Blind equalization based on pdf fitting and convergence analysis

Souhaila Fki, Malek Messai, Abdeldjalil Aissa El Bey, Thierry Chonavel

Blind Equalization Based on PDF Fitting and Convergence Analysis

Souhaila Fki, Malek Messai, Abdeldjalil Aïssa-El-Bey, and Thierry Chonavel

Institut Télécom; Télécom Bretagne; UMR CNRS 6285 Lab-STICC
Technopôle Brest Iroise CS 83818 29238 Brest, France
Université européenne de Bretagne
Email: Firstname.Lastname@telecom-bretagne.eu

Abstract

In this paper, we address M-QAM blind equalization by fitting the probability density functions (pdf) of the equalizer output with the constellation symbols. We propose two new cost functions, based on kernel pdf approximation, which force the pdf at the equalizer output to match the known constellation pdf. The kernel bandwidth of a Parzen estimator is updated during iterations to improve the convergence speed and to decrease the residual error of the algorithms. Unlike related existing techniques, the new algorithms measure the distance error between observed and assumed pdfs for the real and imaginary parts of the equalizer output separately. The advantage of proceeding this way is that the distributions show less modes, which facilitates equalizer convergence, while as for multi-modulus methods phase recovery keeps being preserved. The proposed approaches outperform CMA and classical pdf fitting methods in terms of convergence speed and residual error. We also analyse the convergence properties of the most efficient proposed equalizer via the ordinary differential equation (ODE) method.

Preprint submitted to Signal Processing *May 7, 2014*

Keywords: Blind equalization, pdf, Parzen windowing, Performance analysis, ODE

1. Introduction

In transmissions, multipath propagation introduces intersymbols interference (ISI) that can make it difficult to recover transmitted data. Thus, an equalizer can be used to reduce the ISI. Time domain equalization or alternatively frequency domain equalization can be considered. The latter is very interesting for broadband wireless communications. Indeed, for long channels it is computationally simpler than the corresponding time domain equalization with the same efficiency. However, in this paper, we consider multipath channels with length less than or equal to about 10 and time domain equalization is well suited in this case because then both approaches have approximately the same complexity and the same performance [1]. Moreover, in this paper, we are interested in continuous flow transmission rather than block transmission where frequency-domain equalization is more relevant.
Without knowledge of the channel, the first equalization methods rely on periodic transmission of training sequences that are known from the receiver. Then, adaptation of the equalizer coefficients is done by minimizing a cost function that measures some distance between the actual equalizer output and the desired reference signal. When the transmitter sends a training sequence, the equalizer taps can be easily adapted by using a stochastic optimization technique such as the Least Mean Squares (LMS) algorithm, the cost function of which minimizes the expectation of the squared error [2]. However, in many digital communication systems, the transmission of

a bandwidth consuming training sequence is not suitable. In order to avoid training, blind equalization techniques have been developped to retrieve symbols transmitted through an unknown channel by only using received data and some knowledge upon the statistics of the original sequence. There exist many blind algorithms. Sato algorithm [3] was the first blind technique proposed. The Godard algorithm [4] and the Constant Modulus Algorithm (CMA) [5] which is a particular case of Godard algorithm, are probably the most popular blind equalization techniques. However, they require a long data sequence to converge and show relatively high residual error. To overcome these limitations, several approaches have been proposed in the literature. For instance, we can mention the Normalized-CMA (NCMA), that accelerates convergence by estimating the optimal step size of the algorithm at each iteration [6]. The CMA with a gain stage, where the latter is inserted after the equalizer to control the behaviour of its output power for faster convergence, was proposed in [7]. The square contour algorithm minimizes dispersion of the equalizer output from a square for blind equalization of QAM modulations [8], while the regional multimodulus algorithm, also designed for QAM modulations, performs similar to the supervised normalized least-mean-squares algorithm [9]. The Modified Constant Modulus Algorithm (MCMA), also known as Multi-Modulus Algorithm (MMA) performs blind equalization and carrier phase recovery simultaneously [10], by measuring the errors of real and imaginary parts of the equalizer output separately. The minℓ_1-MMA and MGauss-MMA algorithms [11] outperform the MMA by combining the multi-modulus criterion and an alphabet-matching penalty term.

In the last decade, new blind equalization techniques, based on information theoretic criteria and pdf estimation of transmitted data, have been proposed. These criteria are optimized adaptively, in general by means of stochastic gradient techniques. Among these techniques, Renyi's entropy has been used as a cost function [12]. It involves pdf estimation with the Parzen window kernel method. This equalizer is very sensitive to noise and provides excellent results for some channels but fails to equalize some others. So, an alternative criterion based on forcing the pdf at the equalizer output to match the known constellation pdf has been proposed in [13]. As a cost function, it uses the Kullback-Leibler Divergence (KLD) between the pdfs. The Euclidean distance has also been proposed in [14]. It uses Parzen window with Gaussian kernels for pdf estimation. In [15], a technique based on fitting the pdf of the equalizer output at some relevant points that are determined by the modulus of the constellation symbols was proposed. It is known as sampled-pdf fitting. The authors of [15] also proposed in [16] the Stochastic blind equalization approach that uses the Quadratic Distance (SQD) between the pdf at the equalizer output and the known constellation pdf as a cost function. This method is designed for multilevel modulations and works at symbol rate. Many digital transmission systems with a high number of states use QAM modulations. As the multi-modulus approaches are well suited for such modulations, we propose to use these techniques to equalize QAM constellations. Therefore, in this paper, we propose a new family of blind algorithms based on the SQD fitting, that we call Multi-Modulus SQD-ℓp (MSQD-ℓp). Unlike the method in [15], MSQD-ℓp measures the distance error between observed and assumed pdfs for real and imaginary parts of the

equalizer output separately. The advantage of proceeding this way is that involved distributions show less modes, leading thus to reduced complexity, while preserving phase recovery as for multi-modulus methods. In addition, we benefit from the fact that 1D pdfs can be accurately estimated with less data than 2D pdfs. In this paper, we are particularly interested in the case $p = 1$ that leads to the MSQD-ℓ1 algorithm that involves the absolute values of the real and the imaginary parts of the equalizer output. Thus, the shape of equalized constellation modes is Gaussian which is in accordance with the statistical behavior of received data from a single path propagation, what the equalizer tries to achieve.

These techniques are designed for multilevel modulations, work at the symbol rate and admit a simple stochastic gradient-based implementation. For pdf estimation, we use the Parzen window. The proposed methods outperform CMA and classical pdf fitting approaches, in terms of convergence speed and residual error. As much as possible, it is interesting to analyze the convergence properties of blind equalizers to better understand their performance. In this paper, we focus on performance analysis of the MSQD-ℓ1. To this goal, we employ the Ordinary Differential Equation (ODE) method. Indeed, the ODE approach supplies a solid theoretical framework for such a task [17]. The exact convergence analysis of adaptive blind equalization algorithms is often difficult because they are derived from nonlinear criteria. Therefore, the convergence analysis of the MSQD-ℓ1 is conducted under some usual assumptions that are commonly met in the related literature. The contributions of this paper to the field of blind equalization include:

1. A new family of blind equalization algorithms named MSQD-ℓp that

converges faster than the CMA and the classical SQD pdf fitting [16] and achieves lower residual error;

2. Convergence and performance analysis of the most effective MSQD-$\ell 1$ algorithm based on the ODE method.

This paper is organized as follows. In section 2, we present the blind equalization problem and the SQD pdf fitting method. In section 3, we propose the new cost functions and their corresponding stochastic gradient expressions. The convergence and performance analysis of the MSQD-$\ell 1$ algorithm is developed in section 4. Simulations are presented in section 5 and conclusions of our work are given in section 6.

2. SIGNAL AND EQUALIZER MODEL

2.1. Signal model

To transmit digital data, a sequence $\{\boldsymbol{s}(n)\}_{n\in\mathbb{Z}}$ of independent identically distributed (i.i.d) complex symbols belonging to a digital modulation constellation is sent through a channel of length L_h with impulse response $\boldsymbol{h} = [h_0, h_1, ..., h_{L_h-1}]^T$, where $(.)^T$ denotes the transpose operater. Transmitted data are affected by multipath propagation, resulting in intersymbol interference (ISI) at the receiver side. Therefore, an equalizer is used to reduce this ISI. In our work, we are interested in blind equalization that only requires knowledge of the modulation used to send the data. The basic scheme of a blind equalization system is described in Fig.1. We assume that $\{\boldsymbol{s}(n)\}_{n\in\mathbb{Z}}$ is drawn from a symmetric QAM constellation.

Fig.1 summarizes the transmission model, where $b = \{b(n)\}_{n\in\mathbb{Z}}$ is a circular complex additive white Gaussian noise, independent from s with

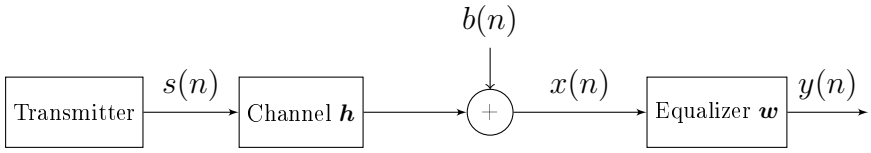

Figure 1: Basic scheme of a blind equalization system

variance $\sigma_b^2 = \mathbb{E}\left[|b(n)|^2\right]$, $x = \{x(n)\}_{n\in\mathbb{Z}}$ is the equalizer input, $\boldsymbol{w}(n) = [w_0(n), w_1(n), ..., w_{L_w-1}(n)]^T$ is the equalizer impulse response, with length L_w and $y(n)$ is the equalized signal at time n. $x(n)$ and $y(n)$ can be modeled as

$$x(n) = \sum_{i=0}^{L_h-1} h_i\, s(n-i) + b(n) \tag{1}$$

and

$$y(n) = \sum_{i=0}^{Lw-1} w_i(n)\, x(n-i) = \boldsymbol{w}(n)^T \boldsymbol{x}(n) \tag{2}$$

where $\boldsymbol{x}(n) = [x(n), x(n-1), ..., x(n-L_w+1)]^T$.
The weights of the equalizer will be adapted by using a gradient stochastic algorithm in the form

$$\boldsymbol{w}(n+1) = \boldsymbol{w}(n) - \mu\, \nabla_{\boldsymbol{w}} J(\boldsymbol{w}) \tag{3}$$

where μ is the step size and $J(\boldsymbol{w})$ is the cost function to be minimized. In order to increase the performance of the equalizer, in this paper, we propose new versions of $J(\boldsymbol{w})$ derived from the SQD pdf fitting technique [16].

2.2. SQD pdf fitting using Parzen Estimator [16]

Equalization techniques based on pdf matching intend to minimize some distance between the data distribution at the equalizer output and some target distribution. Transmitted symbols have a discrete distribution. But,

since they are affected by additive Gaussian noise at the receiver side, it can be assumed that after removing channel multipath effects, the equalizer output should consist of a Gaussian mixture, with Gaussian modes centered at the constellation points. Therefore, a target distribution of this form can be chosen. Then, the stochastic blind equalization based on the Parzen window method forces the pdf of the equalizer output to match the target pdf. In [16], a quadratic distance between pdfs for the cost function was proposed. It is given by

$$J(\boldsymbol{w}) = \int_{-\infty}^{\infty} \left(f_{Y^p}(z) - f_{S^P}(z)\right)^2 dz \tag{4}$$

where, $Y^p = \{|y(n)|^p\}$ and $S^p = \{|s_k|^p\}$ are the sets of the moduli to the power p of equalized symbols and constellation symbols and $f_Z(z)$ denotes the pdf of Z at z. Thus, $J(\boldsymbol{w})$ is intended to match p^{th} moment distributions between the equalizer output and the noisy constellation.

The Parzen window method is used to estimate the current data pdf. Using this nonparametric pdf estimator with the L last symbols, the estimates of the pdfs at time n are given by:

$$\begin{aligned} \hat{f}_{Y^p}(z) &= \frac{1}{L}\sum_{k=0}^{L-1} K_{\sigma_0}(z - |y(n-k)|^p) \\ \hat{f}_{S^p}(z) &= \frac{1}{N_s}\sum_{k=1}^{N_s} K_{\sigma_0}(z - |s_k|^p) \end{aligned} \tag{5}$$

where N_s is the number of complex symbols in the constellation and K_{σ_0} is a Gaussian kernel with standard deviation σ_0, also known as the kernel bandwidth:

$$K_{\sigma_0}(x) = \frac{1}{\sqrt{2\pi}\sigma_0} e^{-\frac{x^2}{2\sigma_0^2}}. \tag{6}$$

According to [16], for $p = 2$ and $L = 1$, the expression of the cost function is given by

$$J(\boldsymbol{w}) = \frac{1}{N_s^2} \sum_{k=1}^{N_s} \sum_{l=1}^{N_s} K_\sigma(|s(l)|^2 - |s(k)|^2) - \frac{2}{N_s} \sum_{k=1}^{N_s} K_\sigma(|y(n)|^2 - |s(k)|^2). \quad (7)$$

where, $\sigma = \sqrt{2}\sigma_0$. Then, the gradient of $J(\boldsymbol{w})$ with respect to the equalizer weights is given by

$$\nabla_{\boldsymbol{w}} J(\boldsymbol{w}) = -\frac{1}{N_s} \sum_{k=1}^{N_s} K_\sigma^{'}(|y(n)|^2 - |s(k)|^2)\, y(n)\, \boldsymbol{x}^*(n) \quad (8)$$

where $K_\sigma^{'}(x) = -\frac{x}{\sqrt{2\pi\sigma^3}} \exp(\frac{-x^2}{2\sigma^2})$ is the derivative of $K_\sigma(x)$ and $(.)^*$ denotes the complex conjugation operator. Then, the equalizer coefficients are updated at symbol rate by inserting Eq.(8) in Eq.(3). This algorithm is initialized with a tap-centered equalizer. In [16], the squared modulus of the symbols for the kernel variables ($p = 2$) is used to design $J(\boldsymbol{w})$. But, squaring does not preserve Gaussianity around noisy constellation points. Fig.4 illustrates this fact for $p = 2$: clearly the Gaussianity of modes is not preserved for $p = 2$ while it is for $p = 1$. Then, with a view to make the criterion statistically more meaningful we propose, in this paper, to also address the case $p = 1$. Indeed, when $p = 1$, since constellation points are apart from the axes, at convergence $|y(n)|$ will be roughly distributed according to a mixture of Gaussian distributions around the constellation points that are in the positive quadrant of the complex plane. This is true provided the SNR remains in usual ranges for QAM modulations under consideration. In addition, it is well known that multimodulus approaches such as MMA [18], that decompose equalization criteria into an in-phase term and a quadrature one, are more efficient than criteria such as the CMA [5], that handle in-phase

and quadrature parts together. In the same way, the criteria that we propose are made of a sum of two terms related to in-phase and quadrature parts of the equalizer output. This will lead to criteria that we name Multimodulus SQD-ℓp (MSQD-ℓp). The advantage of proceeding this way is that involved distributions show less modes, leading thus to reduced complexity, while preserving phase recovery. In addition, we benefit from the fact that 1D pdfs can be accurately estimated with less data than 2D pdfs as show Fig.2 and Fig.3. Indeed, with the SQD algorithm there are M symbols involved in the target pdf whereas with the MSQD-ℓp algorithms there are only $2\sqrt{M}$ modes involved, for an M-QAM modulation.

3. MSQD algorithms

3.1. MSQD-ℓp algorithm

MSQD family consists of algorithms based on cost functions in the form:

$$J(\boldsymbol{w}) = \int_{-\infty}^{\infty} (\hat{f}_{|y_r|^p}(z) - \hat{f}_{|s_r|^p}(z))^2 dz + \int_{-\infty}^{\infty} (\hat{f}_{|y_i|^p}(z) - \hat{f}_{|s_i|^p}(z))^2 dz \qquad (9)$$

where $y_r = \Re\{y\}$, $y_i = \Im\{y\}$ and the pdf estimates are in the form

$$\hat{f}_x(z) = \frac{1}{N_x} \sum_{k=1}^{N_x} K_{\sigma_0}(z - x_k) \qquad (10)$$

x is equal to $|s_r|^p$, $|s_i|^p$, $|y_r|^p$ or $|y_i|^p$. $N_x = N_s$ for $x = |s_{r,i}|^p$ and $N_x = L$ for $x = |y_{r,i}|^p$.

For fixed p, we denote the corresponding criterion by MSQD-ℓp. Expending Eq.(9), we get

$$\begin{aligned} J(\boldsymbol{w}) &= \int_{-\infty}^{\infty} \hat{f}_{|y_r|^p}(z)^2\, dz - 2\int_{-\infty}^{\infty} \hat{f}_{|y_r|^p}(z)\, \hat{f}_{|s_r|^p}(z)\, dz + \int_{-\infty}^{\infty} \hat{f}_{|s_r|^p}(z)^2\, dz \\ &+ \int_{-\infty}^{\infty} \hat{f}_{|y_i|^p}(z)^2 dz - 2\int_{-\infty}^{\infty} \hat{f}_{|y_i|^p}(z) \hat{f}_{|s_i|^p}(z)\, dz + \int_{-\infty}^{\infty} \hat{f}_{|s_i|^p}(z)^2\, dz \quad (11) \end{aligned}$$

In a stochastic gradient optimization approach, in general only instantaneous statistics are involved in the criterion. Thus, we consider a window length $L = 1$ as in [16]. Then, since for Gaussian kernels we have

$$\int_{-\infty}^{\infty} K_{\sigma_0}(y - C_1) K_{\sigma_0}(y - C_2) dy = \frac{1}{2} K_{\sigma_0\sqrt{2}}(C_1 - C_2), \tag{12}$$

thus, according to Eq.(10) and Eq.(11), $J(\boldsymbol{w})$ becomes

$$J(\boldsymbol{w}) = -\frac{1}{N_s}\sum_{k=1}^{N_s} K_\sigma(|y_r(n)|^p - |s_r(k)|^p) - \frac{1}{N_s}\sum_{k=1}^{N_s} K_\sigma(|y_i(n)|^p - |s_i(k)|^p) + Cst. \tag{13}$$

On another hand, $y(n) = \boldsymbol{w}(n)^T \boldsymbol{x}(n)$ rewrites as

$$y(n) = [\boldsymbol{w}_r^T \boldsymbol{x}_r(n) - \boldsymbol{w}_i^T \boldsymbol{x}_i(n)] + j[\boldsymbol{w}_r^T \boldsymbol{x}_i(n) + \boldsymbol{w}_i^T \boldsymbol{x}_r(n)], \tag{14}$$

which leads to $\frac{\partial y(n)}{\partial \boldsymbol{w}_r} = \boldsymbol{x}(n)$ and $\frac{\partial y(n)}{\partial \boldsymbol{w}_i} = j\,\boldsymbol{x}(n)$.
Therefore, the derivative of $J(\boldsymbol{w})$ with respect to equalizer weights is

$$\begin{aligned}
\nabla_{\boldsymbol{w}} J(\boldsymbol{w}) &= \frac{\partial J(\boldsymbol{w})}{\partial \boldsymbol{w}_r} + j\,\frac{\partial J(\boldsymbol{w})}{\partial \boldsymbol{w}_i} \\
&= \frac{p}{2\sqrt{2\pi}N_s\sigma^3}\sum_{k=1}^{N_s}\Big(\operatorname{sign}(y_r(n))\,|y_r(n)|^{p-1}\,(|y_r(n)|^p - |s_r(k)|^p)\; e^{-\frac{(|y_r(n)|^p - |s_r(k)|^p)^2}{2\sigma^2}} \\
&+ j\operatorname{sign}(y_i(n))\,|y_i(n)|^{p-1}\,(|y_i(n)|^p - |s_i(k)|^p)\; e^{-\frac{(|y_i(n)|^p - |s_i(k)|^p)^2}{2\sigma^2}}\Big)\boldsymbol{x}^*(n).
\end{aligned} \tag{15}$$

3.2. MSQD-$\ell 2$ and MSQD-$\ell 1$ algorithms

We consider first the case $p = 2$ since $\ell 2$ norm is often considered in the literature and for comparison to the case $p = 1$ in the simulation part. From Eq.(15) we get the gradient of the MSQD-$\ell 2$ cost function:

$$\begin{aligned}
\nabla_{\boldsymbol{w}} J(\boldsymbol{w}) &= \frac{1}{\sqrt{2\pi}N_s\sigma^3}\sum_{k=1}^{N_s}\Big(y_r(n)(|y_r(n)|^2 - |s_r(k)|^2)\, e^{-\frac{(|y_r(n)|^2 - |s_r(k)|^2)^2}{2\sigma^2}} \\
&+ j\,y_i(n)(|y_i(n)|^2 - |s_i(k)|^2)\, e^{-\frac{(|y_i(n)|^2 - |s_i(k)|^2)^2}{2\sigma^2}}\Big)\boldsymbol{x}^*(n)
\end{aligned} \tag{16}$$

Then, Eq.(3) is used to update equalizer taps. For the case $p = 1$, that is statistically more meaningful, as discussed in section 2, we get an updating term of the equalizer in the form:

$$\begin{aligned}\nabla_{\boldsymbol{w}} J(\boldsymbol{w}) &= \frac{1}{2\sqrt{2\pi}N_s\sigma^3}\sum_{k=1}^{N_s}\Big(\operatorname{sign}(y_r(n))\,(|y_r(n)| - |s_r(k)|)\,e^{-\frac{(|y_r(n)|-|s_r(k)|)^2}{2\sigma^2}} \\ &+ j\operatorname{sign}(y_i(n))\,(|y_i(n)| - |s_i(k)|)\,e^{-\frac{(|y_i(n)|-|s_i(k)|)^2}{2\sigma^2}}\Big)\boldsymbol{x}^*(n) \\ &= \phi(y(n))\,\boldsymbol{x}^*(n) \end{aligned} \tag{17}$$

In section 5, we will show on simulations that, as expected from the discussion at the end of section 2, the MSQD-ℓ1 algorithm that we propose is more effective than the existing SQD algorithm in terms of mean square error, especially for larger constellations. Thus, in the following section, we will restrict our interest to performance analysis of the MSQD-ℓ1 algorithm in terms of stationary stable points and asymptotic steady state.

4. Performance Analysis

4.1. The ODE method

Part of the following analysis is based on the ODE method, the principle of which we briefly recall here. We consider a stochastic gradient algorithm in the form

$$\boldsymbol{\theta}(n) = \boldsymbol{\theta}(n-1) + \mu_n H(\boldsymbol{\theta}(n-1), \boldsymbol{x}(n)) \tag{18}$$

where, $\boldsymbol{\theta}(n)$ is the sequence of estimated parameters and the observation process, $\{\boldsymbol{x}(n)\}_{n\in\mathbb{N}}$ is assumed to be a Markov and ergodic process independent of $\boldsymbol{\theta}$, $\{\mu_n\}_{n\in\mathbb{N}^*}$ is a series of small scalar gains and $H(\boldsymbol{\theta}(n-1), \boldsymbol{x}(n))$ is the function which defines how $\boldsymbol{\theta}(n)$ is adapted based on new observations.

The ODE method associates an ordinary differential equation to the adaptation criterion and is defined by:

$$\dot{\boldsymbol{\theta}} = h(\boldsymbol{\theta}) = \lim_{n \to +\infty} \mathbb{E}\big[H(\boldsymbol{\theta}, \boldsymbol{x}(n)) \,|\boldsymbol{\theta}\big] \tag{19}$$

By assuming a regular behaviour of the function H, it was shown in [19] that the behaviour of the stochastic algorithm Eq.(18) is linked to that of the ODE Eq.(19).

4.2. Stationary Stable Points of the ODE

In this section, we denote by $\boldsymbol{\theta}(t)$ the solution of the ODE. It depends on the initial value $\boldsymbol{\theta}(0) = \boldsymbol{\theta}_0$.
A point $\boldsymbol{\theta}_*$ is a stationary point of the ODE if $h(\boldsymbol{\theta}_*) = 0$.

4.2.1. Stationary points

Let us consider the stochastic MSQD-ℓ1 algorithm:

$$\boldsymbol{w}(n+1) = \boldsymbol{w}(n) - \mu \nabla_{\boldsymbol{w}} J\big(\boldsymbol{w}(n), \boldsymbol{x}(n)\big). \tag{20}$$

Letting $\boldsymbol{\theta}(n) = \boldsymbol{w}(n)$, identifying Eq.(20) and Eq.(18) leads to

$$H\big(\boldsymbol{\theta}(n), \boldsymbol{x}(n+1)\big) = -\nabla_{\boldsymbol{w}} J\big(\boldsymbol{w}(n), \boldsymbol{x}(n)\big) = -\nabla_{\boldsymbol{w}} J\big(\boldsymbol{w}, y(n)\big) = -\phi\big(y(n)\big)\, \boldsymbol{x}^*(n). \tag{21}$$

Then, the ODE is defined by

$$\begin{aligned} \frac{d\boldsymbol{\theta}}{dt} = h(\boldsymbol{w}) &= \lim_{n \to +\infty} \mathbb{E}\big[H(\boldsymbol{w}, \boldsymbol{x}(n)) | \boldsymbol{w}\big] \\ &= \lim_{n \to +\infty} -\mathbb{E}\big[\nabla_{\boldsymbol{w}} J(\boldsymbol{w}, y(n)) | \boldsymbol{w}\big] \end{aligned} \tag{22}$$

The stationary points $\boldsymbol{w}_*$ of the ODE are the solutions of $h(\boldsymbol{w}_*) = 0$. Then we begin by calculating $h(\boldsymbol{w})$.

$$\begin{aligned} h(\boldsymbol{w}) &= \lim_{n\to+\infty} -\mathbb{E}\big[\nabla_{\boldsymbol{w}} J(\boldsymbol{w}, y(n))|\boldsymbol{w}\big] \\ &= -\int_{\mathbb{R}^+} \nabla_{\boldsymbol{w}} J(\boldsymbol{w}, y) p_{|Y|}(y)\, dy \end{aligned} \tag{23}$$

where, $p_{|Y|}(y)$ is the probability density of $|y|$. To calculate $h(\boldsymbol{w})$, we begin by calculating $F_{|Y|\,|s(k)}(y)$ where, $|y|\ |s(k) = |s(k) + \varepsilon|$ and $F_{|Y|\,|s(k)}(y)$ is the cumulative distribution of $|Y|$ given $s(k)$:

$$\begin{aligned} F_{(|Y|\,|s(k))}(y) &= P\Big(|Y| \le y\ |s(k)\Big) \\ &= P\Big(-y \le Y \le y\ |s(k)\Big) \\ &= P\Big(\frac{-y - s(k)}{\sigma_\varepsilon} \le \frac{Y - s(k)}{\sigma_\varepsilon} \le \frac{y - s(k)}{\sigma_\varepsilon}\ |s(k)\Big) \\ &= F_Y\Big(\frac{y - s(k)}{\sigma_\varepsilon}\Big) - F_Y\Big(\frac{-y - s(k)}{\sigma_\varepsilon}\Big) \end{aligned} \tag{24}$$

where F is the cumulative distribution function of the $\mathcal{N}(0,1)$ distribution. Indeed, the ISI at the output of the equalizer can be modelled as a Gaussian distribution [20] and thus $Y|_{s(k)} \sim \mathcal{N}(s(k), \sigma_\varepsilon^2)$ where σ_ε^2 is the variance of the error (ε) between y and $s(k)$. Then, from Eq.(24) and for $y \ge 0$,

$$\begin{aligned} p_{(|Y|\,|s(k))}(y) &= \frac{1}{2\sigma_\varepsilon}\Big[\mathcal{N}\Big(\frac{y - s(k)}{\sigma_\varepsilon}; 0, 1\Big) + \mathcal{N}\Big(\frac{y + s(k)}{\sigma_\varepsilon}; 0, 1\Big)\Big] \\ &= \frac{1}{2}\Big[\mathcal{N}\Big(y - s(k); 0, \sigma_\varepsilon^2\Big) + \mathcal{N}\Big(y + s(k); 0, \sigma_\varepsilon^2\Big)\Big]. \end{aligned} \tag{25}$$

where, $\mathcal{N}(y; m, \sigma^2) = \frac{1}{\sqrt{2\pi}\sigma} e^{\frac{-(y-m)^2}{2\sigma^2}}$. Summing over all possible symbols s_k, we find the following expression of $p_{|Y|}(y)$:

$$p_{|Y|}(y) = \frac{1}{2N_s} \sum_{k=0}^{N_s-1} \big[\mathcal{N}(y - s(k); 0, \sigma_\varepsilon^2) + \mathcal{N}(y + s(k); 0, \sigma_\varepsilon^2)\big]. \tag{26}$$

Then, we obtain the following expression of $h(\boldsymbol{w})$ after replacing $J(\boldsymbol{w}, y)$ by its expression and accounting for symmetry properties of $J(\boldsymbol{w}, y)$ and $p_{|Y|}(y)$,

$$h(\boldsymbol{w}) = \frac{1}{2\pi N_s^2 \sigma\sigma_\varepsilon}\Big[\nabla_{\boldsymbol{w}} \int_{\mathbb{R}^+} \sum_{k=1}^{N_s} e^{-\frac{(y_r-|s_r(k)|)^2}{2\sigma^2}} \sum_{l=1}^{N_s} \big(e^{-\frac{(y_r-|s_r(l)|)^2}{2\sigma_\varepsilon^2}} + e^{-\frac{(y_r+|s_r(l)|)^2}{2\sigma_\varepsilon^2}}\big) dy_r. \tag{27}$$

Thus, after calculating the integral,

$$\begin{aligned} h(\boldsymbol{w}) &= \frac{1}{2N_s^2\sqrt{2\pi(\sigma^2+\sigma_\varepsilon^2)}}\Big[\nabla_{\boldsymbol{w}} \sum_{k=1}^{N_s}\sum_{l=1}^{N_s} e^{-\frac{(|s_r(k)|-|s_r(l)|)^2}{2(\sigma^2+\sigma_\varepsilon^2)}}\Big(1-\mathrm{erfc}\big(\frac{-|s_r(k)|\sigma_\varepsilon^2-|s_r(l)|\sigma^2}{\sigma\sigma_\varepsilon\sqrt{\sigma_\varepsilon^2+\sigma^2}\sqrt{2}}\big)\Big) \\ &+ e^{-\frac{(|s_r(k)|+|s_r(l)|)^2}{2(\sigma^2+\sigma_\varepsilon^2)}}\Big(1-\mathrm{erfc}\big(\frac{-|s_r(k)|\sigma_\varepsilon^2+|s_r(l)|\sigma^2}{\sigma\sigma_\varepsilon\sqrt{\sigma_\varepsilon^2+\sigma^2}\sqrt{2}}\big)\Big)\Big] \\ &= \nabla_{\boldsymbol{w}} A(\sigma_\varepsilon^2) \\ &= \frac{dA(\sigma_\varepsilon^2)}{d\sigma_\varepsilon^2}\nabla_{\boldsymbol{w}}\sigma_\varepsilon^2. \end{aligned} \tag{28}$$

The equalized symbol $y(n)$ can be expressed by $y(n) = \boldsymbol{w}^T\widetilde{\boldsymbol{H}}\boldsymbol{s}(n) + \boldsymbol{w}^T\boldsymbol{b}(n)$ where,

$$\widetilde{\boldsymbol{H}} = \begin{pmatrix} h_0 & h_1 & \cdots & h_{L-1} & 0 & \cdots & & 0 \\ 0 & h_0 & h_1 & \cdots & h_{L-1} & 0 & \cdots & 0 \\ 0 & \ddots & h_0 & h_1 & \cdots & h_{L-1} & \vdots & \\ 0 & \vdots & & h_0 & h_1 & \cdots & h_{L-1} & 0 \\ 0 & & \cdots & 0 & h_0 & h_1 & \cdots & h_{L-1} \end{pmatrix} \tag{29}$$

and $\boldsymbol{s}(n) = \big[s(D-1), ..., s(k_n), ..., s(D-L-L_w+1)\big]^T$ where D is the equalizer delay. Then, the error between $y(n)$ and the value of the transmitted symbol

$s(k_n)$ received at time n is calculated as follows

$$\begin{aligned}\sigma_\varepsilon^2 &= \mathbb{E}\Big[\Big(y(n)-s(k_n)\Big)\Big(y(n)-s(k_n)\Big)^H\Big] \\ &= \mathbb{E}\Big[\Big((\boldsymbol{w}^T\widetilde{\boldsymbol{H}}-\boldsymbol{e}_D)\boldsymbol{s}(n)+\boldsymbol{w}^T\boldsymbol{b}(n)\Big)\Big((\boldsymbol{w}^T\widetilde{\boldsymbol{H}}-\boldsymbol{e}_D)\boldsymbol{s}(n)+\boldsymbol{w}^T\boldsymbol{b}(n)\Big)^H\Big]\end{aligned} \tag{30}$$

where, $\boldsymbol{e}_D=(0,...,1,...,0)^T$, $e_D(i)=\delta_{i,D}$. Then,

$$\sigma_\varepsilon^2=\sigma_s^2\boldsymbol{w}^T\widetilde{\boldsymbol{H}}\widetilde{\boldsymbol{H}}^H\boldsymbol{w}^*-\sigma_s^2\boldsymbol{w}^T\widetilde{\boldsymbol{H}}\boldsymbol{e}_D-\sigma_s^2{\boldsymbol{e}_D}^T\widetilde{\boldsymbol{H}}^H\boldsymbol{w}^*+\sigma_s^2+\sigma_{\boldsymbol{b}}^2||\boldsymbol{w}||^2 \tag{31}$$

Thus,

$$h(\boldsymbol{w})=2\frac{dA(\sigma_\varepsilon^2)}{d\sigma_\varepsilon^2}\Big[\boldsymbol{w}^T[\sigma_s^2\widetilde{\boldsymbol{H}}\widetilde{\boldsymbol{H}}^H+\sigma_{\boldsymbol{b}}^2\boldsymbol{I}_{L_w}]-\sigma_s^2{\boldsymbol{e}_D}^T\widetilde{\boldsymbol{H}}^H\Big]^T \tag{32}$$

Since in practice $|s_r(k_n)| >> \sigma^2+\sigma_\varepsilon^2$, only terms with 0 in the exponential will be non negligible, resulting in $A(\sigma_\varepsilon^2)\approx\frac{1}{2N_s^2\sqrt{2\pi(\sigma^2+\sigma_\varepsilon^2)}}\sum_{k=1}^{N_s}\big(1+\text{erfc}(\frac{|s_r(k_n)|}{\sqrt{2}}\sqrt{\frac{1}{\sigma^2}+\frac{1}{\sigma_\varepsilon^2}})\big)$. Since at convergence, $\frac{1}{\sigma^2}>>1$ and $\frac{1}{\sigma_\varepsilon^2}>>1$, it is easy to check that $\frac{dA(\sigma_\varepsilon^2)}{d\sigma_\varepsilon^2}<0$. Thereby, $\boldsymbol{w}_*^T=\sigma_s^2{\boldsymbol{e}_D}^T\widetilde{\boldsymbol{H}}^H\big[\sigma_s^2\widetilde{\boldsymbol{H}}\widetilde{\boldsymbol{H}}^H+\sigma_{\boldsymbol{b}}^2\boldsymbol{I}_{L_w}\big]^{-1}$ is the only stationnary point. The very nice thing is that $\boldsymbol{w}_*$ is the MMSE filter. Thus, we have proved that the MSQD-ℓ1 algorithm has only one stationnary point, which is the MMSE filter, when σ^2 and σ_ε^2 are much smaller than constellation point amplitudes.

4.2.2. Stability analysis of stationary points

Let us recall that if $\boldsymbol{w}_*$ is a stationary point of the ODE and $\lambda_1,\lambda_2,...\lambda_{L_w}$ are the eigenvalues of $\frac{dh(\boldsymbol{w})}{d\boldsymbol{w}}|_{\boldsymbol{w}=\boldsymbol{w}_*}$, the stability of $\boldsymbol{w}_*$ is determined by the eigenvalues of $\frac{dh(\boldsymbol{w})}{d\boldsymbol{w}}|_{\boldsymbol{w}=\boldsymbol{w}_*}$. Indeed we have the following theorem [19]:

- If $\forall i\in[|1;L_w|],\Re(\lambda_i)<0$, $\boldsymbol{w}_*$ is asymptotically stable

- If $\exists i \in [|1; L_w|], \Re(\lambda_i) > 0$, $\boldsymbol{w}_*$ is unstable
- If $\forall i \in [|1; L_w|], \Re(\lambda_i) \leq 0$ and $\Re(\lambda_{i_0}) = 0$ for $i_0 \in [|1; L_w|]$, we can not conclude from these values.

Then we should calculate $\frac{dh(\boldsymbol{w})}{d\boldsymbol{w}}|_{\boldsymbol{w}=\boldsymbol{w}_*}$.

$$\begin{aligned}\frac{dh(\boldsymbol{w})}{d\boldsymbol{w}} &= \frac{d}{d\boldsymbol{w}}\left[\frac{d}{d\sigma_\varepsilon^2}A(\sigma_\varepsilon^2)\nabla_{\boldsymbol{w}}\sigma_\varepsilon^2\right] \\ &= \frac{d^2A(\sigma_\varepsilon^2)}{d(\sigma_\varepsilon^2)^2}(\nabla_{\boldsymbol{w}}\sigma_\varepsilon^2)(\nabla_{\boldsymbol{w}}\sigma_\varepsilon^2)^T + \frac{dA(\sigma_\varepsilon^2)}{d\sigma_\varepsilon^2}\frac{d\big(\nabla_{\boldsymbol{w}}\sigma_\varepsilon^2\big)}{d\boldsymbol{w}}. \end{aligned} \tag{33}$$

Here, we are interested in the stability of the stationary point $\boldsymbol{w}_* = \sigma_s^2\big[\sigma_s^2\widetilde{\boldsymbol{H}}^*\widetilde{\boldsymbol{H}}^T + \sigma_{\boldsymbol{b}}^2\boldsymbol{I}_{L_w}\big]^{-1}\widetilde{\boldsymbol{H}}^*\boldsymbol{e}_D$. Thus,

$$\frac{dh(\boldsymbol{w})}{d\boldsymbol{w}}|_{\boldsymbol{w}=\boldsymbol{w}_*} = \frac{dA(\sigma_\varepsilon^2)}{d\sigma_\varepsilon^2}\Big[\sigma_s^2\widetilde{\boldsymbol{H}}\widetilde{\boldsymbol{H}}^H + \sigma_{\boldsymbol{b}}^2\boldsymbol{I}_{L_w}\Big]^T < 0 \tag{34}$$

since $(\sigma_s^2\widetilde{\boldsymbol{H}}\widetilde{\boldsymbol{H}}^H + \sigma_{\boldsymbol{b}}^2\boldsymbol{I})$ is a positive definite matrix and $\frac{dA(\sigma_\varepsilon^2)}{d\sigma_\varepsilon^2} < 0$. Thus, we have proved that the MMSE equalizer is the only stationary stable point of the MSQD-ℓ1 algorithm.

4.3. Asymptotic Steady-State MSE Analysis

4.3.1. convergence in mean

The ODE analysis holds for small step size μ. In this section, we study how it should be selected to guarantee convergence. In practice, μ should be chosen small enough. The maximum possible range for μ depends on the channel under consideration and its calculation is supplied in the Appendix (Appendix A). It can be summarized as follows:

Theorem 1. *For stepsize $0 < \mu < \frac{2}{\lambda_{max}}$ where λ_{max} is the largest eigenvalue of the matrix $\widetilde{\boldsymbol{H}}^*\mathbb{E}\left[\boldsymbol{s}(n)^*\phi'(\boldsymbol{w}_*\boldsymbol{x}(n))\boldsymbol{s}(n)^T\right]\widetilde{\boldsymbol{H}}^T + \sigma_{\boldsymbol{b}}^2\mathbb{E}\left[\phi'(\boldsymbol{w}_*\boldsymbol{x}(n))\right]\boldsymbol{I}_{Lw}$, the MSQD-$\ell$1 algorithm converges to the MMSE solution, regardless equalizer initialization.*

4.3.2. MSE equalizer analysis

The asymptotic covariance matrix of the residual error $\boldsymbol{\epsilon}(n) = (\boldsymbol{w}_n - \boldsymbol{w}_*)$ is denoted by $\boldsymbol{\Sigma}(n) = \mathbb{E}[\boldsymbol{\epsilon}(n)\boldsymbol{\epsilon}(n)^H]$. For small step size μ, we have $\bar{y}(n+D) \approx s(k_n)$ and according to [17] (p.102 p.103) $\boldsymbol{\Sigma}_\infty$ can be approximated as the solution of the following matrix equation, called Lyapunov's equation:

$$\boldsymbol{R}_f \boldsymbol{\Sigma}_{\boldsymbol{w}}(\infty) + \boldsymbol{\Sigma}_{\boldsymbol{w}}(\infty)\boldsymbol{R}_f^H = \mu\, \boldsymbol{R}_g \tag{35}$$

where, $\boldsymbol{R}_f = \frac{d}{d\boldsymbol{w}} h(\boldsymbol{w})|_{\boldsymbol{w}=\boldsymbol{w}_*}$ and

$$\boldsymbol{R}_g = -\mathbb{E}\left[H(\boldsymbol{w}_*, \boldsymbol{x}(n))H(\boldsymbol{w}_*, \boldsymbol{x}(n))^H\right] = -\mathbb{E}\left[|\phi(\bar{y}(n+D))|^2 \boldsymbol{x}^*(n)\boldsymbol{x}^T(n)\right]. \tag{36}$$

According to Eq.(34), we have

$$\boldsymbol{R}_f = \Big(\frac{dA(\sigma_\varepsilon^2)}{d\sigma_\varepsilon^2}\Big)[\sigma_s^2 \widetilde{\boldsymbol{H}}\widetilde{\boldsymbol{H}}^H + \sigma_b^2 \boldsymbol{I}_{Lw}]^T. \tag{37}$$

Let us denote $R_x = \mathbb{E}[\boldsymbol{x}(n)\boldsymbol{x}(n)^H] = \sigma_s^2 \widetilde{\boldsymbol{H}}\widetilde{\boldsymbol{H}}^{\boldsymbol{H}} + \sigma_{\boldsymbol{b}}^2 I_{L_w} = \boldsymbol{U}\, \boldsymbol{\Lambda}_x\, \boldsymbol{U}^H$ the eigenvalue decomposition of $\mathbb{E}[\boldsymbol{x}(n)\boldsymbol{x}(n)^H]$, where $\boldsymbol{\Lambda}_x = \mathrm{diag}(\lambda_1, \lambda_2, ..., \lambda_{L_w})$. We can easily verify that $\boldsymbol{R}_f = \boldsymbol{U}^* \boldsymbol{\Lambda}_f \boldsymbol{U}^T$ (see Eq.(37)), where $\boldsymbol{\Lambda}_f(i,i)_{i=1..L_w} \simeq \frac{-\lambda_i}{2N_s\sqrt{2\pi}(\sigma^2+\sigma_\varepsilon^2)^{\frac{3}{2}}}$. We can also write $\boldsymbol{R}_g \simeq \boldsymbol{U}^* \boldsymbol{\Lambda}_g \boldsymbol{U}^T$ and we detail the calculation of the diagonal elements of $\boldsymbol{\Lambda}_g$ in the Appendix (Appendix B). Thus, Eq.(35) becomes:

$$\boldsymbol{\Lambda}_f(\boldsymbol{U}^T\boldsymbol{\Sigma}_{\boldsymbol{w}}(\infty)\boldsymbol{U}^*) + (\boldsymbol{U}^T\boldsymbol{\Sigma}_{\boldsymbol{w}}(\infty)\boldsymbol{U}^*)\boldsymbol{\Lambda}_f = \mu\, \boldsymbol{\Lambda}_g \tag{38}$$

which shows that $\boldsymbol{U}^T\boldsymbol{\Sigma}_{\boldsymbol{w}}(\infty)\boldsymbol{U}^*$ is also diagonal: $\boldsymbol{\Sigma}_{\boldsymbol{w}}(\infty) = \boldsymbol{U}^* \boldsymbol{\Lambda}_{\boldsymbol{w}} \boldsymbol{U}^T$ with $\boldsymbol{\Lambda}_{\boldsymbol{w}} = \mathrm{diag}\{\lambda_{w_1}, \lambda_{w_2}, ..., \lambda_{w_{L_w}}\}$ and

$$\lambda_{w_i} \simeq \mu \frac{\Lambda_g(i,i)}{\Lambda_f(i,i)} \simeq 2\mu N_s \sqrt{2\pi}(\sigma^2 + \sigma_\varepsilon^2)^{\frac{3}{2}} \mathbb{E}\left[|\phi(s(k_n))|^2\right]. \tag{39}$$

4.3.3. MSE

In this stage, we calculate the MSE. Using the classical assumption that the tap coefficient vector $\boldsymbol{w}(n)$ are independent from the input data vector $\boldsymbol{x}(n)$ [21] and that the symbols $s(n)$ are independent of the noise $b(n)$, the MSE is expanded as follows:

$$\begin{aligned}
\sigma_{\varepsilon}^2(\infty) &= \lim_{n\to\infty} \mathbb{E}\left[|y(n) - s(k_n)|^2\right] \\
&= \lim_{n\to\infty} \mathbb{E}\left[|\boldsymbol{w}^T(n)\boldsymbol{x}(n) - s(k_n)|^2\right] \\
&= \mathbb{E}\left[|\boldsymbol{\epsilon}(\infty)^T\boldsymbol{x}(\infty) + \boldsymbol{w}_*{}^T\boldsymbol{x}(\infty) - s(k_\infty)|^2\right] \\
&= \mathbb{E}\left[\boldsymbol{\epsilon}(\infty)^T\boldsymbol{x}(\infty)\boldsymbol{x}(\infty)^H\boldsymbol{\epsilon}(\infty)^*\right] + \boldsymbol{w}_*{}^T\mathbb{E}\left[\boldsymbol{x}(\infty)\boldsymbol{x}(\infty)^H\right]\boldsymbol{w}_*{}^* \\
&+ \sigma_s^2 + 2\Re\left[\mathbb{E}\left[\boldsymbol{\epsilon}(\infty)^T\right]\mathbb{E}\left[\boldsymbol{x}(\infty)\boldsymbol{x}(\infty)^H\right]\boldsymbol{w}_*{}^* - \sigma_s^2\left(\mathbb{E}\left[\boldsymbol{\epsilon}(\infty)\right] + \boldsymbol{w}_*\right)^T\widetilde{\boldsymbol{H}}\boldsymbol{e}_D\right]
\end{aligned} \tag{40}$$

As $\mathbb{E}[\boldsymbol{\epsilon}(\infty)] = 0$ and using the independence between $\boldsymbol{w}_*$ and $\boldsymbol{x}(\infty)$ again, we get:

$$\sigma_{\varepsilon}^2(\infty) = \mathrm{Tr}\left(\boldsymbol{\Lambda}_x\boldsymbol{\Lambda}_w\right) + \boldsymbol{w}_*{}^T R_x \boldsymbol{w}_*{}^* + \sigma_s^2 - 2\sigma_s^2\Re[\boldsymbol{w}_*^T\widetilde{\boldsymbol{H}}\boldsymbol{e}_D] \tag{41}$$

where $\mathrm{Tr}\left(\boldsymbol{\Lambda}_x\boldsymbol{\Lambda}_w\right)$ is the residual error of the equalizer and $\boldsymbol{w}_*{}^T R_x \boldsymbol{w}_*{}^* + \sigma_s^2 - 2\sigma_s^2\Re[\boldsymbol{w}_*^T\widetilde{\boldsymbol{H}}\boldsymbol{e}_D]$ is the MMSE error term [2]. What makes the MSQD-ℓ1 algorithm converge close to the MMSE error is that the value of the first term is very low. However, we had to conduct a detailed performance analysis to calculate the exact value of the MSE.

5. Simulation Results

5.1. Adaptive adjustment of the kernel size

The kernel size σ of the Parzen window influences the convergence speed of the algorithm and its residual error. At the beginning of convergence, it is necessary to choose a large kernel size to enable interaction of the equalized symbol with all the

constellation symbols and thus ensure a fast convergence. On the contrary, when approaching the perfect equalization of transmitted symbols, a small kernel size has to be used to only allow interaction of each equalized symbol with the closest symbol in the constellation. As in [16], the kernel size was adaptively controlled assuming a linear relationship between the kernel size and the decision error:

$$\sigma(n) = aG(n) + b \tag{42}$$

where, $G(n) = \alpha G(n-1) + (1-\alpha) \underbrace{min}_{k=1,\ldots,N_s} \left((|y(n)|^2 - |s_k|^2)^2\right)$, α is a forgetting factor and (a, b) are empirically determined constants.

As mentioned in [16], the minimum of the stochastic cost function is a scaled version of the desired constellation. Then, the original symbols $|s_k|^2$ in Eq.(8) are substituted by $|s_k^c|^2$ as follows:

$$|s_k^c|^2 = Q(\sigma)|s_k|^2 \tag{43}$$

where $Q(\sigma)$ is the compensation factor that depends on the kernel size and is obtained by ensuring that the zero-ISI solution ($y(n) = s(k_n)$) is a minimum of $\mathbb{E}\left[J(\boldsymbol{w})\right]$:

$$\mathbb{E}\left[\nabla_{\boldsymbol{w}} J(\boldsymbol{w})\right] = \frac{1}{N_s} \sum_{k=1}^{N_s} \mathbb{E}\left[K_\sigma^{'}(|s(k_n)|^2 - Q(\sigma)|s_k|^2) s(k_n) \boldsymbol{x}^*(n)\right] = 0 \tag{44}$$

For MSQD-$\ell 2$ and MSQD-$\ell 1$, we adopt the same approach to determine the adequate $Q(\sigma)$ for each algorithm:

$$\mathbb{E}\left[\nabla_{\boldsymbol{w}} J_{\text{MSQD-}\ell 2}(\boldsymbol{w})\right] = 0 \rightarrow Q_{\text{MSQD-}\ell 2}(\sigma)$$
$$\mathbb{E}\left[\nabla_{\boldsymbol{w}} J_{\text{MSQD-}\ell 1}(\boldsymbol{w})\right] = 0 \rightarrow Q_{\text{MSQD-}\ell 1}(\sigma).$$

Thus, the real and imaginary parts of the compensated symbols are related to the true symbols in the constellation as follows: $|s_{r_k}^c|^2 = Q(\sigma)|s_{r_k}|^2$ and $|s_{i_k}^c|^2 =$

$Q(\sigma)|s_{i_k}|^2$. $Q(\sigma)$ is calculated numerically for each modulation. Fig.5 shows the compensation factor $Q(\sigma)$ for 16-QAM, 64-QAM and 256-QAM modulations when using the MSQD-ℓ1 algorithm. For the MSQD-ℓ1 and MSQD-ℓ2, we implement the same steps as the algorithm summarized in [16], using the appropriate cost functions and Q functions.

5.2. Numerical results

To compare blind equalization approaches proposed in this paper, with others existing in the literature, we choosed the same channel as the one used in [16]:

$$H_1 = [0.2258, 0.5161, 0.6452, -0.5161]^T. \tag{45}$$

Performance of the proposed MSQD-ℓ2 and MSQD-ℓ1 methods are compared with those of the CMA, MMA and SQD. For simulations, we employed an equalizer of length $L_w = 21$ initialized using the tap-centered strategy. Table 2 summarizes the parameters which were used to draw the curves in Fig.{6, 7, 8}.

To compare the performance of the proposed algorithms in terms of convergence speed, we set the step size μ for each algorithm such as they converge with the same speed. Thus, in Fig.6, Fig.7 and Fig.8, we can clearly notice that MSQD-ℓ2 and MSQD-ℓ1 outperform the SQD, MMA and CMA algorithms in terms of residual error for 16-QAM, 64-QAM and 256-QAM modulations. On the other hand, when we fix the value of μ for each algorithm such as they converge to the same MSE in Fig.9 and Fig.10, we notice that MSQD-ℓ2 and MSQD-ℓ1 converge faster. All these figures validate the MSQD-ℓ1 performance analysis that we have conducted, since the experimental curve of the MSQD-ℓ1 converges to the theoretical one. To study the performance of the MSQD-ℓ1 algorithm as a function of SNR, we draw in Fig.11 the Symbol Error Rate (SER) for the MMA, SQD, MSQD-ℓ1 algorithms and for an AWGN channel between SNR = 0 dB and SNR=20 dB for a 16-QAM modulation.

To plot these curves, we take the optimal equalizer for each algorithm with the same convergence rate. It is clear in this figure that the MSQD-$\ell 1$ algorithm outperforms the other algorithms in terms of the SER.We can also notice that for a value of SER equal to 10^{-2}, the MSQD-$\ell 1$ has a gain of 1.2 dB compared to the SQD. Moreover, its performance are very close to those obtained with an AWGN channel for any SNR.

The proposed methods were also tested with two other channels. Fig.12 and Fig.13 are obtained when using the complex channel with transfer function:

$$\begin{aligned} H_2(z) &= 10^{-2}[(4.1+1.09i)+(4.95+1.23i)z^{-1}+(6.72+1.7i)z^{-2}+(9.19+2.35i)z^{-3} \\ &+ (79.2+12.81i)z^{-4}+(39.6+8.71i)z^{-5}+(27.15+4.98i)z^{-6} \\ &+ (22.91+4.14i)z^{-7}+(12.87+1.54i)z^{-8}+(10.32+1.19i)z^{-9}] \end{aligned} \tag{46}$$

In Fig.12 and Fig.13, it is clear that MSQD-$\ell 1$ performs better than other equalizers. A typical channel met in communications is the frequency selective channel with an exponential decay profile. In Fig.14 and Fig.15, we show the performance of the proposed methods when using the channel of length $L_{h3} = 10$ with transfer function $H_3(z) = \sum_{l=0}^{L_h-1} h_3(l)z^{-l}$ with $h_3(l) \sim \mathcal{N}(0, Ge^{-\rho l})$ such that $\sum_{l=0}^{L_h-1} \mathbb{E}[|h_3(l)|^2] = 1$. For simulations, we chose $\rho = 0.7$. We can check that with this channel, the MSQD-$\ell 1$ ouperforms the other algorithms and converges to the MMSE equalizer.

The figures show that convergence of the MSQD-$\ell 1$ algorithm is achieved after less than 10000 iterations.

5.3. Computational complexity analysis

For a square M-QAM modulation, the computational complexity is summarized in table 1 where $N_s = \frac{\frac{\sqrt{M}}{2}!}{2!(\frac{\sqrt{(M)}}{2}-2)!} + \frac{\sqrt{M}}{2}$ and $N_s' = \frac{\sqrt{M}}{2}$ when $M > 4$ and

$N_s = N_s^{'}$ when $M = 4$. Therefore, we can conclude that the MSQD-$\ell 1$ is computationally less demanding than the SQD and slightly more demanding compared to the CMA. However, it requires many fewer iterations to converge to a low MSE. In Fig.9 and Fig.10, we can notice that the MSQD-$\ell 1$ converges about 10 times faster than the CMA. Fig.16 shows the number of multiplications required by each algorithm per iteration and Fig.17 shows the global computational cost needed to achieve convergence, according to Fig.9 and Fig.10. We can notice that the global computational complexity of the MSQD-$\ell 1$, is lower than that of SQD and CMA.

6. Conclusion

In this paper, we have proposed new criteria for kernel based blind equalization techniques that force the pdf of the real and imaginary parts of the equalizer output to match that of the true constellation real and imaginary parts by employing the Parzen window method to estimate the data pdf. Performance of the proposed methods has been compared with that of CMA and SQD. We have shown that they converge faster with a reduced residual error. The behaviour of the MSQD-$\ell 1$, most powerful proposed method, has been examined by relating the motion of the parameter estimate errors to a deterministic ODE. The analysis that we have conducted and simulation results prove that the MSQD-$\ell 1$ algorithm brings further validation of the pdf fitting approach for equalization in digital transmission. Although in this paper we only addressed QAM modulations, the proposed methods can be extended to any modulation since the equalization criterion represents some distance between the probability distribution of the equalizer output and that of transmitted data.

Table 1: Computational complexity of CMA, SQD and MSQD-$\ell 1$ algorithms for one iteration

	Multiplications	Exponent
CMA	$8L_w + 4$	0
SQD	$4N_s + 8L_w + 4$	N_s
MSQD$\ell 1$	$6N_s' + 8L_w + 2$	$2N_s'$

Table 2: parameter values used for simulations

16 QAM	CMA	SQD	MSQD-$\ell 2$	MSQD-$\ell 1$
μ	3.5×10^{-5}	10^{-4}	1.3×10^{-4}	7.7×10^{-4}
a	-	3.5	3.5	1.5
b	-	-9.5	-9.5	-1
$1-\alpha$	-	5×10^{-3}	5×10^{-3}	5×10^{-3}
E_0	-	7	7	5
64 QAM				
μ	3.3×10^{-7}	1.2×10^{-6}	9×10^{-7}	4.7×10^{-5}
a	-	3.5	3	2
b	-	-2	-18	-10
$1-\alpha$	-	10^{-3}	10^{-2}	10^{-3}
E_0	-	5	7	6.5
256 QAM				
μ	4×10^{-8}	1.5×10^{-7}	1.5×10^{-7}	7×10^{-5}
a	-	3.5	2.5	4
b	-	-4.5	-15	-1
$1-\alpha$	-	5×10^{-5}	10^{-4}	2×10^{-4}
E_0	-	7	20	7

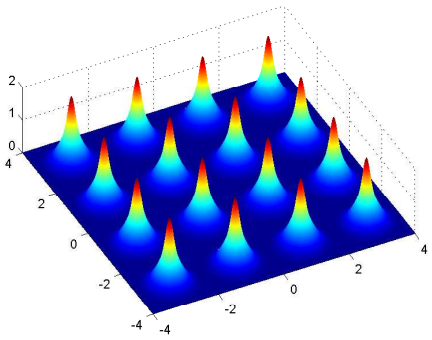

Figure 2: 2D pdfs : $M = 16$ modes for the SQD algorithm

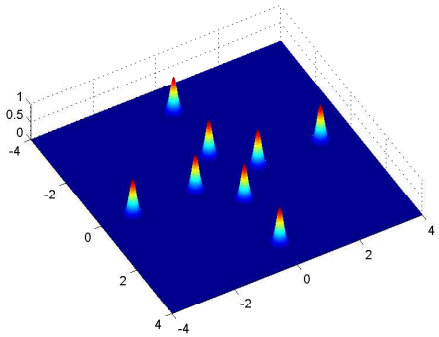

Figure 3: pdfs along real and imaginary axes: $2\sqrt{M} = 8$ modes for the MSQD-ℓp algorithm

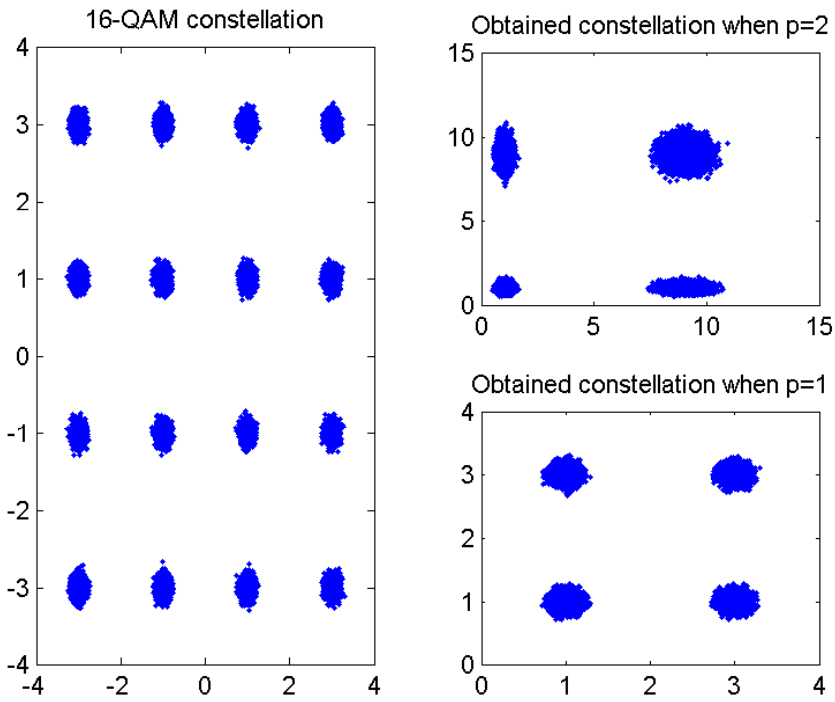

Figure 4: Transformed constellations for l_1, ($p = 1$) and l_2 ($p = 2$) norms of the real and imaginary transmitted data parts for a 16-QAM modulation

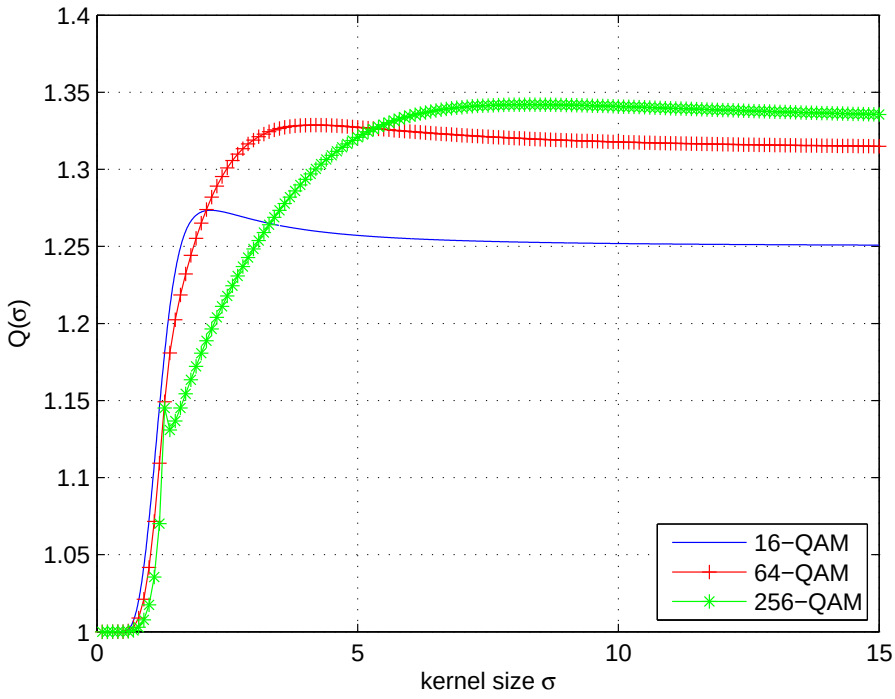

Figure 5: Evolution of the compensation factor $Q(\sigma)$ for MSQD-ℓ1 algorithm.

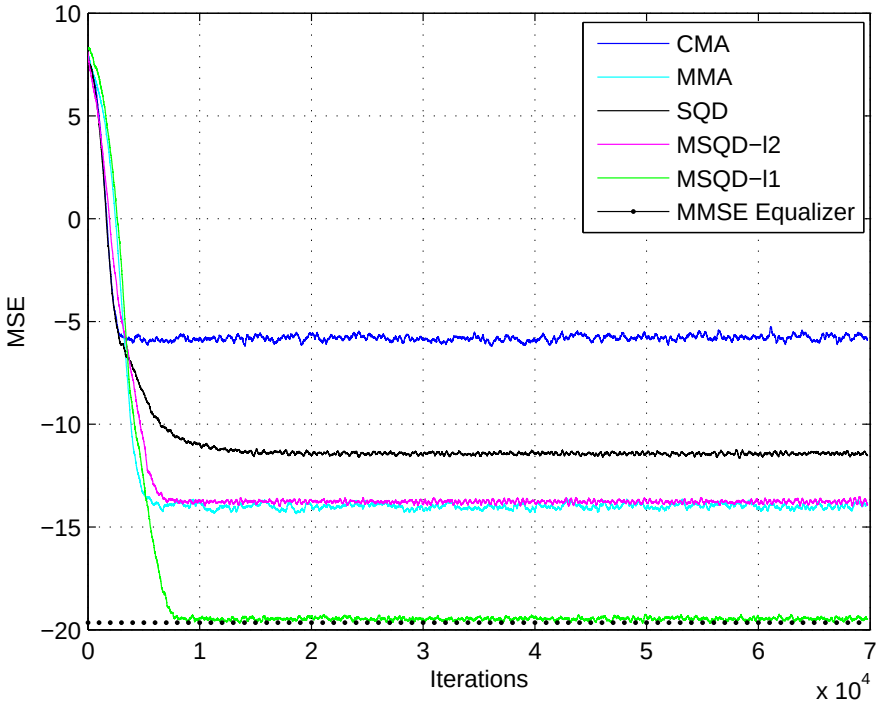

Figure 6: MSE (dB) for 16-QAM and SNR=30 dB using H_1.

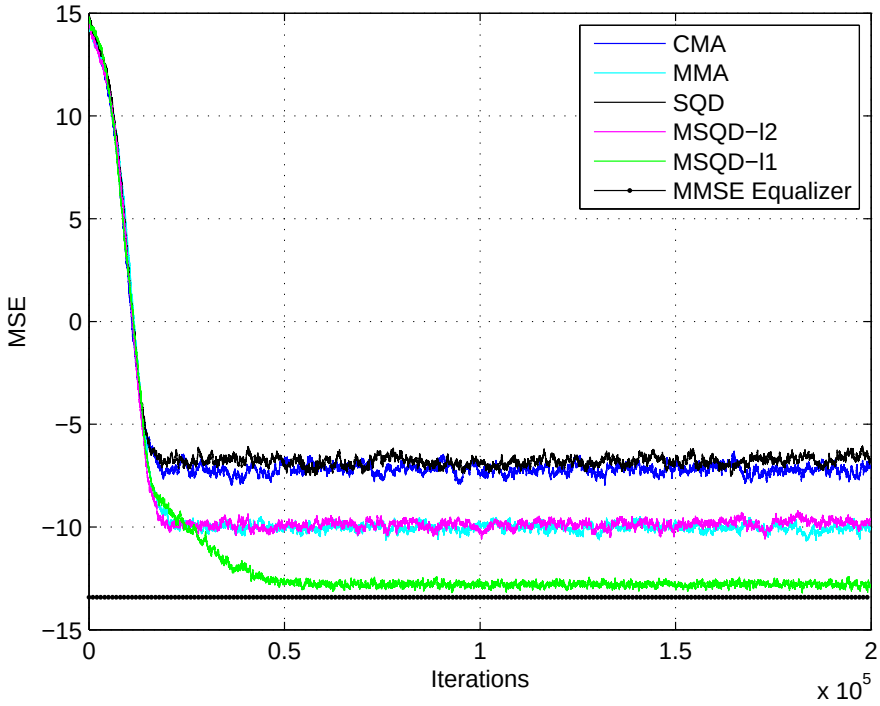

Figure 7: MSE (dB) for 64-QAM and SNR=30 dB using H_1.

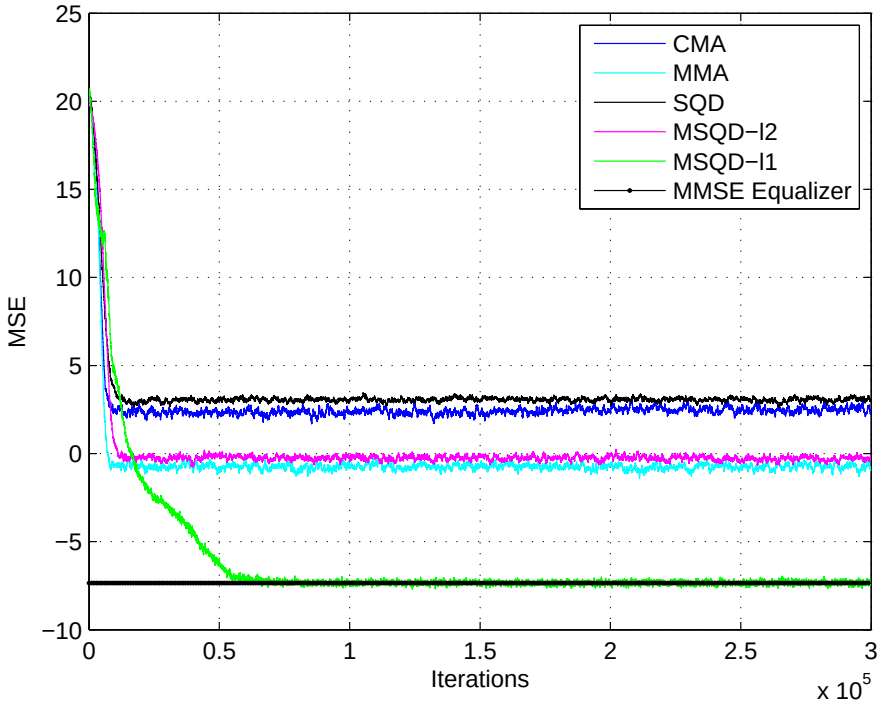

Figure 8: MSE (dB) for 256-QAM and SNR=30 dB using H_1.

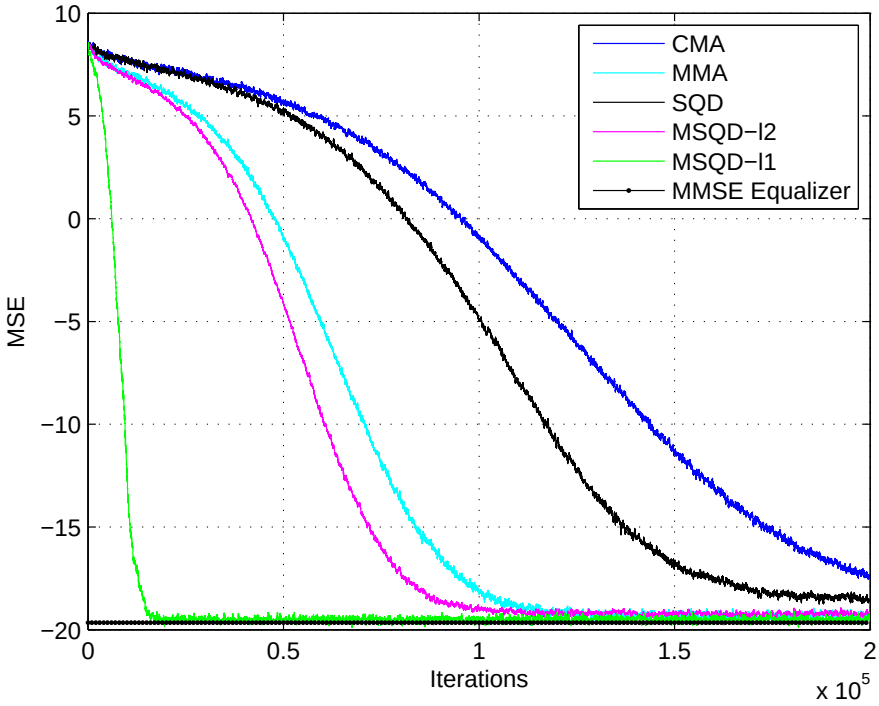

Figure 9: MSE (dB) for 16-QAM and SNR=30 dB using H_1.

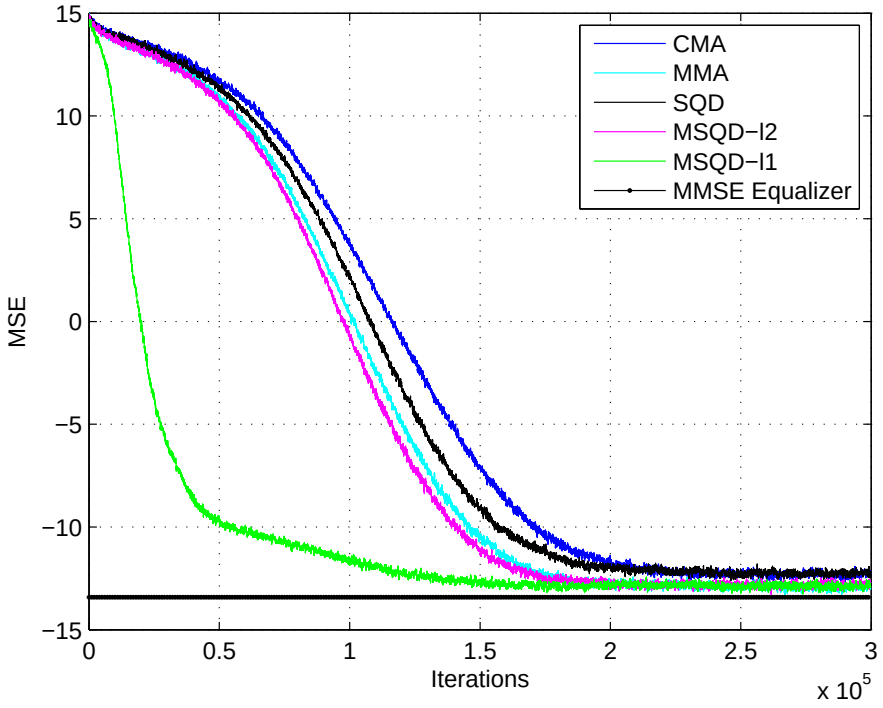

Figure 10: MSE (dB) for 64-QAM and SNR=30 dB using H_1.

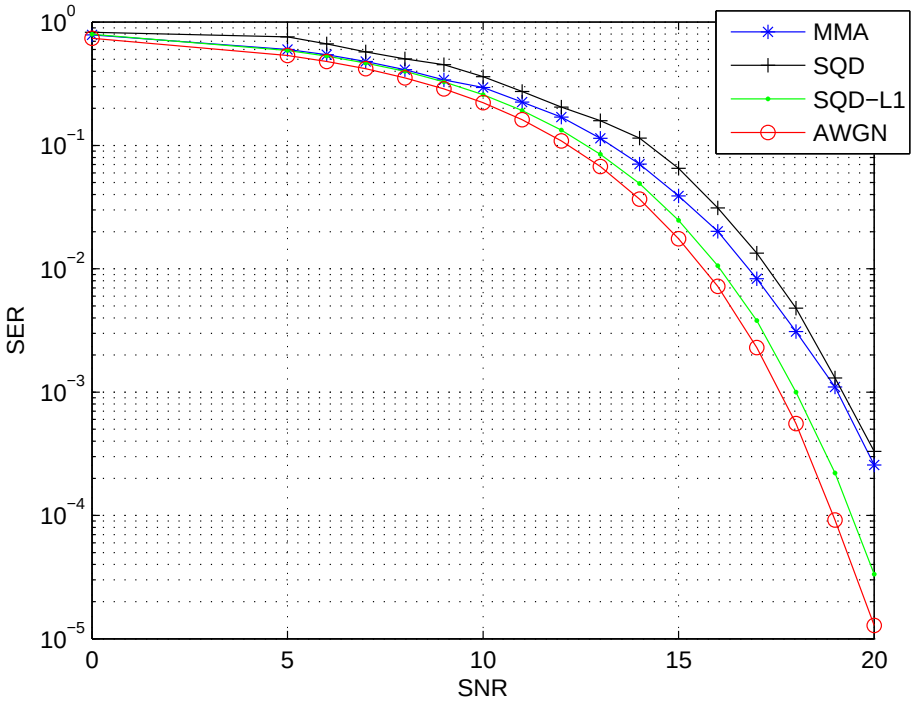

Figure 11: SER for MMA, SQD, MSQD-ℓ1 algorithms using H_1 and 16-QAM modulation.

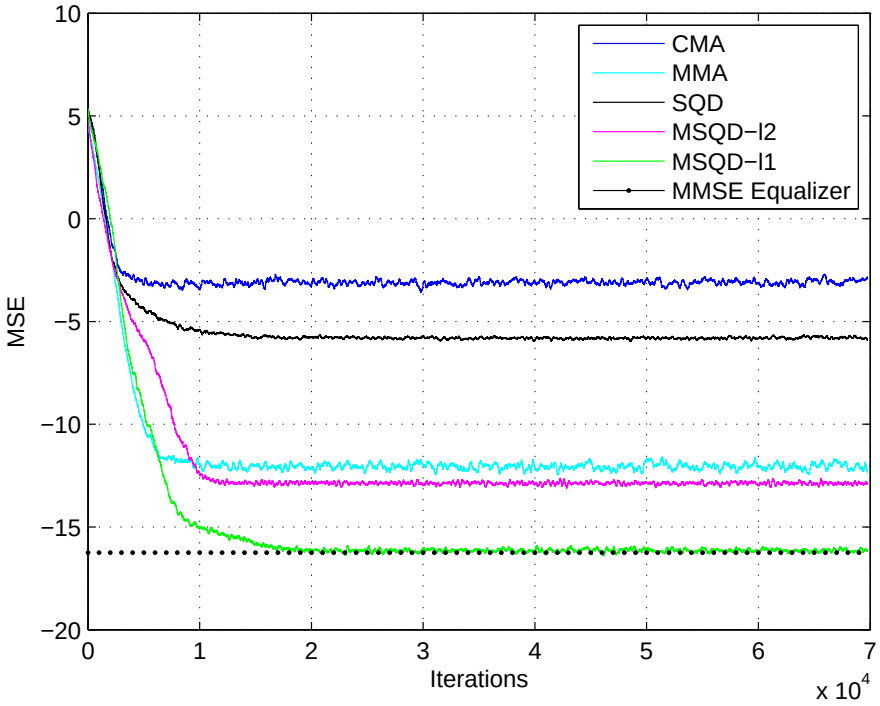

Figure 12: MSE (dB) for 16-QAM and SNR=30 dB using H_2.

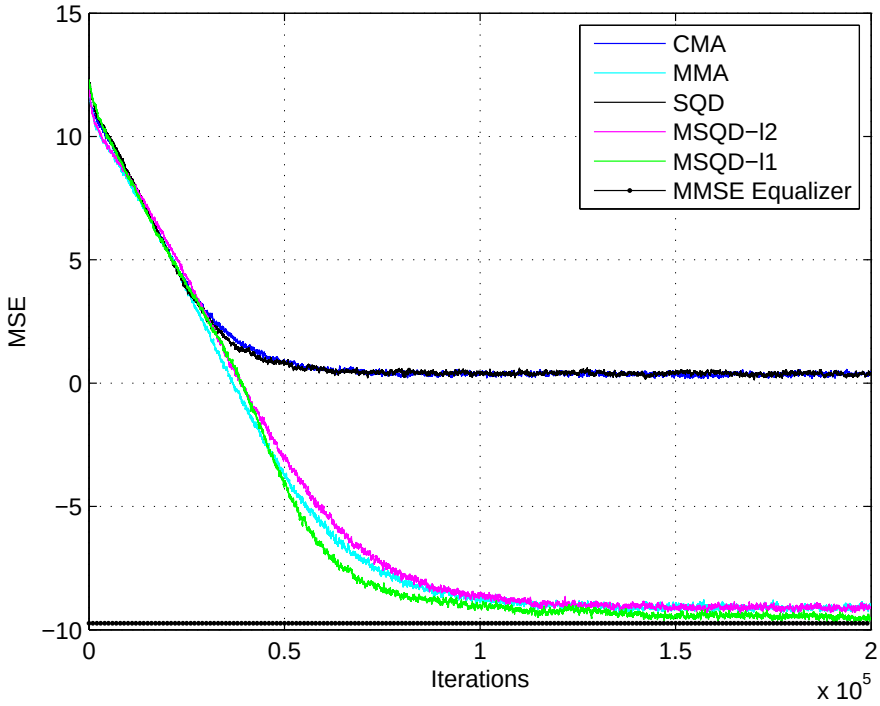

Figure 13: MSE (dB) for 64-QAM and SNR=30 dB using H_2.

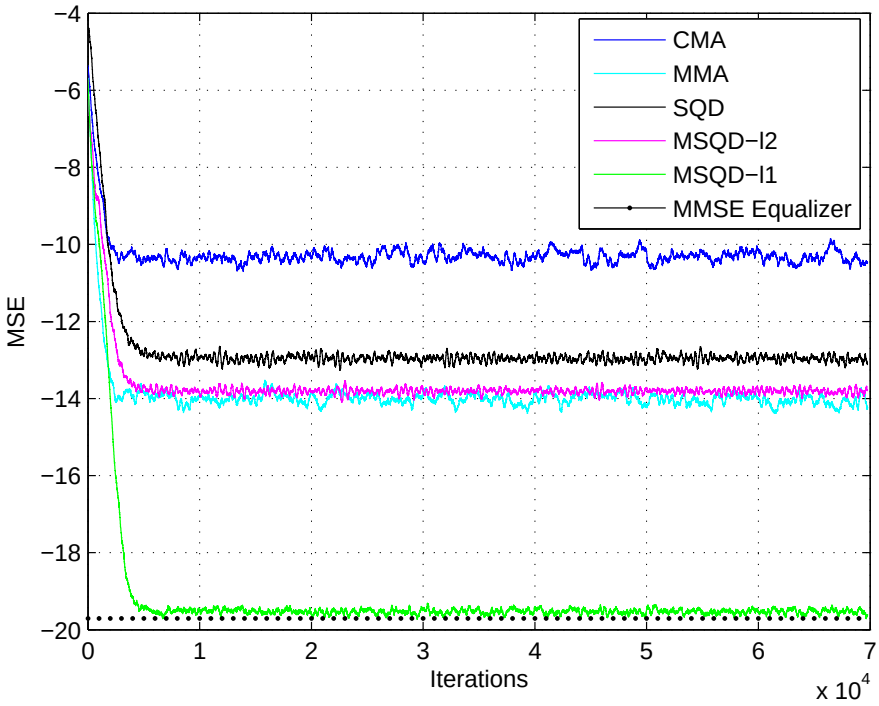

Figure 14: MSE (dB) for 16-QAM and SNR=30 dB using H_3.

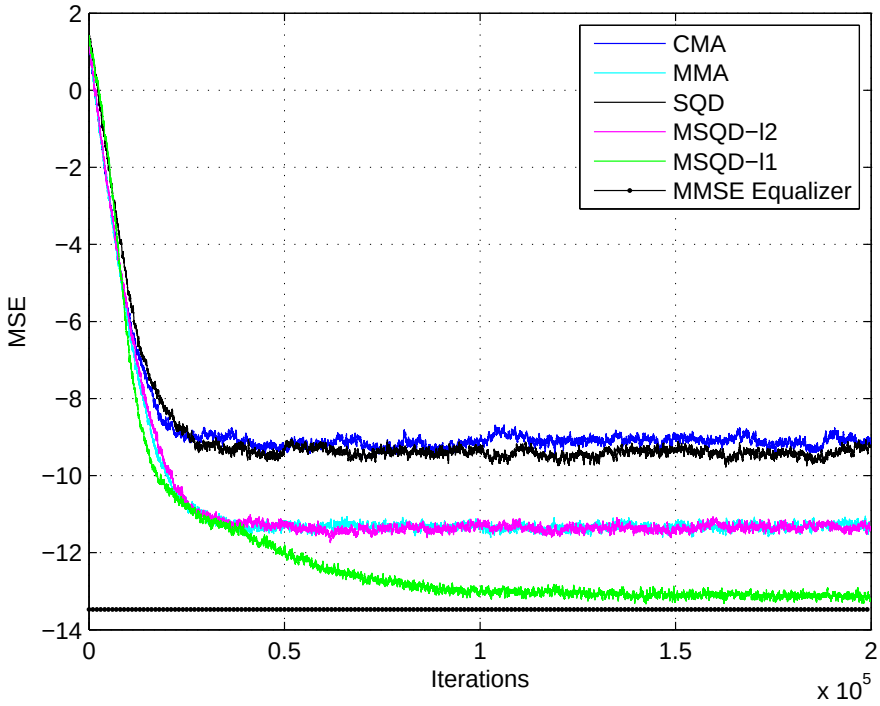

Figure 15: MSE (dB) for 64-QAM and SNR=30 dB using H_3.

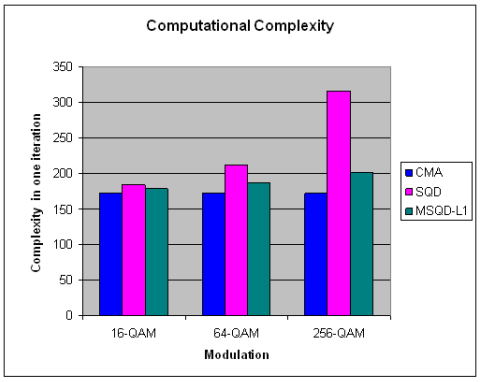

Figure 16: Number of multiplications per iteration for $\{16, 64, 256\}$-QAM modulations

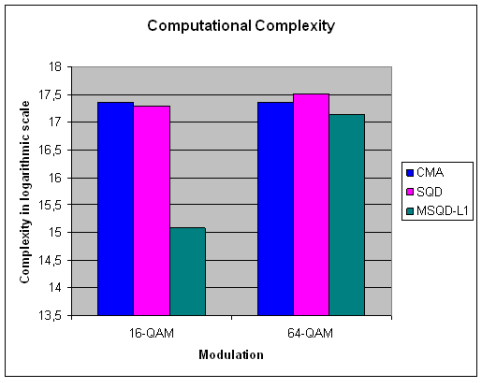

Figure 17: Number of multiplications needed by the equalizers to converge for $\{16, 64\}$-QAM modulations

Appendix A. Calculation of the maximum possible range for μ

Let us note $\bar{y}(n) = \boldsymbol{w}_* \boldsymbol{x}(n)$. At convergence, $y(n) - \bar{y}(n)$ is small and we can apply the taylor expansion to the function $\phi(y(n))$ (see Eq.(17)) at $\bar{y}(n)$. Then,

$$\begin{aligned} \phi(y_n) &= \phi(\bar{y}(n)) + \phi'(\bar{y}(n))(y(n) - \bar{y}(n)) \\ &= \phi(\bar{y}(n)) + \phi'(\bar{y}(n))(\widetilde{\boldsymbol{H}}\boldsymbol{s}(n) + \boldsymbol{b}(n))^T \boldsymbol{\epsilon}(n) \end{aligned} \tag{A.1}$$

where $\boldsymbol{\epsilon}(n) = \boldsymbol{w}(n) - \boldsymbol{w}_*$ and $\widetilde{\boldsymbol{H}}\boldsymbol{s}(n) + \boldsymbol{b}(n) = \boldsymbol{x}(n)$. In addition, we have

$$\boldsymbol{w}(n+1) = \boldsymbol{w}(n) - \mu\, \phi(y(n))\, \boldsymbol{x}(n)^*. \tag{A.2}$$

Thus, substracting $\boldsymbol{w}_*$ in both sides of Eq.(A.2) and using Eq.(A.1), we find that

$$\boldsymbol{\epsilon}(n+1) = \boldsymbol{\epsilon}(n) - \mu\big(\phi(\bar{y}(n))\boldsymbol{x}(n)^* + \phi'(\bar{y}(n))(\widetilde{\boldsymbol{H}}\boldsymbol{s}(n) + \boldsymbol{b}(n))^T \boldsymbol{\epsilon}(n)(\widetilde{\boldsymbol{H}}^* \boldsymbol{s}^*(n) + \boldsymbol{b}^*(n))\big) \tag{A.3}$$

Taking the expectation on the both side of Eq.(A.3) and using the independence between $\bar{y}(n)$ and $\boldsymbol{\epsilon}(n)$, as it was assumed in [21], we get

$$\begin{aligned} \mathbb{E}\left[\boldsymbol{\epsilon}(n+1)\right] &= \mathbb{E}\left[\boldsymbol{\epsilon}(n)\right] - \mu\left(\mathbb{E}\left[\phi(\bar{y}(n))\boldsymbol{x}^*(n)\right]\right. \\ &+ \left.\mathbb{E}[(\widetilde{\boldsymbol{H}}^* \boldsymbol{s}^*(n) + \boldsymbol{b}^*(n))\phi'(\widetilde{y}(n))(\widetilde{\boldsymbol{H}}\boldsymbol{s}(n) + \boldsymbol{b}(n))^T]\mathbb{E}\left[\boldsymbol{\epsilon}n\right]\right) \end{aligned} \tag{A.4}$$

In [22], the authors proved that $E\left[\phi(\bar{y}(n))\boldsymbol{x}^*(n)\right] = 0$ when the cost function approaches one of its minima. Thus, Eq.(A.4) can be simplified to

$$\mathbb{E}\left[\boldsymbol{\epsilon}(n+1)\right] = \left(\boldsymbol{I}_{Lw} - \mu(\sigma_s^2 \widetilde{\boldsymbol{H}}^* \widetilde{\boldsymbol{F}} \widetilde{\boldsymbol{H}}^T + \sigma_{\boldsymbol{b}}^2 \mathbb{E}\left[\phi'(\bar{y}_n)\right] \boldsymbol{I}_{Lw})\right) \mathbb{E}\left[\boldsymbol{\epsilon}(n)\right] \tag{A.5}$$

where $\widetilde{\boldsymbol{F}} = \frac{1}{\sigma_s^2}\mathbb{E}\left[\boldsymbol{s}^*(n)\phi'(\bar{y}(n))\boldsymbol{s}(n)^T\right]$. Consequently

$$\mathbb{E}\left[\boldsymbol{\epsilon}(n+1)\right] = \left(\boldsymbol{I}_{Lw} - \mu(\sigma_s^2 \widetilde{\boldsymbol{H}}^* \widetilde{\boldsymbol{F}} \widetilde{\boldsymbol{H}}^T + \sigma_{\boldsymbol{b}}^2 \mathbb{E}\{\phi'(\bar{y}(n))\}\boldsymbol{I}_{Lw})\right)^n \mathbb{E}\left[\boldsymbol{\epsilon}(0)\right] \tag{A.6}$$

This yields the following condition upon the step size of the algorithm for convergence of the mean error :

$$0 < \mu < \frac{2}{\lambda_{max}} \tag{A.7}$$

where λ_{max} is the largest eigenvalue of $\sigma_s^2 \widetilde{\boldsymbol{H}}^* \widetilde{\boldsymbol{F}} \widetilde{\boldsymbol{H}}^T + \sigma_{\boldsymbol{b}}^2 \mathbb{E}\left[\phi'(\bar{y}(n))\right] \boldsymbol{I}_{Lw}$.

Appendix B. Diagonalization of $\boldsymbol{R}_g$ in the basis $\boldsymbol{U}^*$

$$\begin{aligned}
\boldsymbol{R}_g &= -\mathbb{E}\left[H(\boldsymbol{w}_*, \boldsymbol{x}(n))H(\boldsymbol{w}_*, \boldsymbol{x}(n))^H\right] \\
&= -\mathbb{E}\left[\phi(s(n_k))\boldsymbol{x}^*(n)\boldsymbol{x}^T(n)\phi(s(n_k))^*\right] \\
&= -\boldsymbol{H}^*\mathbb{E}\left[|\phi(s(n_k))|^2\boldsymbol{s}(n)\boldsymbol{s}^T(n)\right]\boldsymbol{H}^T - \mathbb{E}\left[|\phi(s(n_k))|^2\boldsymbol{b}^*(n)\boldsymbol{b}^T(n)\right] \\
&= -\boldsymbol{H}^*\boldsymbol{D}\boldsymbol{H}^T - \sigma_b^2\mathbb{E}\left[|\phi(s(n_k))|^2\right] \qquad \text{(B.1)}
\end{aligned}$$

where, $\boldsymbol{D} = \text{diag}(d_1, ..., d_1, d_2, d_1, ..., d_1)$ with $d_1 = \sigma_s^2\mathbb{E}\left[|\phi(s(n_k))|^2\right]$ and $d_2 = \mathbb{E}\left[|s(n_k)|^2|\phi(s(n_k))|^2\right]$. We have cheked numerically that $|\frac{d_2-d_1}{d_1}|^2$ is very small (around 10^{-4}). Then, we can consider that $D \simeq d_1\boldsymbol{I}_{L+L_w-1}$. Thus, we obtain the following expression of R_g:

$$\begin{aligned}
R_g &\simeq -d_1\boldsymbol{H}^*\boldsymbol{I}_{L+L_w-1}\boldsymbol{H}^T - \sigma_b^2\mathbb{E}\left[|\phi(s(n_k))|^2\right] \\
&\simeq -\mathbb{E}\left[|\phi(s(n_k))|^2\right]\left(\sigma_s^2\boldsymbol{H}^*\boldsymbol{H}^T + \sigma_b^2\boldsymbol{I}_{L_w}\right) \\
&\simeq -\mathbb{E}\left[|\phi(s(n_k))|^2\right]\left(\boldsymbol{U}^*\boldsymbol{\Lambda}_x\boldsymbol{U}^T\right) \\
&\simeq \left(\boldsymbol{U}^*\boldsymbol{\Lambda}_g\boldsymbol{U}^T\right) \qquad \text{(B.2)}
\end{aligned}$$

where $\boldsymbol{\Lambda}_g(i,i) \simeq -\mathbb{E}\left[|\phi(s(n_k))|^2\right]\lambda_i$.

References

[1] D. Falconer, S. Ariyavisitakul, A. Benyamin-Seeyar, B. Eidson, Frequency domain equalization for single-carrier broadband wireless systems, Communications Magazine, IEEE 40 (4) (2002) 58–66.

[2] A. H. Sayed, Adaptive filters, John Wiley & Sons, Inc., Hoboken, New Jersey, 2008.

[3] Y. Sato, A method of self-recovering equalization for multilevel amplitude-modulation systems, IEEE Transactions on Communications 23 (6) (1975) 679 – 682.

[4] D. Godard, Self-recovering equalization and carrier tracking in two-dimensional data communication systems, IEEE Transactions on Communications 28 (11) (1980) 1867 – 1875.

[5] J. Treichler, B. Agee, A new approach to multipath correction of constant modulus signals, IEEE Transactions on Acoustics, Speech and Signal Processing 31 (2) (1983) 459 – 472.

[6] D. Jones, A normalized constant-modulus algorithm, in: Conference Record of the Twenty-Ninth Asilomar Conference on Signals, Systems and Computers, Vol. 1, 1995, pp. 694 –697.

[7] S. W. Kim, C.-H. Choi, An enhanced godard blind equalizer based on the analysis of transient phase, Signal Processing 81 (5) (2001) 919 – 926.

[8] T. Thaiupathump, L. He, S. A. Kassam, Square contour algorithm for blind equalization of QAM signals, Signal Processing 86 (11) (2006) 3357 – 3370.

[9] J. M. Filho, M. D. Miranda, M. T. Silva, A regional multimodulus algorithm for blind equalization of QAM signals: Introduction and steady-state analysis, Signal Processing 92 (11) (2012) 2643 – 2656.

[10] K. N. Oh, Y. O. Chin, Modified constant modulus algorithm: blind equalization and carrier phase recovery algorithm, in: IEEE International Conference on Communications, ICC Seattle, Vol. 1, 1995, pp. 498 –502.

[11] A. Labed, T. Chonavel, A. Aïssa-El-Bey, A. Belouchrani, Min-norm based alphabet-matching algorithm for adaptive blind equalisation of high-order qam signals, in: European transactions on telecommunications, Vol. 24, 2013, pp. 552–556.

[12] I. Santamaria, C. Pantaleon, L. Vielva, J. Principe, A fast algorithm for adaptive blind equalization using order-α renyi's entropy, ICASSP III (2002) 2657–2660.

[13] J. Sala-Alvarez, G. Vazquez-Grau, Statistical reference criteria for adaptive signal processing in digital communications, IEEE Transactions on Signal Processing 45 (1) (1997) 14 –31.

[14] I. Santamaria, C. Pantaleon, L. Vielva, J. Principe, Adaptive blind equalization through quadratic pdf matching, Proceedings of the European Signal Processing Conference, Toulouse, France II (2002) 289–292.

[15] M. Lazaro, I. Santamaria, C. Pantaleon, D. Erdogmus, K. E. Hild II, J. C. Principe, Blind equalization by sampled pdf fitting, 4th International Symposium on Independent Component Analysis and Blind Equalization, Nara, Japan (2003) 1041–1046.

[16] M. Lazaro, I. Santamaria, D. Erdogmus, K. Hild, C. Pantaleon, J. Principe, Stochastic blind equalization based on pdf fitting using parzen estimator, IEEE Transactions on Signal Processing 53 (2) (2005) 696 – 704.

[17] A. Benveniste, M. Metivier, P. Priouret, S. Wilson, Adaptive algorithms and stochastic approximations, Berlin ; New York : Springer, cop, 1990.

[18] J. Yang, J.-J. Werner, G. Dumont, The multimodulus blind equalization and its generalized algorithms, IEEE Journal on Communications 20 (5) (2002) 997 –1015.

[19] T. Chonavel, Statistical Signal Processing, Springer, 2003.

[20] C. Laot, N. Le Josse, A closed-form solution for the finite length constant modulus receiver, in: International Symposium on Information Theory. ISIT, 2005, pp. 865 –869.

[21] L. Garth, A dynamic convergence analysis of blind equalization algorithms, IEEE Transactions on Communications 49 (4) (2001) 624 –634.

[22] Y. Li, K. Liu, Static and dynamic convergence behavior of adaptive blind equalizers, IEEE Transactions on Signal Processing 44 (11) (1996) 2736 –2745.