the nonstationary case, either by minimizing the steady-state error, as we experimentally show, or by considering the transient performance;

5. The extension of the length design to a variable length algorithm: as we can estimate the exponential decay rate with an adaptive filter — the own filter running the algorithm, we can adaptively adjust the filter length to changes in the decay rate of the impulse response.

REFERENCES

[1] B. Widrow and S. D. Stearns, *Adaptive Signal Processing*, Prentice Hall, 1985.

[2] S. Haykin, *Adaptive Filter Theory*, Prentice Hall, 3rd edition, 1996.

[3] P. S. R. Diniz, *Adaptive Filtering: Algorithms and Practical Implementation*, Springer, 3rd edition, 2008.

[4] V. H. Nascimento and M. T. M. Silva, "Adaptive filters," in *Academic Press Library in Signal Processing, Vol. 1, Signal Processing Theory and Machine Learning*, R. Chellappa and S. Theodoridis, Eds., pp. 619–761. Chennai: Academic Press, 2014.

[5] H. Hjalmarsson and L. Ljung, "Estimating model variance in the case of undermodeling," *IEEE Transactions on Automatic Control*, vol. 37, no. 7, pp. 1004–1008, 1992.

[6] M. Knudsen, "Determination of parameter estimation errors due to noise and undermodelling," in *IEEE Instrumentation and Measurement Technology Conference*, 1996, pp. 288–293.

[7] J. Homer, R. R. Bitmead, and I. Mareels, "Quantifying the effects of dimension on the convergence rate of the LMS adaptive FIR estimator," *IEEE Transactions on Signal Processing*, vol. 46, no. 10, pp. 2611–2615, 1998.

[8] Y. Gu, K. Tang, and H. Cui, "Convergence analysis of a deficient-length LMS filter and optimal-length sequence to model exponential decay impulse response," *IEEE Signal Processing Letters*, vol. 10, no. 1, pp. 4–7, January 2003.

[9] K. Mayyas, "Performance analysis of the deficient length LMS adaptive algorithm," *IEEE Transactions on Signal Processing*, vol. 53, no. 8, pp. 2727–2734, August 2005.

[10] K. Mayyas, "Performance analysis of selective coefficient update NLMS algorithm in an undermodeling situation," *Digital Signal Processing*, vol. 23, no. 8, pp. 1967–1973, August 2013.

[11] W. S. Hodgkiss, "Selecting the length of an adaptive transversal filter," in *ICASSP*, 1978, vol. 3, pp. 96–99.

[12] S. C. Douglas and M. Rupp, "Convergence issues in the LMS adaptive filter," in *Digital Signal Processing Handbook CRCnetBASE*, V. K. Madisetti and D. B. Willians, Eds. CRC Press, 1999.

[13] Z. Pritzker and A. Feuer, "Variable length stochastic gradient algorithm," *IEEE Transactions on Signal Processing*, vol. 39, no. 4, pp. 997–1001, April 1991.

[14] V. H. Nascimento, "Improving the initial convergence of adaptive filters: variable-length LMS algorithms," in *14th International Conference on Digital Signal Processing Proceedings (DSP 2002)*, 2002, pp. 667–670.

[15] Y. Gu, K. Tang, and H. Cui, "LMS algorithm with gradient descent filter length," *IEEE Signal Processing Letters*, vol. 11, no. 3, pp. 305–307, March 2004.

[16] Y. Gong and C. F. N. Cowan, "An LMS style variable tap-length algorithm for structure adaptation," *IEEE Transactions on Signal Processing*, vol. 53, no. 7, pp. 2400–2407, July 2005.

[17] Y. Zhang and J. A. Chambers, "Convex combination of adaptive filters for a variable tap-length LMS algorithm," *IEEE Signal Processing Letters*, vol. 13, no. 10, pp. 628–631, October 2006.

[18] Y. Zhang, J. A. Chambers, S. Sanei, P. Kendrik, and T. J. Cox, "A new variable tap-length LMS algorithm to model an exponential decay impulse response," *IEEE Signal Processing Letters*, vol. 14, no. 4, pp. 263–266, April 2007.

[19] C. Schuldt, F. Lindstrom, H. Li, and I. Claesson, "Adaptive filter length selection for acoustic echo cancellation," *Signal Processing*, vol. 89, pp. 1185–1194, January 2009.

[20] M. Zeller, L. A. Azpicueta-Ruiz, and W. Kellermann, "Adaptive FIR filters with automatic length optimization by monitoring a normalized combination scheme," in *2009 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, 2009, pp. 149–152.

[21] P. Stoica and Y. Selen, "Model order selection: a review of information criterion rules," *IEEE Signal Processing Magazine*, pp. 36–47, July 2004.

[22] J. Ding, V. Tarokh, and Y. Yang, "Model selection techniques — an overview," *IEEE Signal Processing Magazine*, vol. 35, pp. 16–34, November 2018.

[23] H. Clergeot, "Filter-order selection in adaptive maximum likelihood estimation," *IEEE Transactions on Information Theory*, vol. 30, no. 2, pp. 199–210, 1984.

[24] C. Breining, P. Dreiseitel, E. Hansler, A. Mader, B. Nitsch, H. Puder, T. Schertler, G. Schmidt, and J. Tilp, "Acoustic echo control - an application of very-high-order adaptive filters," *IEEE Signal Processing Magazine*, vol. 16, 1999.

[25] C. Paleologu, S. Ciochina, and J. Benesty, "Variable step-size NLMS algorithm for under-modeling acoustic echo cancellation," *IEEE Signal Processing Letters*, vol. 15, 2008.

[26] M. Failli, *COST 207: digital land mobile radio communications*, Luxemborg: Commission of the European communities, 1989.

[27] W. C. Y. Lee, *Mobile communications engineering: theory and applications*, McGraw Hill, 2nd edition, 1998.

[28] M. Martinez-Ramon, J. Arenas-Garcia, and A. R. Figueiras-Vidal, "An adaptive combination of adaptive filters for plant identification," in *14th International Conference on Digital Signal Processing Proceedings (DSP 2002)*, 2002, pp. 1195–1198.

[29] J. Arenas-Garcia, A. R. Figueiras-Vidal, and A. H. Sayed, “Mean-square performance of a convex combination of two adaptive filters,” *IEEE Transactions on Signal Processing*, vol. 54, no. 3, pp. 1078–1090, March 2006.

[30] L. F. O. Chamon, H. F. Ferro, and C. G. Lopes, “A data reusage algorithm based on incremental combination of LMS filters,” in *Asilomar Conference on Signals, Systems and Computers*. IEEE, 2012, pp. 406–410.

[31] J. Arenas-Garcia, L. A. Azpicueta-Ruiz, M. T. M. Silva, V. H. Nascimento, and A. H. Sayed, “Combinations of adaptive filters: performance and convergence properties,” *IEEE Signal Processing Magazine*, vol. 33, no. 1, pp. 120–140, January 2016.

[32] R. Claser and V. H. Nascimento, “Low complexity approximation to the Kalman filter using convex combinations of filters of adaptive filters from different families,” in *25th European Signal Processing Conference (EUSIPCO)*, 2017, pp. 2630–2633.

[33] C. G. Lopes and J. C. M. Bermudez, “Evaluation and design of variable step size adaptive algorithms,” in *ICASSP*. IEEE, 2001, pp. 3845–3848.

[34] L. F. O. Chamon, W. B. Lopes, and C. G. Lopes, “Combination of adaptive filters with coefficients feedback,” in *ICASSP*. IEEE, 2012, pp. 3785–3788.

[35] A. H. Sayed, *Fundamentals of Adaptive Filtering*, John Wiley & Sons, 2003.

[36] L. Ljung, *System identification: theory for the user*, Prentice-Hall, 1987.

[37] K. Ogata, *Modern control engineering*, Prentice Hall, 2009.

[38] C. Paleologu, J. Benesty, and S. Ciochina, “A variable step-size affine projection algorithm designed for acoustic echo cancellation,” *IEEE Transactions on Audio Speech and Language Processing*, vol. 16, 2008.

[39] V. Solo, “The limiting behavior of LMS,” *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 37, pp. 1909–1922, 1989.

[40] D. T. M. Slock, “On the convergence behavior of the LMS and the normalized LMS algorithms,” *IEEE Transactions on Signal Processing*, vol. 41, pp. 2811–2825, 1993.

[41] D. Schafhuber, M. Rupp, G. Matz, and F. Hlawatsch, “Adaptive identification and tracking of doubly selective fading channels for wireless MIMO-OFDM systems,” in *IEEE Workshop on Signal Processing Advances in Wireless Communications*, 2003.

[42] J. G. Proakis and M. Salehi, *Digital Communications*, McGraw-Hill, 5 edition, 2007.

[43] F. Reed, P. L. Feintuch, and N. J. Bershad, “Time delay estimation using the LMS adaptive filter - static behavior,” *IEEE Transactions on Acoustic, Speech and Signal Processing*, vol. 29, no. 3, pp. 561–571, March 1981.

[44] P. L. Feintuch, N. J. Bershad, and F. Reed, “Time delay estimation using the LMS adaptive filter - dynamic behavior,” *IEEE Transactions on Acoustic, Speech and Signal Processing*, vol. 29, no. 3, pp. 571–576, March 1981.

[45] L. F. O. Chamon and C. G. Lopes, “Combination of adaptive filters for relative navigation,” in *EUSIPCO*, 2011, pp. 1771–1775.

[46] B. Widrow and E. Walach, *Adaptive inverse control: a signal processing approach*, Wiley-IEEE Press, 2008.

[47] C. Komninakis, C. Fragouli, A. H. Sayed, and R. D. Wesel, "Multi-input multi-output fading channel tracking and equalization using Kalman estimation," *IEEE Transactions on Signal Processing*, vol. 50, no. 5, pp. 1065–1076, May 2002.

[48] A. Leon-Garcia, *Probability, statistics, and random processes for electrical engineering*, Prentice Hall, 3rd edition, 2008.

[49] M. B. Priestley, *Spectral analysis and time series*, Acedemic Press, 1981.

[50] Y. Gu, K. Tang, and H. Cui, "Sufficient condition for tap-length gradient adaption of LMS algorithm," in *ICASSP*, 2004, pp. 461–464.

[51] F. Zhang, *The Schur complement and its applications.*, Springer, 2005.

[52] P. Xue and B. Liu, "Adaptive equalizer using finite-bit power-of-two quantizer," *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 34, no. 6, pp. 1603–1611, December 1986.

[53] E. Eweda, "Convergence analysis and design of an adaptive filter with finite-bit power-of-two quantizer error," *IEEE Transactions on Circuits and Systems - II: Analog and Digital Signal Processing*, vol. 39, no. 2, pp. 113–115, February 1992.

[54] L. A. Azpicueta-Ruiz, A. R. Figueiras-Vidal, and J. Arenas-Garcia, "A normalized adaptation scheme for the convex combination of two adaptive filters," in *ICASSP*. IEEE, 2008, pp. 3301–3304.

[55] C. G. Lopes, E. H. Satorius, P. Estabrook, and A. H. Sayed, "Adaptive carrier tracking for Mars to Earth communications during entry, descent and landing," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 46, no. 4, pp. 1865–1879, 2010.

[56] V. H. Nascimento, M. T. M. Silva, L. A. Azpicueta-Ruiz, and J. Arenas-Garcia, "On the tracking performance of combinations of least mean squares and recursive least squares adaptive filters," in *ICASSP*. IEEE, 2010, pp. 3710–3713.

[57] V. H. Nascimento, M. T. M. Silva, R. Candido, and J. Arenas-Garcia, "A transient analysis for the convex combination of adaptive filters," in *2009 IEEE Workshop on Statistical Signal Processing*. IEEE, 2009, pp. 53–56.

[58] A. Feuer and E. Weinstein, "Convergence analysis of LMS filters with uncorrelated gaussian data," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 33, no. 1, pp. 222–230, 1985.

[59] M. Tarrab and A. Feuer, "Convergence and performance analysis of the normalized LMS algorithm with uncorrelated gaussian data," *IEEE Transactions on Information Theory*, vol. 34, no. 4, pp. 680–691, 1988.

[60] P. A. Regalia and M. Mboup, "Undermodeled adaptive filtering: an a priori error bound for the Steiglitz-McBride method," *IEEE Transactions on Circuits and Systems II*, vol. 43, no. 2, pp. 105–116, 1996.

[61] A. P. Liavas and P. A. Regalia, "Acoustic echo cancellation: do IIR models offer better modeling capabilities than their FIR counterparts?," *IEEE Transactions on Signal Processing*, vol. 46, no. 9, pp. 2499–2504, 1998.

[62] P. M. S. Burt and P. A. Regalia, "Adaptive IIR filtering: convergence speed properties in the undermodelled case," in *ICASSP*. IEEE, 2005, pp. (IV)45–48.

[63] U. Mackenroth, *Robust Control Systems: Theory and Case Studies*, Springer, 2004.

[64] M. Kendall and A. Stuart, *The advanced theory of statistics*, vol. 1, Charles Griffin, 1977.

[65] L. A. Azpicueta-Ruiz, J. Arenas-Garcia, V. H. Nascimento, and M. T. M. Silva, "Reduced-cost combination of adaptive filters for acoustic echo cancellation," in *2014 International Telecommunications Symposium (ITS)*, 2014, pp. 1–5.

[66] A. Gonzalo-Ayuso, M. T. M. Silva, V. H. Nascimento, and J. Arenas-Garcia, "Improving sparse echo cancellation via convex combination of two NLMS filters with different lengths," in *IEEE 2012 IEEE International Workshop on Machine Learning for Signal Processing (MLSP)*, 2012.

[67] L. Isserlis, "On a formula for the product-moment coefficient of any order of a normal frequency distribution in any number of variables," *Biometrika*, vol. 12, pp. 134–139, 1918.

[68] N. J. Bershad, "Analysis of the normalized LMS algorithm with gaussian inputs," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 34, pp. 793–806, 1986.

[69] S. M. Kay, *Fundamentals of statistical signal processing: estimation theory*, Prentice Hall, 1993.

[70] R. B. Wallace and R. A. Goubran, "Improved tracking adaptive noise canceler for nonstationary environments," *IEEE Transactions on Signal Processing*, vol. 40, no. 3, pp. 700–703, March 1992.

[71] M. K. Tsatsanis, G. B. Giannakis, and G. Zhou, "Estimation and equalization of fading channels with random coefficients," *Signal Processing*, vol. 53, pp. 211–229, 1996.

[72] B. Widrow, J. M. McCool, M. G. Larimore, and C. R. Johnson Jr., "Stationary and nonstationary learning characteristics of the LMS adaptive filter," *Proceedings of the IEEE*, vol. 64, no. 8, pp. 1151–1162, August 1976.

[73] S. Haykin, A. H. Sayed, J. R. Zeidler, P. Yee, and P. C. Wei, "Adaptive tracking of linear time-variant systems by extended RLS algorithms," *IEEE Transactions on Signal Processing*, vol. 45, no. 5, pp. 1118–1128, May 1997.

[74] T. Y. Al-Naffouri and M. Moinuddin, "Exact performance analysis of the ϵ-NLMS algorithm for colored circular Gaussian inputs," *IEEE Transactions on Signal Processing*, vol. 58, no. 10, pp. 5080–5090, 2010.

[75] R. Claser and V. H. Nascimento, "On the tracking performance of adaptive filters and their combinations," *IEEE Transactions on Signal Processing*, vol. 69, no. 5, pp. 3104–3116, May 2021.